

# **Mathematical Problems in Engineering**

## **Theory, Methods, and Applications**

---

**Special Issue**

**Optimization Theory, Methods, and Applications in Engineering**

**Guest Editors:** Jung-Fa Tsai, John Gunnar Carlsson, Dongdong Ge,  
Yi-Chung Hu, and Jianming Shi

---



# **Optimization Theory, Methods, and Applications in Engineering**

Mathematical Problems in Engineering

---

## **Optimization Theory, Methods, and Applications in Engineering**

Guest Editors: Jung-Fa Tsai, John Gunnar Carlsson,  
Dongdong Ge, Yi-Chung Hu, and Jianming Shi



Copyright © 2012 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  
E. M. Abdel-Rahman, Canada  
Rashid K. Abu Al-Rub, USA  
Salvatore Alfonzetti, Italy  
Igor Andrianov, Germany  
Sebastian Anita, Romania  
W. Assawinchaichote, Thailand  
Erwei Bai, USA  
Ezzat G. Bakhoum, USA  
José Manoel Balthazar, Brazil  
Rasajit K. Bera, India  
Jonathan N. Blakely, USA  
Stefano Boccaletti, Spain  
Daniela Boso, Italy  
M. Boutayeb, France  
Michael J. Brennan, UK  
John Burns, USA  
Salvatore Caddemi, Italy  
Piermarco Cannarsa, Italy  
Jose E. Capilla, Spain  
Carlo Cattani, Italy  
Marcelo M. Cavalcanti, Brazil  
Diego J. Celentano, Chile  
Mohammed Chadli, France  
Arindam Chakraborty, USA  
Yong-Kui Chang, China  
Michael J. Chappell, UK  
Kui Fu Chen, China  
Xinkai Chen, Japan  
Kue-Hong Chen, Taiwan  
Jyh Horng Chou, Taiwan  
Slim Choura, Tunisia  
Cesar Cruz-Hernandez, Mexico  
Swagatam Das, India  
Filippo de Monte, Italy  
Maria de Pinho, Portugal  
Antonio Desimone, Italy  
Yannis Dimakopoulos, Greece  
Baocang Ding, China  
Joao B. R. Do Val, Brazil  
Daoyi Dong, Australia  
Balram Dubey, India  
Horst Ecker, Austria  
M. Onder Efe, Turkey  
Elmetwally Elabbasy, Egypt  
Alex Elias-Zuniga, Mexico  
Anders Eriksson, Sweden  
Vedat S. Erturk, Turkey  
Qi Fan, USA  
Moez Feki, Tunisia  
Ricardo Femat, Mexico  
Rolf Findeisen, Germany  
R. A. Fontes Valente, Portugal  
C. R. Fuerte-Esquivel, Mexico  
Zoran Gajic, USA  
Ugo Galvanetto, Italy  
Xin-Lin Gao, USA  
Furong Gao, Hong Kong  
Behrouz Gatmiri, Iran  
Oleg V. Gendelman, Israel  
Didier Georges, France  
P. Batista Gonçalves, Brazil  
Oded Gottlieb, Israel  
Fabrizio Greco, Italy  
Quang Phuc Ha, Australia  
M. R. Hajj, USA  
Thomas Hanne, Switzerland  
Tasawar Hayat, Pakistan  
Katica R. Hedrih, Serbia  
M.I. Herreros, Spain  
Wei-Chiang Hong, Taiwan  
J. Horacek, Czech Republic  
Chuangxia Huang, China  
Gordon Huang, Canada  
Yi Feng Hung, Taiwan  
Hai-Feng Huo, China  
Asier Ibeas, Spain  
Anuar Ishak, Malaysia  
Reza Jazar, Australia  
Zhijian Ji, China  
J. Jiang, China  
J. J. Judice, Portugal  
Tadeusz Kaczorek, Poland  
Tamas Kalmar-Nagy, USA  
Tomasz Kapitaniak, Poland  
Hamid Reza Karimi, Norway  
Metin O. Kaya, Turkey  
Nikolaos Kazantzis, USA  
Farzad Khani, Iran  
Kristian Krabbenhoft, Australia  
Jurgen Kurths, Germany  
Claude Lamarque, France  
F. Lamnabhi-Lagarrigue, France  
Marek Lefik, Poland  
Stefano Lenci, Italy  
Roman Lewandowski, Poland  
Ming Li, China  
Shanling Li, Canada  
Tao Li, China  
Jian Li, China  
Shihua Li, China  
Teh-Lu Liao, Taiwan  
P. Liatsis, UK  
Jui-Sheng Lin, Taiwan  
Shueei M. Lin, Taiwan  
Yuji Liu, China  
Wanquan Liu, Australia  
Bin Liu, Australia  
Paolo Lonetti, Italy  
V. C. Loukopoulos, Greece  
Junguo Lu, China  
Chien-Yu Lu, Taiwan  
Alexei Mailybaev, Brazil  
Manoranjan Maiti, India  
O. Daniel Makinde, South Africa  
R. Martinez-Guerra, Mexico  
Driss Mehdi, France  
Roderick Melnik, Canada  
Xinzhu Meng, China  
Y. Vladimirovich Mikhlin, Ukraine  
G. Milovanovic, Serbia  
Ebrahim Momoniat, South Africa  
Trung Nguyen Thoi, Vietnam  
Hung Nguyen-Xuan, Vietnam  
Ben T. Nohara, Japan  
Anthony Nouy, France

Sotiris K. Ntouyas, Greece	Massimo Scalia, Italy	Youqing Wang, China
Gerard Olivar, Colombia	Mohamed A. Seddeek, Egypt	Yongqi Wang, Germany
Claudio Padra, Argentina	Alexander P. Seyranian, Russia	Cheng C. Wang, Taiwan
Francesco Pellicano, Italy	Leonid Shaikhet, Ukraine	Moran Wang, USA
Matjaz Perc, Slovenia	Cheng Shao, China	Yijing Wang, China
Vu Ngoc Phat, Vietnam	Daichao Sheng, Australia	Gerhard-Wilhelm Weber, Turkey
A. Pogromsky, The Netherlands	Tony Sheu, Taiwan	Jeroen A. S. Witteveen, USA
Seppo Pohjolainen, Finland	Jian-Jun Shu, Singapore	Kwok-Wo Wong, Hong Kong
Stanislav Potapenko, Canada	Zhan Shu, UK	Ligang Wu, China
Sergio Preidikman, USA	Dan Simon, USA	Zheng-Guang Wu, China
Carsten Proppe, Germany	Luciano Simoni, Italy	Wang Xing-yuan, China
Hector Puebla, Mexico	Grigori M. Sisoev, UK	X. Frank Xu, USA
Justo Puerto, Spain	Christos H. Skiadas, Greece	Xuping Xu, USA
Dane Quinn, USA	Davide Spinello, Canada	Jun-Juh Yan, Taiwan
K. Ramamani Rajagopal, USA	Sri Sridharan, USA	Xing-Gang Yan, UK
Gianluca Ranzi, Australia	Rolf Stenberg, Finland	Suh-Yuh Yang, Taiwan
Sivaguru Ravindran, USA	Changyin Sun, China	Mahmoud T. Yassen, Egypt
G. Rega, Italy	Jitao Sun, China	Mohammad I. Younis, USA
Pedro Ribeiro, Portugal	Xi-Ming Sun, China	Huang Yuan, Germany
J. Rodellar, Spain	Andrzej Swierniak, Poland	S. P. Yung, Hong Kong
R. Rodriguez-Lopez, Spain	Allen Tannenbaum, USA	Ion Zaballa, Spain
A. J. Rodriguez-Luis, Spain	Cristian Toma, Romania	Arturo Zavala-Rio, Mexico
Ignacio Romero, Spain	Irina N. Trendafilova, UK	Ashraf M. Zenkour, Saudi Arabia
Hamid Ronagh, Australia	Alberto Trevisani, Italy	Yingwei Zhang, USA
Carla Roque, Portugal	Jung-Fa Tsai, Taiwan	Xu Zhang, China
Rubén Ruiz García, Spain	John Tsinias, Greece	Lu Zhen, China
Manouchehr Salehi, Iran	Kuppalapalle Vajravelu, USA	Liancun Zheng, China
Miguel A. F. Sanjuán, Spain	Victoria Vampa, Argentina	Jian Guo Zhou, UK
Ilmar Ferreira Santos, Denmark	Josep Vehi, Spain	Zexuan Zhu, China
Nickolas S. Sapidis, Greece	Stefano Vidoli, Italy	Mustapha Zidi, France
Bozidar Sarler, Slovenia	Xiaojun Wang, China	
Andrey V. Savkin, Australia	Dan Wang, China	

# Contents

**Optimization Theory, Methods, and Applications in Engineering**, Jung-Fa Tsai, John Gunnar Carlsson, Dongdong Ge, Yi-Chung Hu, and Jianming Shi  
Volume 2012, Article ID 759548, 7 pages

**A Stone Resource Assignment Model under the Fuzzy Environment**, Liming Yao, Jiuping Xu, and Feng Guo  
Volume 2012, Article ID 265837, 26 pages

**Dynamic Programming and Heuristic for Stochastic Uncapacitated Lot-Sizing Problems with Incremental Quantity Discount**, Yuli Zhang, Shiji Song, Cheng Wu, and Wenjun Yin  
Volume 2012, Article ID 582323, 21 pages

**Sparse Signal Recovery via ECME Thresholding Pursuits**, Heping Song and Guoli Wang  
Volume 2012, Article ID 478931, 22 pages

**Variable Neighborhood Search for Parallel Machines Scheduling Problem with Step Deteriorating Jobs**, Wenming Cheng, Peng Guo, Zeqiang Zhang, Ming Zeng, and Jian Liang  
Volume 2012, Article ID 928312, 20 pages

**Global Sufficient Optimality Conditions for a Special Cubic Minimization Problem**, Xiaomei Zhang, Yanjun Wang, and Weimin Ma  
Volume 2012, Article ID 871741, 16 pages

**A Two-Stage DEA to Analyze the Effect of Entrance Deregulation on Iranian Insurers: A Robust Approach**, Seyed Gholamreza Jalali Naini and Hamid Reza Nouralizadeh  
Volume 2012, Article ID 423524, 24 pages

**Solving Constrained Global Optimization Problems by Using Hybrid Evolutionary Computing and Artificial Life Approaches**, Jui-Yu Wu  
Volume 2012, Article ID 841410, 36 pages

**A Hybrid Algorithm Based on ACO and PSO for Capacitated Vehicle Routing Problems**, Yucheng Kao, Ming-Hsien Chen, and Yi-Ting Huang  
Volume 2012, Article ID 726564, 17 pages

**New Bounds for Ternary Covering Arrays Using a Parallel Simulated Annealing**, Himer Avila-George, Jose Torres-Jimenez, and Vicente Hernández  
Volume 2012, Article ID 897027, 19 pages

**A VNS Metaheuristic with Stochastic Steps for Max 3-Cut and Max 3-Section**, Ai-fan Ling  
Volume 2012, Article ID 475018, 16 pages

**Opposition-Based Barebones Particle Swarm for Constrained Nonlinear Optimization Problems**, Hui Wang  
Volume 2012, Article ID 761708, 12 pages

**A New Hybrid Nelder-Mead Particle Swarm Optimization for Coordination Optimization of Directional Overcurrent Relays**, An Liu and Ming-Ta Yang  
Volume 2012, Article ID 456047, 18 pages

**Evaluating the Performance of Taiwan Homestay Using Analytic Network Process,**  
Yi-Chung Hu, Jen-Hung Wang, and Ru-Yu Wang  
Volume 2012, Article ID 827193, 24 pages

**Mixed Mortar Element Method for  $P_1^{NC}/P_0$  Element and Its Multigrid Method for the Incompressible Stokes Problem,** Yaqin Jiang and Jinru Chen  
Volume 2012, Article ID 979307, 18 pages

**Applying Neural Networks to Prices Prediction of Crude Oil Futures,** John Wei-Shan Hu,  
Yi-Chung Hu, and Ricky Ray-Wen Lin  
Volume 2012, Article ID 959040, 12 pages

**A Novel Method for Technology Forecasting and Developing R&D Strategy of Building Integrated Photovoltaic Technology Industry,** Yu-Jing Chiu and Tao-Ming Ying  
Volume 2012, Article ID 273530, 24 pages

**Solving the Tractor and Semi-Trailer Routing Problem Based on a Heuristic Approach,**  
Hongqi Li, Yue Lu, Jun Zhang, and Tianyi Wang  
Volume 2012, Article ID 182584, 12 pages

**Combining Diffusion and Grey Models Based on Evolutionary Optimization Algorithms to Forecast Motherboard Shipments,** Fu-Kwun Wang, Yu-Yao Hsiao, and Ku-Kuang Chang  
Volume 2012, Article ID 849634, 10 pages

**Adaptive Method for Solving Optimal Control Problem with State and Control Variables,**  
Louadj Kahina and Aidene Mohamed  
Volume 2012, Article ID 209329, 15 pages

**A Hybrid Network Model to Extract Key Criteria and Its Application for Brand Equity Evaluation,** Chin-Yi Chen and Chung-Wei Li  
Volume 2012, Article ID 971303, 14 pages

**Solving Packing Problems by a Distributed Global Optimization Algorithm,** Nian-Ze Hu,  
Han-Lin Li, and Jung-Fa Tsai  
Volume 2012, Article ID 931092, 13 pages

**Hybrid Optimization Approach for the Design of Mechanisms Using a New Error Estimator,**  
A. Sedano, R. Sancibrian, A. de Juan, F. Viadero, and F. Egaña  
Volume 2012, Article ID 151590, 20 pages

**Goal-Programming-Driven Genetic Algorithm Model for Wireless Access Point Deployment Optimization,** Chen-Shu Wang and Ching-Ter Chang  
Volume 2012, Article ID 780637, 14 pages

**Multithreshold Segmentation Based on Artificial Immune Systems,** Erik Cuevas,  
Valentin Osuna-Enciso, Daniel Zaldivar, Marco Pérez-Cisneros, and Humberto Sossa  
Volume 2012, Article ID 874761, 20 pages

**Quality Improvement and Robust Design Methods to a Pharmaceutical Research and Development**, Byung Rae Cho and Sangmun Shin  
Volume 2012, Article ID 193246, 14 pages

**A Label Correcting Algorithm for Partial Disassembly Sequences in the Production Planning for End-of-Life Products**, Pei-Fang (Jennifer) Tsai  
Volume 2012, Article ID 569429, 13 pages

**Improved Degree Search Algorithms in Unstructured P2P Networks**, Guole Liu, Haipeng Peng, Lixiang Li, Yixian Yang, and Qun Luo  
Volume 2012, Article ID 923023, 18 pages

**The Number of Students Needed for Undecided Programs at a College from the Supply-Chain Viewpoint**, Jin-Ling Lin, Jy-Hsin Lin, and Kao-Shing Hwang  
Volume 2012, Article ID 276519, 12 pages

**A Review of Deterministic Optimization Methods in Engineering and Management**, Ming-Hua Lin, Jung-Fa Tsai, and Chian-Son Yu  
Volume 2012, Article ID 756023, 15 pages

**A Hybrid Genetic Algorithm for the Multiple Crossdocks Problem**, Zhaowei Miao, Ke Fu, and Feng Yang  
Volume 2012, Article ID 316908, 18 pages

**A Nonlinear Multiobjective Bilevel Model for Minimum Cost Network Flow Problem in a Large-Scale Construction Project**, Jiuping Xu, Yan Tu, and Ziqiang Zeng  
Volume 2012, Article ID 463976, 40 pages

**A Selection Approach for Optimized Problem-Solving Process by Grey Relational Utility Model and Multicriteria Decision Analysis**, Chih-Kun Ke and Mei-Yu Wu  
Volume 2012, Article ID 293137, 14 pages

**Predictor-Corrector Primal-Dual Interior Point Method for Solving Economic Dispatch Problems: A Postoptimization Analysis**, Antonio Roberto Balbo, Márcio Augusto da Silva Souza, Edméa Cássia Baptista, and Leonardo Nepomuceno  
Volume 2012, Article ID 376546, 26 pages

**Applying Hierarchical Bayesian Neural Network in Failure Time Prediction**, Ling-Jing Kao and Hsin-Fen Chen  
Volume 2012, Article ID 953848, 11 pages

**A Fuzzy Dropper for Proportional Loss Rate Differentiation under Wireless Network with a Multi-State Channel**, Yu-Chin Szu  
Volume 2012, Article ID 827137, 16 pages

**A Cost-Effective Planning Graph Approach for Large-Scale Web Service Composition**, Szu-Yin Lin, Guan-Ting Lin, Kuo-Ming Chao, and Chi-Chun Lo  
Volume 2012, Article ID 783476, 21 pages

**An Optimal Classification Method for Biological and Medical Data**, Yao-Huei Huang,  
Yu-Chien Ko, and Hao-Chun Lu  
Volume 2012, Article ID 398232, 17 pages

**A Modified PSO Algorithm for Minimizing the Total Costs of Resources in MRCSP**,  
Mohammad Khalilzadeh, Fereydoon Kianfar, Ali Shirzadeh Chaleshtari, Shahram Shadrokh,  
and Mohammad Ranjbar  
Volume 2012, Article ID 365697, 18 pages

**Comment on “Highly Efficient Sigma Point Filter for Spacecraft Attitude and Rate  
Estimation”**, Baiqing Hu, Lubin Chang, An Li, and Fangjun Qin  
Volume 2012, Article ID 170391, 5 pages

**A Filter Algorithm with Inexact Line Search**, Meiling Liu, Xueqian Li, and Qinmin Wu  
Volume 2012, Article ID 349178, 20 pages

**Optimal Incentive Pricing on Relaying Services for Maximizing Connection Availability in  
Multihop Cellular Networks**, Ming-Hua Lin and Hao-Jan Hsu  
Volume 2012, Article ID 324245, 17 pages

## *Editorial*

# **Optimization Theory, Methods, and Applications in Engineering**

**Jung-Fa Tsai,<sup>1</sup> John Gunnar Carlsson,<sup>2</sup> Dongdong Ge,<sup>3</sup>  
Yi-Chung Hu,<sup>4</sup> and Jianming Shi<sup>5</sup>**

<sup>1</sup> Department of Business Management, National Taipei University of Technology, No. 1, Section 3, Chung-Hsiao East Road, Taipei 10608, Taiwan

<sup>2</sup> Program in Industrial and Systems Engineering, University of Minnesota, 111 Church Street SE, Minneapolis, MN 55455, USA

<sup>3</sup> Antai College of Economics & Management, Shanghai Jiao Tong University, Room 423, North Building, 535 Fahua Zhen Road, Changning, Shanghai 200052, China

<sup>4</sup> Department of Business Administration, Chung Yuan Christian University, Chung-Li 320, Taiwan

<sup>5</sup> Department of Information and Electronic Engineering, Muroran Institute of Technology, 27-1 Mizumoto-cho, Muroran 050-8585, Japan

Correspondence should be addressed to Jung-Fa Tsai, jftsai@ntut.edu.tw

Received 4 September 2012; Accepted 4 September 2012

Copyright © 2012 Jung-Fa Tsai et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Over years of development, optimization theory and methods have grown in their ability to handle various practical problems. In light of advances in computing systems, optimization approaches have become one of the most promising techniques for engineering applications. To close the gap between optimization theory and the practice of engineering, this special issue intends to provide the details of recent advances of optimization sciences and promote the applications of optimization methods in engineering. This special issue also provides a forum for researchers and practitioners to review and disseminate quality research work on optimization approaches and their applications in the context of engineering and to identify critical issues for further developments.

The papers accepted in the special issue include original research articles as well as review articles on all aspects of optimization including deterministic approaches, continuous, mixed-integer and discrete optimization, stochastic optimization, particle swarm optimization, neural network, simulated annealing, genetic algorithm, and hybrid methods. Some of the papers are dedicated to the development of advanced optimization methods for direct or indirect use in engineering problems such as network, scheduling, production planning, industrial engineering, and manufacturing systems. Contributions containing computational

issues, search strategies, and modeling and solution techniques to practical problems such as multicriteria decision making and management information system are also involved. According to the characteristics of the accepted papers, the special issue is organized as the following four parts and each part is composed of several important papers to the part's scope.

### *Deterministic Optimization*

The paper "*Global sufficient optimality conditions for a special cubic minimization problem*" by X. Zhang et al. presents some sufficient global optimality conditions for a special cubic minimization problem with box constraints or binary constraints by extending the global subdifferential approach proposed by V. Jeyakumar in 2006. Numerical examples demonstrate that the optimality conditions can effectively be used for identifying global minimizers of the certain nonconvex cubic minimization problem.

Baiqing Hu, Lubin Chang, An Li, and Fangjun Qin propose a comment on "*Highly efficient sigma point filter for spacecraft attitude and rate estimation*" by Fan and Zeng (2009). In the comment, the geometric simplex sigma points (GSSPs) can be derived from a numerical integration formula of degree 2 and can be generalized for different degrees, that is, the GSSP can be derived through the orthogonal transformation from the basic points set of degree 2. Moreover, their method can be used to construct more accurate sigma points set for certain dynamic problems.

*"Adaptive method for solving optimal control problem with state and control variables"* by L. Kahina and A. Mohamed solves the problem of optimal control with state and control variables by adaptive method and technology of linear programming. The obtained results show that it is possible to construct very fast solving algorithms based on the adequate consideration of the dynamic structure of the problem in question.

M. Liu and X. Li propose a filter algorithm with inexact line search for nonlinear programming problems that ensures superlinear local convergence without second order correction steps. In their paper "*A filter algorithm with inexact line search*" a filter is constructed by employing the norm of the gradient of the Lagrangian function to the infeasibility measure. Numerical experiences show the efficiency of their filter algorithm.

The paper "*Predictor-corrector primal-dual interior point method for solving economic dispatch problems: a postoptimization analysis*" by A. R. Balbo et al. proposes a predictor-corrector primal-dual interior point method involving line search procedures to solve the economic dispatch problem. Their method applies the Fibonacci search technique in the predictor step and an Armijo line search in the corrector step. The comparative results with other methods described in the literature demonstrate the efficiency of their method.

The paper "*An optimal classification method for biological and medical data*" by Y.-H. Huang et al. proposes a union of hyper spheres by the mixed-integer nonlinear program to classify biological and medical datasets and a piecewise linearization technique is used to reformulate the nonlinear program to obtain a global optimum. Numerical examples illustrate that the proposed method is computationally more efficient than current methods.

*"A review of deterministic optimization methods in engineering and management"* by M.-H. Lin et al. introduces recent advances in deterministic methods for solving signomial programming problems and mixed-integer nonlinear programming problems. A number of important applications in engineering and management are also reviewed to reveal the usefulness of the optimization methods.

### *Heuristic Algorithms*

The paper “*Opposition-based barebones particle swarm for constrained nonlinear optimization problems*” by H. Wang presents a modified barebones particle swarm optimization to solve constrained nonlinear optimization problems and simulation results show that the presented approach achieves a promising performance.

In “*Multithreshold segmentation based on artificial Immune systems*,” E. Cuevas et al. present an algorithm for multi-threshold segmentation which is based on the artificial immune systems technique. The clonal selection algorithm-based method shows a fast convergence and a low sensitivity to initial conditions and improves complex time-consuming computations commonly required by gradient-based methods.

In “*New bounds for ternary covering arrays using a parallel simulated annealing*” by H. Avila-George et al. three parallel approaches for simulated annealing: the independent, semi-independent, and cooperative searches are applied to the covering array construction problem. The empirical evidences indicate that cooperative approach offers the best execution times and the same bounds compared to the independent and semi-independent approaches.

In “*Applying hierarchical bayesian neural network in failure time prediction*,” L.-J. Kao and H.-F. Chen apply the Hierarchical Bayesian neural network (HBNN) approach to predict the failure time and utilize the Gibbs sampler of Markov Chain Monte Carlo to estimate statistical model parameters. The results of sensitivity analysis show that HBNN can provide not only the predictive distribution but also the heterogeneous parameter estimates for each path.

In “*Applying neural networks to prices prediction of crude oil futures*,” J. Wei-Shan Hu et al. attempt to accurately forecast prices of crude oil futures by adopting three popular neural networks methods including the multilayer perceptron, the Elman recurrent neural network, and recurrent fuzzy neural network (RFNN). Experimental results indicate that the use of neural networks to forecast the crude oil futures prices is appropriate and consistent learning is achieved by employing different training times. Moreover, the RFNN outperforms the other two neural networks in forecasting crude oil futures prices.

The paper “*Solving the tractor and semi-trailer routing problem based on a heuristic approach*” by H. Li et al. develops a heuristic algorithm to solve the tractor and semi-trailer routing problem (TSRP). The computational study shows that their method takes relatively little time to obtain satisfactory solutions.

L. Yao and J. Xu tackle a stone resource assignment problem with the aim of decreasing dust and waste water emissions. In their paper “*A stone resource assignment model under the fuzzy environment*” a bilevel multiobjective optimization model with fuzzy coefficients is constructed. They design a fuzzy simulation-based improved simulated annealing algorithm (FS-ISA) to search for Pareto optimal solutions. They also present a case study to demonstrate the practicality and efficiency of their model.

“*A modified PSO algorithm for minimizing the total costs of resources in MRCSP*” by M. Khalilzadeh et al. introduces a multimode resource-constrained project scheduling problem with finish-to-start precedence relations among project activities, considering renewable and nonrenewable resource costs. The authors formulate this problem as a mixed-integer programming model and present a metaheuristic algorithm based on a modified particle swarm optimization approach. Experimental results reveal the effectiveness and efficiency of their algorithm.

The paper “*A nonlinear multiobjective bilevel model for minimum cost network flow problem in a large-scale construction project*” by J. Xu et al. deals with a minimum cost network flow problem in a large-scale construction project using a nonlinear multiobjective bilevel model

with birandom variables. Results and analysis are presented to highlight the performances of the proposed method which is more effective and efficient compared to a genetic algorithm and a simulated annealing algorithm.

*"Sparse signal recovery via ECME thresholding pursuits"* by H. Song and G. Wang develops ECME thresholding pursuits (EMTPs) for sparse signal recovery. Two effective support detection strategies (hard thresholding and dynamic thresholding) are devised for the sparse signals with components having a fast decaying distribution of nonzero components. The experimental studies are presented to demonstrate that EMTP offers an appealing alternative to state-of-the-art algorithms for sparse signal recovery.

### *Hybrid Methods*

The paper *"Solving constrained global optimization problems by using hybrid evolutionary computing and artificial life approaches"* by J.-Y. Wu presents a hybrid real-coded genetic algorithm with a particle swarm optimization (PSO) algorithm and a hybrid artificial immune algorithm with a PSO algorithm for solving 13 constrained global optimization problems, including 6 nonlinear programming and 6 generalized polynomial programming optimization problems. Experimental results indicate that the proposed algorithms converge to a global optimum solution.

In *"Combining diffusion and grey models based on evolutionary optimization algorithms to forecast motherboard shipments,"* F.-K. Wang et al. develop a combined model based on the rolling Grey model and the Bass diffusion model to forecast motherboard shipments. The results indicate that the proposed model using a hybrid algorithm outperforms other methods for the fitting and forecasting processes in terms of mean absolute percentage error.

The paper *"Hybrid optimization approach for the design of mechanisms using a new error estimator"* by A. Sedano et al. proposes a hybrid optimization algorithm that combines the advantages of both stochastic and deterministic approaches to the design of mechanisms. Two engineering applications, a four-bar linkage and an injection machine, are presented to demonstrate the accuracy, robustness and efficiency of their proposed approach.

*"A Hybrid genetic algorithm for the multiple crossdocks problem"* by Z. Miao et al. considers multiple crossdocks problem through fixed transportation schedules with time windows, capacity, and penalty. The authors prove that the problem is NP-hard in the strong sense and develop a hybrid genetic algorithm integrating greedy technique and variable neighborhood search method to solve the problem effectively and efficiently.

*"A new hybrid Nelder-Mead Particle swarm optimization for Coordination optimization of directional overcurrent relays"* by A. Liu and M.-T. Yang proposes a new hybrid Nelder-Mead simplex search method and particle swarm optimization (proposed NM-PSO) algorithm to solve the directional overcurrent relays coordination optimization problem. The findings demonstrate that the performance of the proposed NM-PSO is better than that of PSO and original NM-PSO in terms of computation speed, rate of convergence, and feasibility.

*"A VNS metaheuristic with stochastic steps for Max 3-cut and Max 3-section"* by A.-f. Ling proposes a local search algorithm and a variable neighborhood global search algorithm with two stochastic search steps to obtain the global solution by establishing a neighborhood structure of the Max 3-cut problem. Numerical results show that the proposed heuristic algorithm can obtain efficiently the high-quality solutions and has the better numerical performance than 0.836-approximate algorithm for the NP-Hard Max 3-cut and Max 3-section problems.

The paper “*Dynamic programming and heuristic for stochastic uncapacitated lot-sizing problems with incremental quantity discount*” by Y. Zhang et al. considers the stochastic uncapacitated lot-sizing problems with incremental quantity discount where the uncertain parameters are supposed to evolve as discrete time stochastic processes. The obtained results reveal that the presented algorithm outperforms the commercial solver CPLEX and other heuristics in both quality of solution and run time.

“*A hybrid algorithm based on ACO and PSO for capacitated vehicle routing problems*” by Y. Kao et al. proposes a new hybrid algorithm based on two main swarm intelligence approaches, ant colony optimization (ACO) and particle swarm optimization (PSO), for solving capacitated vehicle routing problems. Computational results show that the performance of the proposed method is competitive in terms of solution quality when compared with existing ACO-and PSO-based approaches.

### *Challenging Applications*

The paper “*Goal-programming-driven genetic algorithm model for wireless access point deployment optimization*” by C.-S. Wang and C.-T. Chang proposes a method that integrates a goal-programming-driven model and a genetic algorithm to resolve the multiple objectives appropriate wireless access point deployment problem. Three experiment results demonstrate the utility and stability of the proposed method.

S.-Y. Lin et al. paper “*A cost-effective planning graph approach for large-scale web service composition*” proposes a novel cost-effective Web service composition mechanism, utilizes planning graph based on backward search algorithm to find multiple feasible solutions, and recommends a best composition solution according to the lowest service cost.

In “*A fuzzy dropper for proportional loss rate differentiation under wireless network with a multi-state channel*,” Y.-C. Szu proposes a novel packet dropper for fuzzy controlling the proportional loss rate differentiation in a wireless network with multi-state channel. Simulation results reveal that the fuzzy proportional loss rate dropper does achieve accurate loss rate proportion, lower queuing delay and loss rate, and higher throughput, compared with other methods in the wireless environment.

The paper “*Evaluating the performance of taiwan homestay using analytic network process*” by Y.-C. Hu and J.-H. Wang develops and constructs a set of evaluation indicators tailor-made for homestay sector through discussion of literatures and interviewing experts so that the evaluation framework would be more comprehensive and more practical.

The paper “*Improved degree search algorithms in unstructured P2P networks*” by G. Liu et al. proposes two memory function degree search algorithms: memory function maximum degree algorithm and memory function preference degree algorithm and studies their performance including the search success rate and the search message quantity in scale-free networks, random graph networks, and small-world networks.

The paper “*Variable neighborhood search for parallel machines scheduling problem with step-deteriorating jobs*” by W. Cheng et al. studies a scheduling problem of minimizing the total completion time on identical parallel machines where the processing time of a job is a step function of its starting time and a deteriorating date that is individual to all jobs. The computational results show that the proposed approaches obtain near-optimal solutions in a reasonable computational time even for large-sized problems.

The paper “*Solving packing problems by a distributed global optimization algorithm*” by N.-Z. Hu et al. develops a novel method to convert the nonlinear objective function in a packing program into an increasing function with a single variable and two fixed parameters.

The transformed linear program is then decomposed into several subproblems by specifying various parameter values, which is solvable simultaneously by a distributed computation algorithm to obtain a global optimum.

The paper "*A hybrid network model to extract key criteria and its application for brand equity evaluation*" by C.-Y. Chen and C.-W. Li develops a hybrid model based on social network analysis and maximum mean deentropy algorithms for extracting evaluation criteria which considers substitutions between the criteria. The effectiveness and feasibility of the hybrid model are demonstrated by examples of evaluating brand equity.

The paper "*Mixed mortar element method for  $P_1^{NC}/P_0$  element and its multigrid method for the incompressible Stokes problem*" by Y. Jiang and J. Chen discusses a mortar-type  $P_1^{NC}/P_0$  element method for the incompressible Stokes problem. The inf-sup condition and the optimal error estimate are proved. The study also proposes a W-cycle multigrid for solving the discrete problem and demonstrates the optimal convergence of the multigrid method.

In "*Quality improvement and robust design methods to a pharmaceutical research and development*" by B. R. Cho and S. Shin, new robust design experimental and optimization models for time-oriented data are applied to the pharmaceutical production research and development. Compared to the traditional Taguchi optimization model, the proposed experimental methodology is particularly useful for experiments with time-oriented pharmaceutical characteristics.

The paper "*A novel method for technology forecasting and developing R&D strategy of building integrated photovoltaic technology industry*" by Y.-J. Chiu and T.-M. Ying proposes a hybrid approach to explore the life cycle of building-integrated photovoltaic (BIPV) technology and develop the R&D strategy of related industries. The proposed approach comprises patent analysis, logistic growth model analysis, and patent matrix map analysis. The authors also provide three-dimensional matrix of degree of protection, R&D capability, and benefit creation to select R&D strategies for BIPV industry.

J.-H. Lin et al. propose a systematic method to analyze student recruitment numbers for future needs, based on the concept of material resource planning (MRP). In their paper "*The number of students needed for undeclared programs at a college from the supply-chain viewpoint*," the relationship between a curricular structure tree and the associated commonalities is studied and a quantified model of commonality and recruitment planning for appropriate curriculum design is proposed. The authors use two simple examples to illustrate the implementation of MRP in analysis of the replenishment levels in an education system.

The paper "*A selection approach for optimized problem-solving process by grey relational utility model and multi-criteria decision analysis*" by C.-K. Ke and M.-Y. Wu proposes an approach to assist workers in determining the optimal selection order of candidate actions based on a Grey relational utility model and a multi-criteria decision analysis. Experimental results from analyzing a high-tech company's knowledge base log demonstrate that their selection approach is effective.

M.-H. Lin and H.-J. Hsu addresses an incentive pricing problem for relaying services in multi-hop cellular networks. In their paper "*Optimal incentive pricing on relaying services for maximizing connection availability in multi-hop cellular networks*" a mathematical programming model is constructed and then solved to determine an optimal incentive price for each intermediate node providing relaying services. The computational results demonstrate that their proposed approach maximizes connection availability of the networks compared to fixed-rate or location-based methods.

*"A two stage dea to analyze the effect of entrance deregulation on iranian insurers: a robust approach"* by S. G. J. Naini et al. analyzes technical efficiency for Iranian insurance companies

between 2003 and 2010, a period that insurers experienced intense volatility due to the entrance deregulation of the market by two-stage data envelopment analysis. The major results show that ownership type and failure to meet the risk management rules are the main drivers of efficiency.

*"A label correcting algorithm for partial disassembly sequences in the production planning for end-of-life products"* by P.-F. Tsai investigates a single period partial disassembly optimization problem to generate an optimal disassembly sequence in product recovery of the end-of-life products. The study presents a heuristic procedure that utilizes a label correcting algorithm which is a polynomial-time algorithm to find a solution. Numerical examples are also used to demonstrate the effectiveness of the proposed solution procedure.

Finally, the guest editors would like to thank the authors for their contributions, the referees for their time and energy in reviewing the paper, and all the staff of Hindawi involved in the preparation of this special issue for their support to publish the articles related to this subject. Without their cooperation, it would have not been possible to edit this special issue.

*Jung-Fa Tsai  
John Gunnar Carlsson  
Dongdong Ge  
Yi-Chung Hu  
Jianming Shi*

*Research Article*

## A Stone Resource Assignment Model under the Fuzzy Environment

**Liming Yao, Jiuping Xu, and Feng Guo**

*Uncertainty Decision-Making Laboratory, Sichuan University, Chengdu 610064, China*

Correspondence should be addressed to Jiuping Xu, [xujiuping@scu.edu.cn](mailto:xujiuping@scu.edu.cn)

Received 1 January 2012; Revised 12 April 2012; Accepted 7 May 2012

Academic Editor: Jianming Shi

Copyright © 2012 Liming Yao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a bilevel multiobjective optimization model with fuzzy coefficients to tackle a stone resource assignment problem with the aim of decreasing dust and waste water emissions. On the upper level, the local government wants to assign a reasonable exploitation amount to each stone plant so as to minimize total emissions and maximize employment and economic profit. On the lower level, stone plants must reasonably assign stone resources to produce different stone products under the exploitation constraint. To deal with inherent uncertainties, the object functions and constraints are defuzzified using a possibility measure. A fuzzy simulation-based improved simulated annealing algorithm (FS-ISA) is designed to search for the Pareto optimal solutions. Finally, a case study is presented to demonstrate the practicality and efficiency of the model. Results and a comparison analysis are presented to highlight the performance of the optimization method, which proves to be very efficient compared with other algorithms.

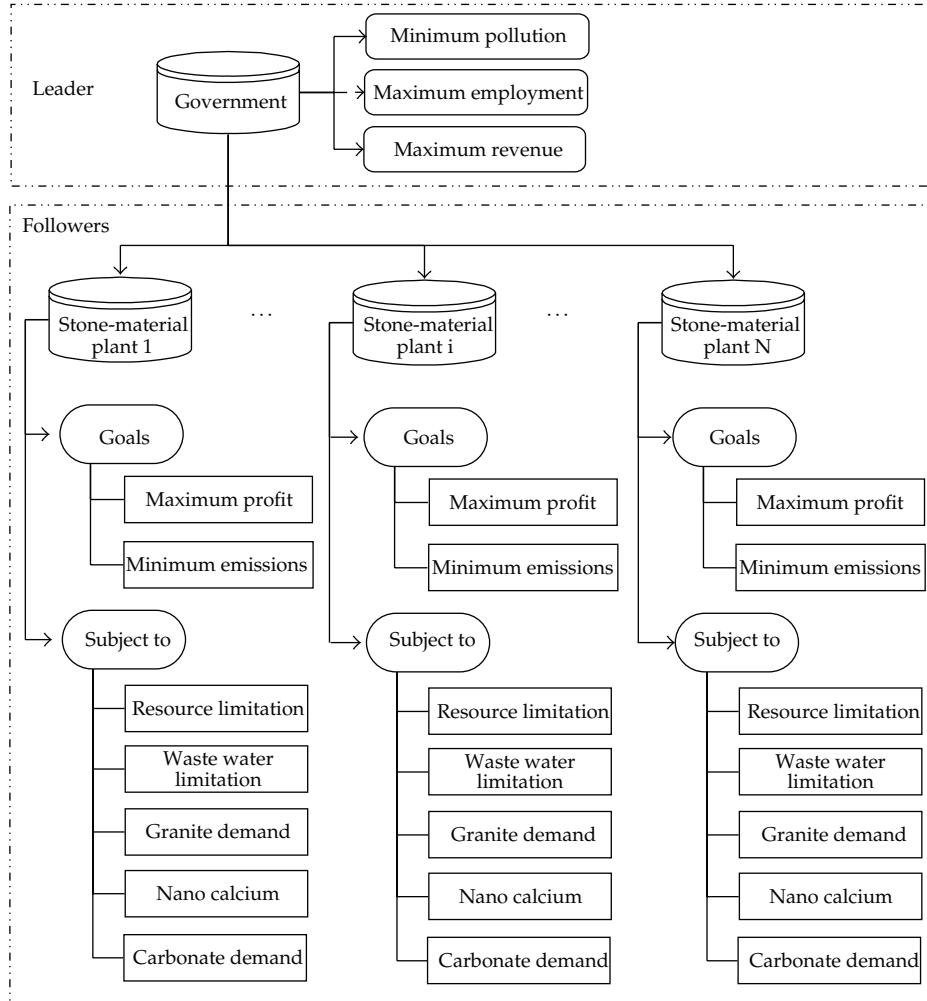
### 1. Introduction

The dust and the waste water from the stone industry can cause serious damage to the regional ecological environment. The overexploitation and the stone processing have resulted in the vegetation decrement, and the pollution of air and water in those areas with rich stone resources. The annual amount of waste generated include 700,000 tons of slurry waste as well as 1 million tons of solid waste. The consequent dumping of this waste in open areas has created several environmental problems and has negatively impacted agriculture, local inhabitants, and groundwater [1]. Therefore, it is urgent to normalize the quarrying and processing of the stone resource. Some technologies are introduced to save energy and reduce the emission in the stone industry by many scholars [2, 3]. Some other scholars [4–6] considered the use of the marble powder to reduce the waste, but few literatures discussed the quantitative relationship between the emission and the exploiting

and processing amount. In fact, a reasonable assignment of stone resources could significantly reduce the emissions. This paper considers the government as the upper level and the stone plant as the lower level to develop a bi-level model. For the stone industry, the objectives of the government authority are to minimize environmental pollution and maximize social employment and economic revenue. This can be achieved by optimizing the amount of stone extracted and exploited between the participating plants, which are assumed to cooperate and act as a lower-level decision maker. Noting that industrial symbiosis implicitly requires the cooperative behavior of the participants [7, 8], the government can influence the plants by imposing disincentives by assigning different amounts to stone plants according to their production scale and clean technology level. The plants operate independently of each other. Each plant has its own goals, which are to maximize the profit from the sale of nano calcium carbonate, marble products, granite slabs, and man-made slabs and to minimize the emissions of stone dust and waste water.

To develop the bi-level optimization model for assigning the stone resources, some emission coefficients have to be effectively estimated. It is usually difficult to collect the exact data of emissions of stone dust and waste water when exploiting the stone mine and processing stone products. The fuzzy number is an efficient tool to describe the variables without crisp information. The membership function of fuzzy sets can be used to describe the possibility that emission coefficients take the value according to the experience of those people in the stone industry. Actually, there has been some studies describing the uncertainty by fuzzy sets. For example, Petrovic et al. [9] used fuzzy sets to describe the customer demand, supply deliveries along the SC and the external or market supply, and develop a supply chain model with fuzzy coefficients. Lee and Yao [10] fuzzify the demand quantity and the production quantity per day to solve the economic production quantity. These studies inspire us to use the fuzzy sets to interpret the vague and imprecise about the emissions of stone dust and waste water. For the fuzzy bi-level optimization problem, a satisfactory (near-optimal or “satisficing”) solution can be reached by providing tolerances in the objective functions and constraints and by defining corresponding degrees of satisfaction through membership functions to indicate the preference of the decision makers which is typical of decision making in a fuzzy environment [11]. The followers then communicate their results to the leader, who modifies his goals and control variables if the original tolerances are not met. The process continues iteratively until a solution which satisfies the goals of both leader and follower is reached.

A bi-level multiobjective model with fuzzy coefficients is always an NP hard problem, and it is especially difficult for nonlinear bi-level programming under a fuzzy environment to find a numerical solution. Some existing methods mainly focus on metaheuristics which include the genetic algorithm [12], the simulated annealing [13], and the hybrid tabu-ascent algorithm [14]. However, as these need to be designed for single-objective problems with crisp coefficients, it is difficult to find a usual or normal pattern for a bi-level model with fuzzy coefficients. This paper proposes an improved simulated annealing based on a fuzzy simulation to search for a Pareto optimal solution after a possibilistic check. The following sections of this paper are organized as follows. In Section 2, the reason a bi-level multi-objective model is used to optimize the stone industry is explained. The process of data fuzzification is introduced in detail. A possibilistic bi-level multi-objective programming model is developed. In Section 3, a fuzzy simulation-based improved simulated algorithm is proposed to solve the bi-level multi-objective programming model with fuzzy coefficients. In Section 4, a practical case is presented to show the significance of the proposed models and algorithms. Finally, conclusions are given in Section 5.



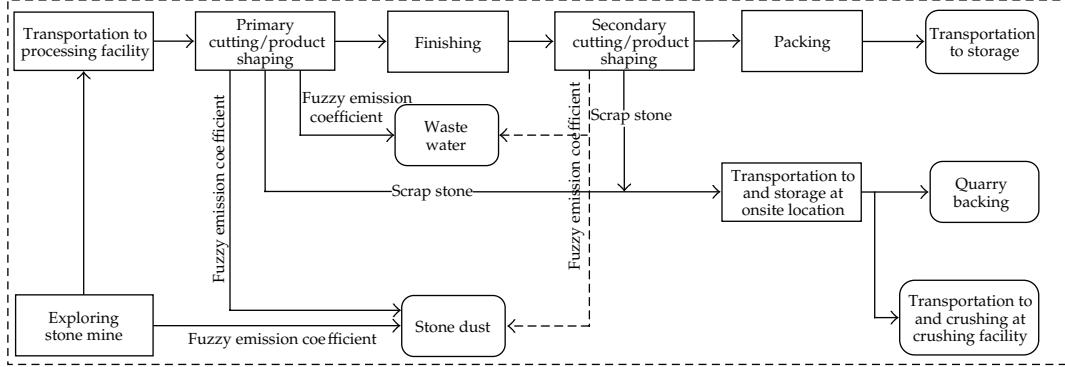
**Figure 1:** Stone industry decision making hierarchy.

## 2. Mathematical Modelling

In order to develop the mathematical model, some basic background and descriptions are introduced.

### 2.1. Key Problems Description

For the stone industry, local government and stone plants play important roles to perform the responsibilities, respectively. Government has the authority to decide the amount that should be exploited and then needs to make a sustainable plan to avoid overexploitation and pollution. On the other hand, stone plants need to make the production plan according to the stone quota that government gives. As shown in Figure 1, the local government has environmental protection and maximum employment as its most important goals and then considers the revenue. On the lower level, the stone-material plants usually consider



**Figure 2:** Process flow diagram for granite processing operations.

economic profit as their first goal. Due to a limitation on the amount that can be extracted and the environmental protection requirements, they also have to consider minimizing emissions. At the same time under a government policy and according to capacity, they also need to think about employment. In addition, it is necessary to increase the investment to improve emission reduction capacity to satisfy sustainable development requirements. Considering the above, the problem should be regarded as a bi-level optimization model in which the government authority is the upper-level decision maker and the stone plants are the lower-level decision makers. It is assumed that there is a perfect exchange of information between all the participants such that the objectives and constraints are known.

As shown in Figure 2, the granite is first exploited from the stone mine and cut into the primary products such as granite slabs and man-made slabs; then, these are processed into the floor or other products. Scrap materials are usually processed into fine powder calcium carbonate and nano calcium carbonate to meet market demand. During the complete process, a great deal of stone dust and waste water are produced. Since it is technically difficult to collect the exact data of emissions, we usually make a rough estimation by the difference of weight before and after exploiting and processing and then look for the possibility for every weight by the professional advices in the stone industry. Therefore, the fuzzy number is an efficient tool to describe this situation by its membership function. Actually, the fuzzy environment has been successfully studied and applied in many areas, such as flow shop scheduling problem [17], supply chain problem [18], and assignment problem [19]. These studies show the necessity of considering fuzzy environment in practical problems. It is also the motivation for considering fuzzy environment in the stone resources assignment problems.

## 2.2. Assumptions and Notations

Before the data fuzzification and developing the optimization model, some assumptions should be introduced.

- (1) Emission of stone dust and waste water is proportional to the amount of stone processed into products.
- (2) Employment level is also proportional to the amount of stone processed into products.

- (3) The constant cost of product  $j$  only exists when the stone-material plant produces product  $j$ .
- (4) Since the government endows different subsidies to plants, it is assumed that each plant has its own tax rate  $S_i$  and the tax is proportional to the turnover of all stone products.

The notations are used to describe the subsidy model in the investigation are referred to in the Abbreviations Section.

### 2.3. Data Fuzzification Based on Crossover Validation Test

Often there is little historical data to describe emission reduction due to the raw development of the last decade. For example, some research considers the transport cost as uncertain coefficients because of the changing weather and the unpredictable road condition [20, 21]. In this paper, the emission coefficients cannot be estimated using the statistical methods and then have to fuzzified according to those insufficient data.

The essence of fuzzification is to find an approximate membership function to describe the fuzzy number [22]. Many scholars have described some methods of determining the membership functions that are essentially based on direct methods of inquiry made on human beings and corrected using indirect methods [23, 24]. Some other scholars propose the automatic methods to determine the membership functions when no expert is available or in the case when there are so many data [25]. In the present paper, we will propose the fuzzification methods by combining the 5-parameter membership function and crossover validation test. Taking the stone dust emission coefficients  $\widetilde{Ed}$  as an example, the process for fuzzifying can be summarized as follows.

*Step 1.* Split the data set  $S$  of stone dust emission coefficients  $\widetilde{Ed}$  into a training set  $S_{tr}$  and a validation set  $S_v$ .

*Step 2.* Find the smallest, middle, and largest data in  $S$ ; denote them  $Ed_s$ ,  $Ed_m$ , and  $Ed_l$ , respectively.

*Step 3.* Compute the left and right slopes for the data in  $S_{tr}$  by the following equations; respectively,

$$Ed_\alpha(x) = \begin{cases} \frac{Ed_s + Ed_m - 2x}{Ed_s + Ed_m}, & \text{if } Ed_s \leq x \leq \frac{Ed_s + Ed_m}{2}, \\ \frac{2x - (Ed_s + Ed_m)}{2Ed_m}, & \text{if } \frac{Ed_s + Ed_m}{2} \leq x \leq Ed_m, \end{cases} \quad (2.1)$$

$$Ed_\beta(x) = \begin{cases} \frac{Ed_l + Ed_m - 2x}{Ed_l + Ed_m}, & \text{if } Ed_s \leq x \leq \frac{Ed_l + Ed_m}{2}, \\ \frac{2x - (Ed_l + Ed_m)}{2Ed_m}, & \text{if } \frac{Ed_l + Ed_m}{2} \leq x \leq Ed_m. \end{cases} \quad (2.2)$$

Then we get the set of left slopes  $Ed^L = \{\alpha \mid Ed_\alpha(x), x \in S_{tr}, x \leq Ed_m\}$  and the set of left slopes  $Ed^R = \{\beta \mid Ed_\beta(x), x \in S_{tr}, x \geq Ed_m\}$ .

*Step 4.* Define the membership function as follows:

$$\mu_{\widetilde{Ed}}(x) = \begin{cases} 0, & \text{if } x < Ed_s, \\ 2^{\alpha-1} \left( \frac{x - Ed_s}{Ed_m - Ed_s} \right)^\alpha, & \text{if } Ed_s \leq x \leq \frac{Ed_s + Ed_m}{2}, \\ 1 - 2^{\alpha-1} \left( \frac{Ed_m - x}{Ed_m - Ed_s} \right)^\alpha, & \text{if } \frac{Ed_s + Ed_m}{2} < x < Ed_m, \\ 1 - 2^{\beta-1} \left( \frac{x - Ed_m}{Ed_l - Ed_m} \right)^\beta, & \text{if } Ed_m \leq x < \frac{Ed_l + Ed_m}{2}, \\ 2^{\beta-1} \left( \frac{Ed_l - x}{Ed_l - Ed_m} \right)^\beta, & \text{if } \frac{Ed_l + Ed_m}{2} \leq x \leq Ed_l, \\ 0, & \text{if } x > Ed_l, \end{cases} \quad (2.3)$$

where  $\alpha \in Ed^L$  and  $\beta \in Ed^R$ .

*Step 5.* Take all the data in  $S_v$  in the above equation and compute the membership  $\mu_{\widetilde{Ed}}(x; \alpha, \beta)$ , where  $x \in S_v$ ,  $\alpha \in Ed^L$  and  $\beta \in Ed^R$ .

*Step 6.* Carry out the crossover validation test proposed by Kohavi [26]. Compute the memberships of  $x_i \in S_v$  for any combination  $(\alpha, \beta) \in (Ed^L, Ed^R)$ . Then compute the percentage of correct results by the following equation:

$$PCC_v = 100 \times \frac{1}{NV} \sum_{(x_i, \mu_{\widetilde{Ed}}(x_i; \alpha, \beta)) \in S_v} \delta(x_i, \mu_{\widetilde{Ed}}(x_i; \alpha, \beta)), \quad (2.4)$$

where  $PCC_v$  denotes the percentage of correct results over the validation set  $S_v$ ,  $NV$  is the number of data points in validation set  $S_v$ , and  $\delta(\mu_{\widetilde{Ed}}(x_i; \alpha_1, \beta_1), \mu_{\widetilde{Ed}}(x_i; \alpha_2, \beta_2)) = 1$  if  $\mu_{\widetilde{Ed}}(x_i; \alpha_1, \beta_1) = \mu_{\widetilde{Ed}}(x_i; \alpha_2, \beta_2)$ , while  $\delta(\mu_{\widetilde{Ed}}(x_i; \alpha_1, \beta_1), \mu_{\widetilde{Ed}}(x_i; \alpha_2, \beta_2)) = 0$  if  $\mu_{\widetilde{Ed}}(x_i; \alpha_1, \beta_1) \neq \mu_{\widetilde{Ed}}(x_i; \alpha_2, \beta_2)$ .

*Step 7.* Find the combination  $(\alpha, \beta)$  by which the largest percentage of correct results can be obtained when carrying out the crossover validation test with each other. Then we get the membership function.

## 2.4. Model Formulation

The bi-level multiobjective optimization model under a fuzzy environment for assigning stone resources can be mathematically formulated as follows.

#### 2.4.1. Government Model (Upper Level)

As the upper level, the government has the obligation to protect the local environment, solve employment issues, and promote economic revenue. Generally, the following goals are usually considered by the government.

To achieve minimum emissions, including the stone dust ( $\sum_{i=1}^m \widetilde{E}d_i Y_i$ ) when all plants exploit the stone mine, the stone dust ( $\sum_{i=1}^m \sum_{j=1}^n \widetilde{e}d_{ij} X_{ij}$ ) when plants produce stone products, and total waste water ( $\sum_{i=1}^m \sum_{j=1}^n \widetilde{ew}_{ij} X_{ij}$ ) when all the plants exploit the stone mine and produce those stone products is the first objective. Since  $\widetilde{E}d_i$ ,  $\widetilde{e}d_{ij}$ , and  $\widetilde{ew}_{ij}$  are all fuzzy numbers which are obtained by fuzzification due to insufficient historical data, it is usually difficult to derive precise minimum emissions, and decision makers only require a minimum objective ( $\bar{F}_1$ ) under some possibilistic level ( $\delta_1^U$ ) [27]. Hence the following possibilistic objective function and constraint are derived:

$$\min \bar{F}_1 \quad (2.5)$$

subject to (s.t.)

$$\text{Pos} \left\{ \sum_{i=1}^m \widetilde{E}d_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (\widetilde{e}d_{ij} X_{ij} + \widetilde{ew}_{ij} X_{ij}) \leq \bar{F}_1 \right\} \geq \delta_1^U, \quad (2.6)$$

where Pos is the possibility measure proposed by Dubois and Prade [28] and  $\delta_1^U$  is the possibilistic level representing the possibility that decision makers achieve the minimum objective. All fuzzy arithmetic in (2.6) and the following equations come from the operation proposed by Kaufmann and Gupta [29].

To achieve maximum employment  $F_2$  which consisted of constant workers ( $P_i$ ) and variable workers ( $p_{ij} X_{ij}$ ), the following objective function is obtained:

$$\max F_2 = \sum_{i=1}^m \left( \sum_{j=1}^n p_{ij} X_{ij} + P_i \right). \quad (2.7)$$

To achieve the maximum economic output which can be obtained by multiplying unit amount ( $c_j$ ), conversion rate ( $\theta_{ij}$ ), and amount of stone ( $X_{ij}$ ), the following objective function is obtained:

$$\max F_3 = \sum_{i=1}^m S_i \left( \sum_{j=1}^n c_j \theta_{ij} X_{ij} \right). \quad (2.8)$$

Generally, some mandatory conditions must be satisfied when the government makes a decision. These are listed as follows.

The total exploration quantity ( $\sum_{i=1}^m Y_i$ ) cannot exceed the upper limitation ( $R^U$ ) of the total stone resources in the region:

$$\sum_{i=1}^m Y_i \leq R^U. \quad (2.9)$$

The stone dust from exploiting ( $\sum_{i=1}^m \widetilde{Ed}_i Y_i$ ) and producing ( $\sum_{i=1}^m \sum_{j=1}^n \widetilde{ed}_{ij} X_{ij}$ ) and the waste water ( $\sum_{i=1}^m \sum_{j=1}^n \widetilde{ew}_{ij} X_{ij}$ ) should be less than the predetermined levels ( $ED^U$  and  $EW^U$ ) in order to guarantee air and water quality. Two constraints are derived under the possibilistic levels ( $\delta_2^U$  and  $\delta_3^U$ ):

$$\text{Pos} \left\{ \sum_{i=1}^m \widetilde{Ed}_i Y_i + \sum_{i=1}^m \sum_{j=1}^n \widetilde{ed}_{ij} X_{ij} \leq ED^U \right\} \geq \delta_2^U, \quad (2.10)$$

$$\text{Pos} \left\{ \sum_{i=1}^m \sum_{j=1}^n \widetilde{ew}_{ij} X_{ij} \leq EW^U \right\} \geq \delta_3^U. \quad (2.11)$$

The output of some products ( $\sum_{i=1}^m \theta_{ij} X_{ij}$ ) should meet the market demand ( $D_j^L$ ). For example, the nano calcium carbonate is very popular in many areas, so the stone plants should provide enough output to meet the demand:

$$\sum_{i=1}^m \theta_{ij} X_{ij} \geq D_j^L \quad \forall j. \quad (2.12)$$

#### 2.4.2. Plant Model (Lower Level)

On the lower level, the stone plants usually pursue maximum profit and then try to reduce the emissions. Thus, the following two objectives are introduced.

Each plant wishes to achieve maximum profit which consisted of total sales ( $\sum_{j=1}^n c_j \theta_{ij} X_{ij}$ ) minus the production cost ( $f(X_{ij})$ ) and the inventory cost ( $h_i(Y_i - \sum_{j=1}^n X_{ij})$ ); then the following objective function is determined:

$$\max H_i^1 = \sum_{j=1}^n c_j \theta_{ij} X_{ij} - \sum_{j=1}^n f(X_{ij}) - h_i \left( Y_i - \sum_{j=1}^n X_{ij} \right), \quad (2.13)$$

where  $f(X_{ij})$  is the production-cost function as follows [27]:

$$f(X_{ij}) = \begin{cases} t_{ij} X_{ij} + C_{ij}, & \text{if } X_{ij} > 0, \\ 0, & \text{if } X_{ij} = 0. \end{cases} \quad (2.14)$$

Every plant also wishes to achieve minimum emissions. However, since the emissions  $\widetilde{ed}_{ij}$  and  $\widetilde{ew}_{ij}$  are fuzzy numbers, it is usually difficult to determine the precise minimum

emissions, and decision makers only require a minimum objective ( $\overline{H}_i^2$ ) under some probabilistic level ( $\sigma_i^L$ ). Hence, the probabilistic constraint is as follows:

$$\min \overline{H}_i^2 \quad (2.15)$$

subject to

$$\text{Pos} \left\{ \sum_{i=1}^m \sum_{j=1}^n (\tilde{e}d_{ij}X_{ij} + \tilde{e}\tilde{w}_{ij}X_{ij}) \leq \overline{H}_i^2 \right\} \geq \sigma_i^L, \quad (2.16)$$

where  $\sigma_i^L$  is the probabilistic level under which decision makers require the minimum objective.

Since production in all the plants is influenced by government policy and market demand, there are some conditions that need to be satisfied.

The amount used for production ( $\sum_{j=1}^n X_{ij}$ ) should not exceed the total limitation ( $Y_i$ ):

$$\sum_{j=1}^n X_{ij} \leq Y_i. \quad (2.17)$$

The inventory amount ( $Y_i - \sum_{j=1}^n X_{ij}$ ) should not exceed the maximum limitation ( $IV_i^U$ ):

$$Y_i - \sum_{j=1}^n X_{ij} \leq IV_i^U. \quad (2.18)$$

The production cost which consisted of two parts including product cost ( $\sum_{j=1}^n f(X_{ij})$ ) and total inventory cost ( $\sum_{j=1}^n h_i(Y_i - \sum_{j=1}^n X_{ij})$ ) should not exceed the predetermined level ( $PC_i^U$ ):

$$\sum_{j=1}^n f(X_{ij}) + h_i \left( Y_i - \sum_{j=1}^n X_{ij} \right) \leq PC_i^U. \quad (2.19)$$

Some products ( $\theta_{ij}X_{ij}$ ) should not be less than the lowest production level ( $P_{ij}^L$ ) in plant  $i$ :

$$\theta_{ij}X_{ij} \geq P_{ij}^L. \quad (2.20)$$

#### 2.4.3. Bilevel Model

In such a complicated system, both the leader and the followers should simultaneously consider the objectives and constraints and then make the decision. Therefore, from

(2.5)~(2.20), the complete bi-level multiobjective optimization model under a fuzzy environment is as follows:

$$\left\{
 \begin{array}{l}
 \min \bar{F}_1 \\
 \max F_2 = \sum_{i=1}^m \left( \sum_{j=1}^n p_{ij} X_{ij} + P_i \right) \\
 \max F_3 = \sum_{i=1}^m S_i \left( \sum_{j=1}^n c_j \theta_{ij} X_{ij} \right) \\
 \text{s.t. } \left\{ \begin{array}{l}
 \text{Pos} \left\{ \sum_{i=1}^m \widetilde{E}d_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (\widetilde{e}d_{ij} X_{ij} + \widetilde{e}w_{ij} X_{ij}) \leq \bar{F}_1 \right\} \geq \delta_1^U \\
 \text{Pos} \left\{ \sum_{i=1}^m \widetilde{E}d_i Y_i + \sum_{i=1}^m \sum_{j=1}^n \widetilde{e}d_{ij} X_{ij} \leq ED^U \right\} \geq \delta_2^U \\
 \text{Pos} \left\{ \sum_{i=1}^m \sum_{j=1}^n \widetilde{e}w_{ij} X_{ij} \leq EW^U \right\} \geq \delta_3^U \\
 \sum_{i=1}^m \theta_{ij} X_{ij} \geq D_j^L \quad \forall j \\
 \max H_i^1 = \sum_{j=1}^n c_j \theta_{ij} X_{ij} - \sum_{j=1}^n f(X_{ij}) - h_i \left( Y_i - \sum_{j=1}^n X_{ij} \right) \\
 \min \bar{H}_i^2 \\
 \text{s.t. } \left\{ \begin{array}{l}
 \text{Pos} \left\{ \sum_{i=1}^m \sum_{j=1}^n (\widetilde{e}d_{ij} X_{ij} + \widetilde{e}w_{ij} X_{ij}) \leq \bar{H}_i^2 \right\} \geq \sigma_i^L \\
 \sum_{j=1}^n X_{ij} \leq Y_i \\
 Y_i - \sum_{j=1}^n X_{ij} \leq IV_i^U \\
 \sum_{j=1}^n f(X_{ij}) + h_i \left( Y_i - \sum_{j=1}^n X_{ij} \right) \leq PC_i^U \\
 \theta_{ij} X_{ij} \geq P_{ij}^L.
 \end{array} \right.
 \end{array}
 \right\} \quad (2.21)$$

### 3. Solution Approach

Generally, bi-level programming is an NP-hard problem, and it is difficult to determine an optimal solution [30–32]. In the proposed model, decision makers on the upper and lower levels have to face more than two conflicting objectives and then make a decision under a fuzzy environment. This significantly increases the difficulty of finding an optimal strategy for both the upper and lower levels. Therefore, the fuzzy simulation-based improved

simulated annealing (FS-ISA) is designed to solve the bi-level optimization model with fuzzy coefficients.

### 3.1. Fuzzy Simulation for Possibilistic Constraints

Fuzzy simulation is usually proposed to approximate the possibility measure according to the membership function of a fuzzy number [27]. Taking the constraint (2.5) as an example, we will introduce the key principle of the fuzzy simulation and find the minimum  $\bar{F}_1$  such that the constraint holds.

Let  $Y_i^*$  and  $X_{ij}^*$  be predetermined feasible solutions for  $i = 1, 2, \dots, m$ ,  $j = 1, 2, \dots, n$ , which will be regarded as input variables. Firstly, set  $\bar{F}_1 = M$ , where  $M$  is a sufficiently large number. Secondly, randomly generate  $\lambda_i$ ,  $\eta_{ij}$ , and  $\kappa_{ij}$  from the  $\delta_1^U$ -level set of the fuzzy numbers  $\bar{E}\bar{d}_i$ ,  $\bar{e}\bar{d}_{ij}$ , and  $\bar{e}\bar{w}_{ij}$ , respectively. Thirdly, compute the value  $\bar{f} = \sum_{i=1}^m \lambda_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (\eta_{ij} + \kappa_{ij}) X_{ij}$ . If  $\bar{F}_1 > \bar{f}$ , replace it with  $\bar{f}$ . Finally, repeat this process for  $N$  times. The value  $\bar{F}_1$  is regarded as the estimation. Then the simulation process can be summarized in Procedure 1.

Sometimes, we need to check whether a solution satisfies the possibilistic constraint. This means that we need to compute the possibility and compare it with the predetermined possibilistic level. Then another simulation is applied to check the constraint. Taking the constraint (2.10) as an example, we will introduce how to simulate the possibility  $L = \text{Pos}\{\sum_{i=1}^m \sum_{j=1}^n \bar{e}\bar{w}_{ij} X_{ij} \leq EW^U\}$ .

Let  $X_{ij}^*$  be predetermined solution for  $i = 1, 2, \dots, m$ ,  $j = 1, 2, \dots, n$ , which will be regarded as input variables. Give a lower estimation of the possibility  $L$ , denoted by  $\delta$ . Then we randomly generate  $\kappa_{ij}$  from the  $\delta$ -level set of the fuzzy numbers  $\bar{e}\bar{w}_{ij}$ . If the  $\delta$ -level set is not easy for a computer to describe, we can give a larger region, for example, a hypercube containing the  $\delta$ -level set. Certainly, the smaller the region, the more effective the fuzzy simulation. Now we set

$$\mu = \max \left\{ \mu_{\bar{e}\bar{w}_{ij}}, i = 1, 2, \dots, m, j = 1, 2, \dots, n \right\}. \quad (3.1)$$

If  $\sum_{i=1}^m \sum_{j=1}^n \kappa_{ij} X_{ij} \leq EW^U$  and  $L < \mu$ , then we set  $L = \mu$ . Repeat this process  $N$  times. The value  $L$  is regarded as an estimation of the possibility. Then the process for constrain check can be summarized in Procedure 2.

### 3.2. Fuzzy Simulation-Based Improved Simulated Annealing Algorithm

Simulated annealing algorithm (SA) is proposed for the problem of finding, numerically, a point of the global optimization of a function defined on a subset of a  $n$ -dimensional Euclidean space [33–35]. Many fruitful results are obtained in the past decades. Steel [36, 37] calls simulated annealing the most exciting algorithmic development of the decade. For the multiobjective optimization problems, some scholars have introduced many progressive simulated annealing algorithms to solve them. Especially, Suppapatnarm et al. [38] designed a simulated annealing algorithm along with archiving the Pareto optimal solutions coupled with return to base strategy (SMOSA) to explore the trade-off between multiple objectives in optimization problems. Suman and Kumar [39, 40] introduced four

**Input:** Decision variables  $Y_i$  and  $X_{ij}$   
**Output:** The minimum  $\bar{F}_1$   
**Step 1.** Set  $\bar{F}_1 = M$ , where  $M$  is sufficiently large number;  
**Step 2.** Randomly generate  $\lambda_i$ ,  $\eta_{ij}$  and  $\kappa_{ij}$  from the  $\delta_1^U$ -level set of the fuzzy numbers  $\widetilde{Ed}_i$ ,  $\widetilde{ed}_{ij}$  and  $\widetilde{ew}_{ij}$  respectively;  
**Step 3.** Compute  $\bar{f} = \sum_{i=1}^m \lambda_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (\eta_{ij} + \kappa_{ij}) X_{ij}$  and replace  $\bar{F}_1$  with  $\bar{f}$  provided that  $\bar{F}_1 > \bar{f}$ ;  
**Step 4.** Repeat the second and third steps  $N$  times;  
**Step 5.** Return  $\bar{F}_1$ .

**PROCEDURE 1:** Fuzzy simulation for possibilistic constraints.

**Input:** Decision variables  $X_{ij}$   
**Output:** The possibility  
**Step 1.** Set  $L = \alpha$  as a lower estimation  
**Step 2.** Randomly generate  $\kappa_{ij}$  from the  $\delta$ -level set of the fuzzy numbers  $\widetilde{ew}_{ij}$   
**Step 3.** Set  $\mu = \max\{\mu_{\widetilde{ew}_{ij}}, i = 1, 2, \dots, m, j = 1, 2, \dots, n\}$   
**Step 4.** If  $\sum_{i=1}^m \sum_{j=1}^n \kappa_{ij} X_{ij} \leq EW^U$  and  $L < \mu$ , set  $L = \mu$   
**Step 5.** Repeat the second and third steps  $N$  times  
**Step 6.** Return  $L$ .

**PROCEDURE 2:** Possibilistic constraint check.

simulated annealing algorithms including SMOSA, UMOSA, PSA, and WMOSA to solve multiobjective optimization of constrained problems with varying degree of complexity and then proposed a new algorithm PDMOSA. Sanghamitra et al. [41] proposed a simulated annealing-based multiobjective optimization algorithm (AMOSA) that incorporates the concept of archive in order to provide a set of trade-off solutions for the problem under consideration.

In the following part, we will incorporate the fuzzy simulation into the SMOSA algorithm proposed by Suppapitnarm and Parks [16] and use the interactive method to search the Pareto optimal solution for the bi-level multiobjective optimization with fuzzy possibilistic constraints. Take the problem (A.10) as an example and denote  $\mathbf{X}_i = (X_{i1}, X_{i2}, \dots, X_{in})$  and  $\mathbf{Y} = (Y_1, Y_2, \dots, Y_m)$ , the process of FS-ISA can be summarized in Procedure 3.

Above all, the whole procedure of FS-ISA for bi-level multiobjective optimization problems with fuzzy coefficients is described in Figure 3.

## 4. A Case Study

In the following, a practical example in China is introduced to demonstrate the complete modelling and algorithm process.

**Input:** The initial temperature  $t_0$   
**Output:** Pareto-solution  $Y_i^*$  and  $X_{ij}^*$  for all  $i$  and  $j$

**Step 1.** Randomly generate a feasible solution  $Y$  according to the fuzzy simulation for probabilistic constraints and take it as the initial parameter for the lower level;

**Step 2.** Solve all the multiobjective optimization problems on the lower level by SMOSA based on the fuzzy simulation and we obtain the Pareto optimal solution  $X_i$  for all  $i$ . Put  $G = (X_1^T, X_2^T, \dots, X_m^T, Y^T)$  into a Pareto set of solutions and compute all objective values of the upper and lower levels;

**Step 3.** Generate a new solution  $G^1 = (X_1^{1T}, X_2^{1T}, \dots, X_m^{1T}, Y^{1T})$  in the neighborhood of  $G$  by the random perturbation;

**Step 4.** Check the feasibility by fuzzy simulation according to all the constraints on both levels. If not, return to Step 3;

**Step 5.** Compute the objective values on both level, respectively. Compare the generated solution with all solutions in the Pareto set and update the Pareto set if necessary;

**Step 6.** Replace the current solution  $G$  with the generated solution  $G^1$  if  $G^1$  is archived and go to Step 7;

**Step 7.** Accept the generated solution  $Y^1$  as the input solution for the lower level if it is not archived with the probability:  $probability(p) = min(1, exp\{-\Delta s_i/t_i\})$ , where  $\Delta s_i = F_1^*(G) - F_1^*(G^1) + F_2(G^1) - F_2(G) + F_3(G^1) - F_3(G)$ . If the generated solution is accepted, take it into the lower level and solve them. Then we get a new solution  $G^{1*} = (X_1^{1T*}, X_2^{1T*}, \dots, X_m^{1T*}, Y^{1T})$  and put it into the Pareto set. If not, go to Step 9;

**Step 8.** Compare  $G^{1*}$  and  $G^1$  according to the evaluation function based on the compromise approach proposed by Xu and Li [15]. If  $G^{1*}$  is more optimal than  $G^1$ , let  $G = G^{1*}$ . If not  $G = G^1$

**Step 9.** Periodically, restart with a randomly selected solution from the Pareto set.  
While periodically restarting with the archived solutions, Suppapitnarm et al. [16] have recommended biasing towards the extreme ends of the trade-off surface;

**Step 10.** Periodically reduce the temperature by using a problem-dependent annealing schedule

**Step 11.** Repeat steps 2–10, until a predefined number of iterations is carried out.

**PROCEDURE 3:** FS-ISA algorithm for bi-level multi-objective programming.

#### 4.1. Data and Computation

Yingjing County is a famous county in China for its rich mineral products. The granite in this area has stable physical and chemical properties so that it can be processed into many useful stone products, mainly including granite slabs, man-made composite slabs, granite sands, and nano calcium carbonates (see Figure 4). Figure 5 shows the actual stone industry process from exploitation to production. Due to the vegetation deterioration, air and water pollution, and an aggravation of the ecological environment caused by the disordered exploitation and production manner, it is urgent for both the Yingjing government and the stone plants to optimize the assignment strategy.

Up to now, around 1 billion  $m^3$  of granite available is being exploited in Yingjing County according to the investigation. At present, only 7 stone plants have been built in this county, but the government plans to extend this to 10 stone plants in 2013, with all ten plants sharing the granite resource. From the historical data, stone dust and waste water emission coefficients are fuzzified and crossover validation tested. The test demonstrates that when the membership function is triangular, the percentage of correct results is the largest 92.32%. Therefore, the emission coefficients are regarded as fuzzy numbers in Tables 1 and 3. According to the environmental sector in this county, stone dust emissions should not exceed 2500 tonnes and waste water emission should not exceed 2500 tonnes. Although it is difficult

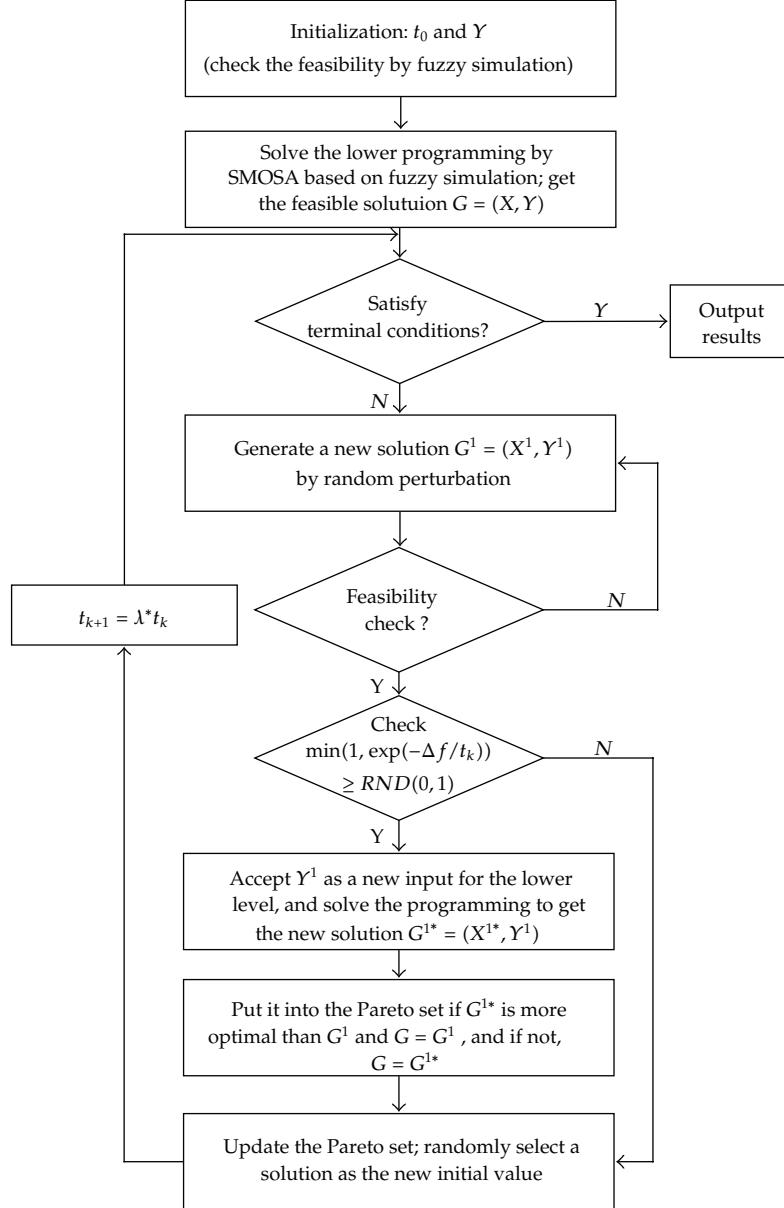


Figure 3: Flow chart for FS-ISA.

to satisfy the constrained index in a short time due to uncertainty, the possibility of holding the two constraints should not be less than 0.9 which indicates that the possibilistic levels  $\delta_2^U$  and  $\delta_3^U$  for the government should also be 0.9. For total emissions, the environmental sector requires the minimum objective to be under the possibilistic level  $\delta_1^U = 0.85$ . As the demand and the price of the four stone products sharply increase, the government requires that their output from all the plants should at least satisfy the basic market demand  $D_j^L$  ( $j = 1, \dots, 4$ ) as

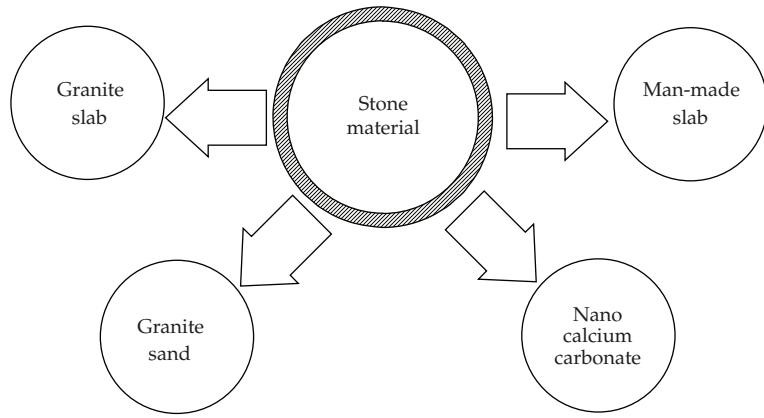


Figure 4: Products from the granite.

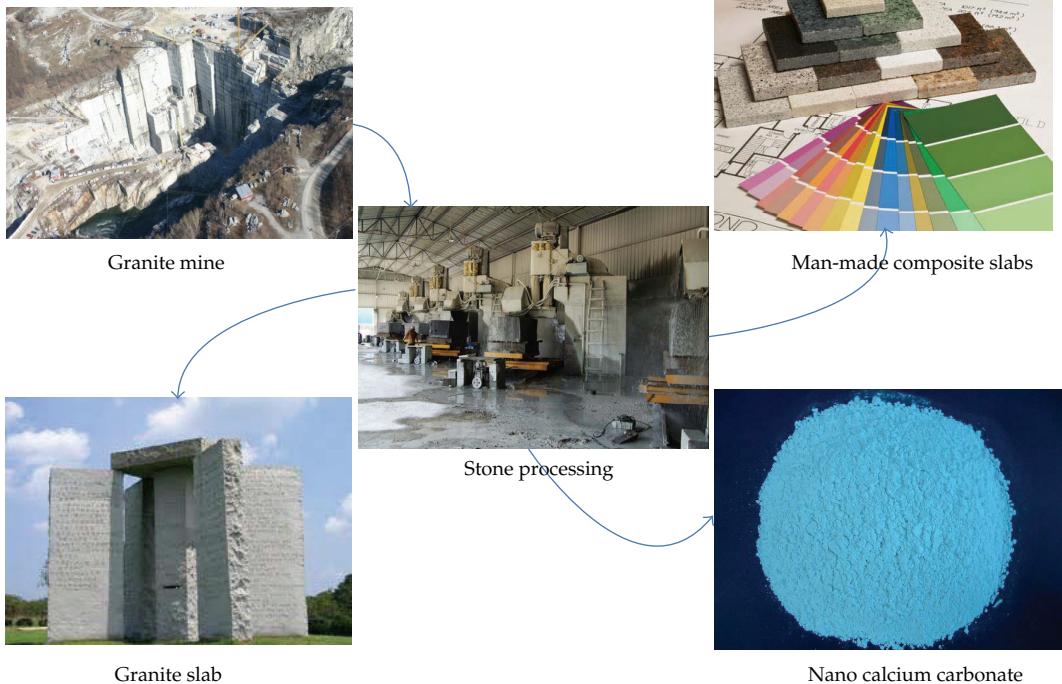


Figure 5: Basic flow chart of stone industry.

in Table 2. Each stone products' unit price is in Table 2. For the 10 stone plants, the inventory and the production upper limitations for each plant are listed in Table 1. The possibilistic level  $\delta_i^L$  that plant  $i$  needs to obtain for minimum emissions is in Table 1. Since every plant has a different capacity for controlling emissions, the fixed and unit variable cost, emission coefficients, and constant costs are different as outlined in Table 3. The transformation rate  $\theta_{ij}$  and the lower limitation of product  $j$  in plant  $i$  are also listed in Table 3.

**Table 1:** Parameters for every stone plant.

Stone plants	Parameters					
	$\widetilde{Ed}_i$ (kg/m <sup>3</sup> )	$P_i$ (Person)	$S_i$	$h_i$ (Yuan/m <sup>3</sup> )	$IV_i^U$ (M m <sup>3</sup> )	$PC_i^U$ (M Yuan)
Kai Quan	(24.5, 25.7, 27.8)	65	0.3655	0.32	4.5	1350
Feng Huang	(21.2, 26.8, 29.3)	60	0.3705	0.35	6.2	1670
Li Du	(19.2, 23.8, 25.7)	65	0.3735	0.33	4.3	1410
Hong Yuan	(22.6, 27.9, 28.3)	62	0.3615	0.28	7.8	1630
Xiang Zong	(21.0, 23.5, 27.1)	60	0.3675	0.38	2.3	1000
Ji Cheng	(20.6, 23.3, 27.9)	55	0.3485	0.42	2.6	1100
Hui Huang	(18.3, 21.2, 25.2)	60	0.3525	0.35	5.4	1650
Hong Yun	(19.2, 24.8, 29.1)	60	0.3725	0.36	4.6	1320
De Sheng	(25.4, 28.1, 30.3)	70	0.3615	0.33	5.6	1230
Guo Jian	(27.2, 29.3, 32.8)	80	0.3435	0.41	6.2	1670

**Table 2:** Parameters of each product.

Parameters	Stone products			
	Nano calcium carbonates	Granite slabs	Granite sand	Man-made composite slabs
$c_j$	1325 (Yuan/ton)	65 (Yuan/m <sup>2</sup> )	25 (Yuan/ton)	30 (Yuan/m <sup>2</sup> )
$D_j$	$8.5 \times 10^7$ (ton)	$3.06 \times 10^8$ (m <sup>2</sup> )	$5.13 \times 10^6$ (ton)	$1.56 \times 10^8$ m <sup>2</sup>

Taking all the numerical values into (2.21) and setting the initial temperature  $T_0 = 500$ , the last temperature is 0 and the cooling method is 1 decrement once. The neighbourhood can be developed as  $Y_i^1 = Y_i^0 + rh$  and  $X_{ij}^1 = X_{ij}^0 + rh$ , where  $r$  is a random number in  $(-1,1)$  and  $h$  is the step length (here  $h = 2.0$ ). After a simulation of many cycles, the Pareto optimal solution and the objective value are determined as shown in Tables 4 and 5. The results illustrate that although some plants have the highest productive efficiency, their high emission coefficient will result in the low exploiting quotas such as Kai Quan, Guo Jian, and De Sheng. On the other hand, stone plants will tend to produce the high value-added but low emission products due to the environmental pressure and the limitation of exploiting quotas, such as nano calcium carbonates and man-made composite slabs. However, stone plants will abundantly produce the traditional products such as granite slabs because of the huge cost of those new products.

#### 4.2. Sensitivity Analysis

In fact, the decision maker is able to adjust the parameter to obtain different level solutions. From theoretical deduction, it is known that the probabilistic level is a key factor impacting the results. If the accuracy of  $\delta_i^U$  and  $\delta_i^L$  decreases, the feasible set is expanded and then a better Pareto optimal solution and a better Pareto optimal point are determined. From Table 5, it can be seen that the emissions increase and the economic profit and the employment decrease as the probabilistic level  $\delta_i^U$  ( $i = 1, 2, 3$ ) decreases indicating that the government requirements are less strict which results in the stone plants pursuing the economic profit and neglecting the emissions and the employment objectives. Finally, the total emissions increase and the government tax revenue decreases. On the other hand, if the probabilistic level  $\delta_i^U$  ( $i = 1, 2, 3$ )

**Table 3:** Parameters for product  $j$  produced by plant  $i$ .

Stone plants	Stone products	Parameters					
		$P_{ij}$	$t_{ij}$	$C_{ij}$	$\theta_{ij}$	$P_{ij}^L$	$\overline{ed}_{ij}$
Kai Quan	NPCC	0.03	650	286	1.03	7.0	(2.21, 3.42, 5.23)
	GSl	0.01	38	367	3.62	262.2	(21.2, 22.3, 25.6)
	GSa	0.01	10	80	0.95	0	(26.4, 28.5, 32.7)
	MmCS	0.02	22	440	8.34	35.5	(1.57, 2.68, 4.39)
Feng Huang	NPCC	0.03	580	254	1.12	7.2	(2.68, 3.67, 4.59)
	GSl	0.01	44	380	3.14	232.3	(24.7, 26.3, 27.6)
	GSa	0.01	16	68	0.90	2.4	(27.6, 29.0, 33.9)
	MmCS	0.02	24	448	8.11	43.1	(2.03, 3.56, 4.57)
Li Du	NPCC	0.03	620	272	0.85	9.9	(1.86, 2.31, 4.03)
	GSl	0.01	40	392	2.67	279.0	(17.4, 19.3, 20.2)
	GSa	0.01	13	88	0.93	1.9	(23.5, 26.5, 28.7)
	MmCS	0.02	26	380	8.35	67.4	(1.45, 2.23, 4.05)
Hong Yuan	NPCC	0.03	685	310	0.78	0	(3.05, 4.21, 5.67)
	GSl	0.01	42	370	3.78	298.8	(20.8, 23.3, 23.9)
	GSa	0.01	13	86	0.92	2.1	(26.5, 28.3, 30.4)
	MmCS	0.02	23	380	8.26	22.0	(1.57, 2.68, 4.39)
Xiang Zong	NPCC	0.03	632	267	1.26	10.2	(2.17, 3.33, 4.78)
	GSl	0.01	44	380	3.82	307.2	(20.3, 23.5, 26.7)
	GSa	0.01	12	82	0.91	1.9	(26.4, 27.3, 29.3)
	MmCS	0.02	20	350	8.27	39.4	(1.46, 2.79, 3.45)
Ji Cheng	NPCC	0.03	630	264	1.26	11.1	(2.14, 3.39, 5.46)
	GSl	0.01	42	354	3.46	344.5	(22.5, 23.8, 24.7)
	GSa	0.01	12	85	0.92	2.2	(25.8, 27.9, 28.9)
	MmCS	0.02	23	350	8.26	0	(1.76, 2.77, 4.25)
Hui Huang	NPCC	0.03	635	260	1.12	12.7	(2.15, 3.56, 5.00)
	GSl	0.01	35	340	3.87	451.7	(20.7, 23.2, 24.7)
	GSa	0.01	15	85	0.90	0	(28.6, 29.9, 33.4)
	MmCS	0.02	20	350	8.26	43.6	(1.68, 2.70, 4.25)
Hong Yun	NPCC	0.03	660	290	1.00	10.0	(2.54, 3.68, 5.42)
	GSl	0.01	40	380	3.54	331.3	(22.3, 23.5, 24.2)
	GSa	0.01	12	85	0.90	2.4	(25.3, 26.7, 30.3)
	MmCS	0.02	24	385	8.42	0	(1.33, 2.55, 4.72)
De Sheng	NPCC	0.03	630	276	1.12	8.0	(2.35, 3.67, 4.68)
	GSl	0.01	42	383	3.11	203.5	(20.5, 21.4, 23.8)
	GSa	0.01	13	85	0.92	0	(23.6, 25.2, 28.6)
	MmCS	0.02	25	378	8.02	48.0	(1.67, 2.78, 4.23)
Guo Jian	NPCC	0.03	780	320	1.88	12.5	(3.14, 4.37, 7.86)
	GSl	0.01	40	380	3.62	210.4	(21.2, 22.3, 25.6)
	GSa	0.01	13	86	0.91	0	(27.3, 29.4, 33.8)
	MmCS	0.02	22	367	8.13	51.1	(1.32, 2.59, 4.21)

NPCC: Nano calcium carbonates; GSl: granite slabs; GSa: granite sand; MmCS: man-made composite slabs.

**Table 4:** Assignment results for different products.

Stone plants	Total	Stone products				
		Nano calcium carbonates	Graniteslabs	Granitesand	Man-made composite slabs	
Kai Quan	85.2	6.82	72.40	1.70	4.30	
Feng Huang	88.6	6.47	74.16	2.66	5.32	
Li Du	126.2	11.61	104.49	2.02	8.08	
Hong Yuan	91.7	7.70	79.05	2.29	2.66	
Xiang Zong	95.4	8.11	80.42	2.10	4.77	
Ji Cheng	112.9	8.81	99.58	2.37	2.15	
Hui Huang	135.4	11.37	116.71	2.03	5.28	
Hong Yun	112.5	10.01	93.60	2.70	6.19	
De Sheng	79.8	7.18	65.44	1.20	5.99	
Guo Jian	72.3	6.65	58.13	1.23	6.29	

**Table 5:** Objectives for both the upper and lower levels.

Notation	$F_1^*$	$F_2$	$F_3$	$H_1^1$	$H_2^1$	$H_3^1$	$H_4^1$	$H_5^1$	$H_6^1$	$H_7^1$	$H_8^1$	$H_9^1$
$\delta_i^U = 0.95$	66289	12841	61240	5443	5281	9722	5648	6304	7403	10387	7930	5419
$\delta_i^U = 0.90$	68362	13216	62530	5587	5362	9910	5753	6421	7489	11253	8016	5578
$\delta_i^U = 0.85$	69137	13781	63110	5612	5374	9983	5842	6511	7570	11891	8117	5632
Notation	$H_{10}^1$	$H_1^{2*}$	$H_2^{2*}$	$H_3^{2*}$	$H_4^{2*}$	$H_5^{2*}$	$H_6^{2*}$	$H_7^{2*}$	$H_8^{2*}$	$H_9^{2*}$	$H_{10}^{2*}$	—
$\delta_i^U = 0.95$	3990	3524	3994	4894	3887	4085	4953	5867	4624	3112	2866	—
$\delta_i^U = 0.90$	4114	3678	4953	4930	3922	4137	4953	5952	4731	3220	2917	—
$\delta_i^U = 0.85$	3990	3524	3994	4894	3887	4236	5078	6013	4827	3315	3013	—

increases, the government requirements are more strict and hence the total emissions decrease and the government tax revenue increases.

Similarly, for the following level, if the possibilistic levels  $\delta_i^L$  ( $i = 1, 2, \dots, 10$ ) decrease, the plants pay less attention to the stone dust and waste water emissions resulting in an increase in profit and consequently more emission.

### 4.3. Comparison Analysis

For the proposed case, all the emission coefficients including  $\widetilde{Ed}_i$ ,  $\widetilde{ed}_{ij}$ , and  $\widetilde{ew}_{ij}$  are fuzzified as triangular fuzzy numbers according to the real-life situation. Because all the equations in the model are linear, it actually can be easily converted into a crisp model without uncertain coefficients by the possibility measure. Lemma A.1 is given to show the process in the Appendix section, and we can get the crisp model according to (A.10). Taking all the numerical values into (A.10) and setting the same parameters for ISA, we can easily get the optimal solutions. The error analysis and computation time are listed in Table 6 and Table 7, respectively. It is obvious that the results from solving the crisp equivalent model are close to the results from simulating the model. It shows that the the fuzzy simulation technique is reasonable and efficient to solve the

**Table 6:** Errors analysis by solving crisp equivalent model.

Stone plants	Total	Stone products			Man-made composite slabs
		Nano calcium carbonates	Granite slabs	Granite sand	
Kai Quan	1.32%	0.25%	-0.56%	0.32%	0.08%
Feng Huang	0.83%	-0.27%	-0.12%	1.12%	0.43%
Li Du	-1.49%	1.04%	-1.65%	0.57%	-0.13%
Hong Yuan	0.17%	0.12%	-0.31%	0.09%	0.12%
Xiang Zong	-0.28%	0.15%	0.36%	0.35%	1.25%
Ji Cheng	0.36%	0.22%	0.18%	-0.24%	0.35%
Hui Huang	-0.21%	0.13%	0.27%	0.28%	-0.12%
Hong Yun	0.22%	0.33%	-0.08%	0.12%	0.24%
De Sheng	-0.34%	-0.11%	0.16%	0.27%	0.34%
Guo Jian	0.62%	1.04%	0.20%	0.03%	0.38%

**Table 7:** Computing time and memory by ISA and GA.

No.	Size of tested problem			$T_0$	Gen	ISA		FS-ISA		FS-GA	
	Resources	Plants	Decision variables			ACT	Memory	ACT	Memory	ACT	Memory
1	1	5	20	500	—	65	100	120	100	—	—
	1	10	60	—	500	—	—	—	—	245	100
2	1	10	60	500	—	245	600	425	600	—	—
	1	10	60	—	500	—	—	—	—	620	600
3	4	10	60	500	—	455	2400	1560	2400	—	—
	4	10	60	—	500	—	—	—	—	1020	2400

ACT: average computing time (second); Memory: required memory space to represent a solution.

model for bi-level multiobjective optimization problems with fuzzy coefficients. At the same time, it is found from Table 7 that the average computational time by ISA is less than the time by FS-ISA. It is also reasonable because the process of fuzzy simulation for possibilistic constraint will spend much time to get the approximate value. However, not all possibilistic constraints can be directly converted into crisp ones. Lemma A.1 is efficient only for the special membership functions such as the triangular and trapezoidal fuzzy numbers.

To illustrate that FS-ISA is suitable for this kind of fuzzy bi-level model, the results are compared with a genetic algorithm (GA). GA is one of the most popular algorithms. Many scholars also made the comparison between SA and GA in solving bi-level optimization problems [12, 42, 43]. They regard that different data scales will result in huge differences on the computational efficiency. To ensure the fairness, we also design the GA based on the fuzzy simulation for the bi-level multiobjective optimization with fuzzy coefficients. We set the chromosome number 20, the crossover rate 0.6, the mutation rate 0.8, and the iterative number 500. The average computing time and memory are listed in Table 7. Experiments show that the similar optimal results can be obtained by both FS-ISA and FS-GA, but the computational efficiency is different when the number of stone resources and stone plants

changes. It is found that when the number of stone resources and stone plants is small, FS-ISA is more efficient than GA in solving the bi-level multiobjective optimization and much more computational effort is needed for FS-GA to achieve the same optimal solution as FS-ISA. However, when the data scale is large, FS-GA can reach a more optimal solution at the expense of more computation time. The result is in accordance with the findings by Xu et al. [43]. Of course, if the fuzzy bi-level multi-objective optimization model can be easily converted into the crisp model, we can obtain a more accurate solution and spend less time by ISA than that by FS-ISA.

## 5. Conclusions

In this paper, we have developed a bi-level multi-objective optimization model with probabilistic constraints under the fuzzy environment. In the model, the government is considered as the leader level for minimizing the emissions of the stone dust and the waste water and maximizing the employment and economic growth, and then stone plants are considered as the follower level for maximizing the profit and minimizing the emissions. Then we propose an algorithm FS-ISA to solve the model. Finally, a practical case proves that the proposed model and algorithm are efficient.

Although the model proposed in this paper should be helpful for solving some real-world problems, it only dealt with by the probabilistic constraints. If DM has different purposes such as maximizing the possibility that the predetermined goals are achieved, we can apply dependent-chance constraint to deal with it. In further research to be undertaken, a detailed analysis will be given.

## Appendix

**Lemma A.1.** Assume that  $\widetilde{Ed}_i$ ,  $\widetilde{ed}_{ij}$ , and  $\widetilde{ew}_{ij}$  ( $i = 1, 2, \dots, m$ ;  $j = 1, 2, \dots, n$ ) are L-R fuzzy numbers with the following membership functions:

$$\mu_{\widetilde{Ed}_i}(t) = \begin{cases} L\left(\frac{Ed_i - t}{\alpha_i^{Ed}}\right), & t < Ed_i, \alpha_i^{Ed} > 0, \\ R\left(\frac{t - Ed_i}{\beta_i^{Ed}}\right), & t \geq Ed_i, \beta_i^{Ed} > 0, \end{cases} \quad (\text{A.1})$$

$$\mu_{\widetilde{ed}_{ij}}(t) = \begin{cases} L\left(\frac{ed_{ij} - t}{\alpha_{ij}^{ed}}\right), & t < ed_{ij}, \alpha_{ij}^{ed} > 0, \\ R\left(\frac{t - ed_{ij}}{\beta_{ij}^{ed}}\right), & t \geq ed_{ij}, \beta_{ij}^{ed} > 0, \end{cases} \quad (\text{A.2})$$

$$\mu_{\bar{ew}_{ij}}(t) = \begin{cases} L\left(\frac{ew_{ij} - t}{\alpha_{ij}^{ew}}\right), & t < ew_{ij}, \alpha_{ij}^{ew} > 0, \\ R\left(\frac{t - ew_{ij}}{\beta_{ij}^{ew}}\right), & t \geq ew_{ij}, \beta_{ij}^{ew} > 0, \end{cases} \quad (\text{A.3})$$

where  $\alpha_i^{Ed}$ ,  $\beta_i^{Ed}$  are positive numbers expressing the left and right spreads of  $\bar{Ed}$ ,  $\alpha_{ij}^{ed}$ ,  $\beta_{ij}^{ed}$  are positive numbers expressing the left and right spreads of  $\bar{ed}$ , and  $\alpha_{ij}^{ew}$ ,  $\beta_{ij}^{ew}$  are positive numbers expressing the left and right spreads of  $\bar{ew}$ ,  $i = 1, 2, \dots, m$ ,  $j = 1, 2, \dots, n$ . Reference functions  $L, R : [0, 1] \rightarrow [0, 1]$  with  $L(1) = R(1) = 0$  and  $L(0) = R(0) = 1$  are nonincreasing, continuous functions. Then one has  $\text{Pos}\{\sum_{i=1}^m \bar{Ed}_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (\bar{ed}_{ij} X_{ij} + \bar{ew}_{ij} X_{ij}) \leq \bar{F}_1\} \geq \delta_1^U$  if and only if

$$\bar{F}_1 \geq \sum_{i=1}^m Ed_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (ed_{ij} + ew_{ij}) X_{ij} - L^{-1}(\delta_1^U) \left( \sum_{i=1}^m \alpha_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n (\alpha_{ij}^{ed} + \alpha_{ij}^{ew}) X_{ij} \right). \quad (\text{A.4})$$

*Proof.* Let  $\omega \in [0, 1]$  be any positive real number and  $L((Ed_i - x)/\alpha_i^{Ed}) = L((ed_{ij} - y)/\alpha_{ij}^{ed}) = L((ew_{ij} - z)/\alpha_{ij}^{ew}) = \omega$ , then from (A.3) we have

$$x = Ed_i - \alpha_i^{Ed} L^{-1}(\omega), \quad y = ed_{ij} - \alpha_{ij}^{ed} L^{-1}(\omega), \quad z = ew_{ij} - \alpha_{ij}^{ew} L^{-1}(\omega). \quad (\text{A.5})$$

For any  $Y_j, X_{ij} \geq 0$  ( $i = 1, 2, \dots, m; j = 1, 2, \dots, n$ ), it easily follows that

$$\begin{aligned} t &= \sum_{i=1}^m x Y_i + \sum_{i=1}^m \sum_{j=1}^n (y X_{ij} + z X_{ij}) \\ &= \left[ \sum_{i=1}^m Ed_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (ed_{ij} + ew_{ij}) X_{ij} \right] - \left( \sum_{i=1}^m \alpha_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n (\alpha_{ij}^{ed} + \alpha_{ij}^{ew}) X_{ij} \right) L^{-1}(\omega). \end{aligned} \quad (\text{A.6})$$

Therefore, we have

$$L\left(\frac{\sum_{i=1}^m Ed_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (ed_{ij} + ew_{ij}) X_{ij} - t}{\sum_{i=1}^m \alpha_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n (\alpha_{ij}^{ed} + \alpha_{ij}^{ew}) X_{ij}}\right) = \omega. \quad (\text{A.7})$$

It is also proved by similar technique that

$$R\left(\frac{t - (\sum_{i=1}^m Ed_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (ed_{ij} + ew_{ij}) X_{ij})}{\sum_{i=1}^m \beta_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n (\beta_{ij}^{ed} + \beta_{ij}^{ew}) X_{ij}}\right) = \omega. \quad (\text{A.8})$$

Hence, it is easily found that  $\sum_{i=1}^m \widetilde{Ed}_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (\widetilde{ed}_{ij} X_{ij} + \widetilde{ew}_{ij} X_{ij})$  is also a *L-R* fuzzy number with the left spread  $\sum_{i=1}^m \alpha_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n (\alpha_{ij}^{ed} + \alpha_{ij}^{ew}) X_{ij}$  and the right spread  $\sum_{i=1}^m \beta_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n (\beta_{ij}^{ed} + \beta_{ij}^{ew}) X_{ij}$ . According to the definition of possibility measure proposed by Dubois and Prade [28], it can be obtained as follows:

$$\begin{aligned} & \text{Pos}\left\{\sum_{i=1}^m \widetilde{Ed}_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (\widetilde{ed}_{ij} X_{ij} + \widetilde{ew}_{ij} X_{ij}) \leq \bar{F}_1\right\} \geq \delta_1^U \\ & \iff L\left(\frac{\sum_{i=1}^m Ed_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (ed_{ij} + ew_{ij}) X_{ij} - \bar{F}_1}{\sum_{i=1}^m \alpha_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n (\alpha_{ij}^{ed} + \alpha_{ij}^{ew}) X_{ij}}\right) \geq \delta_1^U \\ & \iff \frac{\sum_{i=1}^m Ed_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (ed_{ij} + ew_{ij}) X_{ij} - \bar{F}_1}{\sum_{i=1}^m \alpha_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n (\alpha_{ij}^{ed} + \alpha_{ij}^{ew}) X_{ij}} \leq L^{-1}(\delta_1^U) \quad (\text{A.9}) \\ & \iff \sum_{i=1}^m Ed_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (ed_{ij} + ew_{ij}) X_{ij} \\ & \quad - L^{-1}(\delta_1^U) \left( \sum_{i=1}^m \alpha_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n (\alpha_{ij}^{ed} + \alpha_{ij}^{ew}) X_{ij} \right) \leq \bar{F}_1. \end{aligned}$$

This completes the proof.  $\square$

From Lemma A.1, the model (2.21) is equivalent to the following bi-level multi-objective programming problem:

$$\left\{
 \begin{array}{l}
 \min F_1^* = \sum_{i=1}^m Ed_i Y_i + \sum_{i=1}^m \sum_{j=1}^n (ed_{ij} + ew_{ij}) X_{ij} - L^{-1}(\delta_1^U) \left( \sum_{i=1}^m \alpha_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n (\alpha_{ij}^{ed} + \alpha_{ij}^{ew}) X_{ij} \right) \\
 \max F_2 = \sum_{i=1}^m \left( \sum_{j=1}^n p_{ij} X_{ij} + P_i \right) \\
 \max F_3 = \sum_{i=1}^m S_i \left( \sum_{j=1}^n c_j \theta_{ij} X_{ij} \right) \\
 \quad \left. \begin{array}{l}
 \sum_{i=1}^m Ed_i Y_i + \sum_{i=1}^m \sum_{j=1}^n ed_{ij} X_{ij} - L^{-1}(\delta_2^U) \left( \sum_{i=1}^m \alpha_i^{Ed} Y_i + \sum_{i=1}^m \sum_{j=1}^n \alpha_{ij}^{ed} X_{ij} \right) \leq ED^U \\
 \sum_{i=1}^m \sum_{j=1}^n ew_{ij} X_{ij} - L^{-1}(\delta_3^U) \sum_{i=1}^m \sum_{j=1}^n \alpha_{ij}^{ew} X_{ij} \leq EW^U \\
 \sum_{i=1}^m \theta_{ij} X_{ij} \geq D_j^L \quad \forall j
 \end{array} \right. \\
 \text{s.t.} \quad \left. \begin{array}{l}
 \max H_i^1 = \sum_{j=1}^n c_j \theta_{ij} X_{ij} - \sum_{j=1}^n f(X_{ij}) - h_i \left( Y_i - \sum_{j=1}^n X_{ij} \right) \\
 \min H_i^{2*} = \sum_{j=1}^n (ed_{ij} + ew_{ij}) X_{ij} - L^{-1}(\delta_3^U) \sum_{j=1}^n (\alpha_{ij}^{ed} + \alpha_{ij}^{ew}) X_{ij} \\
 \quad \left. \begin{array}{l}
 \sum_{j=1}^n X_{ij} \leq Y_i \\
 Y_i - \sum_{j=1}^n X_{ij} \leq IV_i^U \\
 \sum_{j=1}^n f(X_{ij}) + h_i \left( Y_i - \sum_{j=1}^n X_{ij} \right) \leq PC_i^U \\
 \theta_{ij} X_{ij} \geq P_{ij}^L
 \end{array} \right. \\
 \quad \left. \begin{array}{l}
 \sum_{j=1}^n X_{ij} \leq Y_i \\
 Y_i - \sum_{j=1}^n X_{ij} \leq IV_i^U \\
 \sum_{j=1}^n f(X_{ij}) + h_i \left( Y_i - \sum_{j=1}^n X_{ij} \right) \leq PC_i^U \\
 \theta_{ij} X_{ij} \geq P_{ij}^L
 \end{array} \right.
 \end{array} \right. \\
 \end{array} \right. \quad (A.10)$$

## Abbreviations

### Indices

- $i$  : Index of stone-material plants,  $i = 1, 2, \dots, m$
- $j$  : Index of stone products,  $j = 1, 2, \dots, n$ .

### Parameters

- $\widetilde{Ed}_i$  : Stone dust emissions coefficient when plant  $i$  exploits
- $\widetilde{ed}_{ij}$  : Stone dust emissions coefficient when plant  $i$  produces product  $j$

$\bar{w}_{ij}$  : Waste water emissions coefficient when that plant  $i$  produces product  $j$

$p_{ij}$  : Employment coefficient that plant  $i$  produces product  $j$

$P_i$  : Basic employment that plant  $i$  needs

$S_i$  : Unit tax rate that plant  $i$  pays to the government

$c_j$  : Unit price of product  $j$

$t_{ij}$  : Unit variable cost when plant  $i$  produces product  $j$

$h_i$  : Unit cost when plant  $i$  holds remnant stone materials

$\theta_{ij}$  : Transformation rate when plant  $i$  produces product  $j$

$C_{ij}$  : Constant cost if plant  $i$  produces product  $j$

$R^U$  : Total stone resources upper limitation in the region

$D_j^L$  : Lower limitation of product  $j$  demand

$ED^U$  : Stone dust total emissions upper limitation in the region

$EW^U$  : Waste water total emissions upper limitation in the region

$IV_i^U$  : Inventory upper limitation for plant  $i$

$PC_i^U$  : Production cost upper limitation for plant  $i$

$P_{ij}^L$  : Lower limitation for product  $j$  in plant  $i$ .

### Decision variables

$Y_i$  : Amount that the government allows plant  $i$  to exploit

$X_{ij}$  : Amount that plant  $i$  uses to produce the product  $j$ .

### Acknowledgments

The work is supported by the Key Program of National Natural Science Foundation of China (Grant no. 70831005) and also supported by the “985” Program of Sichuan University (Innovative Research Base for Economic Development and Management).

### References

- [1] M. Al-Jabari and H. Sawalha, “Treating stone cutting waste by flocculation-sedimentation,” in *Proceedings of the Sustainable Environmental Sanitation and Water Services Conference, 28th WEDC Conference*, Calcutta, India, 2002.
- [2] K. Nasserine, Z. Mimi, B. Bevan, and B. Elian, “Environmental management of the stone cutting industry,” *Journal of Environmental Management*, vol. 90, no. 1, pp. 466–470, 2009.
- [3] N. Almeida, F. Branco, and J. R. Santos, “Recycling of stone slurry in industrial activities: application to concrete mixtures,” *Building and Environment*, vol. 42, no. 2, pp. 810–819, 2007.

- [4] V. Corinaldesi, G. Moriconi, and T. R. Naik, "Characterization of marble powder for its use in mortar and concrete," *Construction and Building Materials*, vol. 24, no. 1, pp. 113–117, 2010.
- [5] F. Saboya Jr., G. C. Xavier, and J. Alexandre, "The use of the powder marble by-product to enhance the properties of brick ceramic," *Construction and Building Materials*, vol. 21, no. 10, pp. 1950–1960, 2007.
- [6] N. Bilgin, H. A. Yeremb, S. Arslan et al., "Use of waste marble powder in brick industry," *Construction and Building Materials*, vol. 29, pp. 449–457, 2012.
- [7] A. Singh and H. H. Lou, "Hierarchical pareto optimization for the sustainable development of industrial ecosystems," *Industrial and Engineering Chemistry Research*, vol. 45, no. 9, pp. 3265–3279, 2006.
- [8] S. R. Lim and M. P. Jong, "Cooperative water network system to reduce carbon footprint," *Environmental Science and Technology*, vol. 42, no. 16, pp. 6230–6236, 2008.
- [9] D. Petrovic, R. Roy, and R. Petrovic, "Supply chain modelling using fuzzy sets," *International Journal of Production Economics*, vol. 59, no. 1, pp. 443–453, 1999.
- [10] H. M. Lee and J. S. Yao, "Economic production quantity for fuzzy demand quantity and fuzzy production quantity," *European Journal of Operational Research*, vol. 109, no. 1, pp. 203–211, 1998.
- [11] R. E. Bellman and L. A. Zadeh, "Decision-making in a fuzzy environment," *Management Science*, vol. 17, pp. B141–B164, 1970.
- [12] S. R. Hejazi, A. Memariani, G. Jahanshahloo, and M. M. Sepehri, "Linear bilevel programming solution by genetic algorithm," *Computers & Operations Research*, vol. 29, no. 13, pp. 1913–1925, 2002.
- [13] M. Al-Salamah, "Temporary workforce planning with firm contracts: a model and a simulated annealing heuristic," *Mathematical Problems in Engineering*, vol. 2011, Article ID 209693, 18 pages, 2011.
- [14] M. Gendreau, P. Marcotte, and G. Savard, "A hybrid tabu-ascent algorithm for the linear bilevel programming problem," *Journal of Global Optimization*, vol. 8, no. 3, pp. 217–233, 1996.
- [15] J. Li, J. Xu, and M. Gen, "A class of multiobjective linear programming model with fuzzy random coefficients," *Mathematical and Computer Modelling*, vol. 44, no. 11–12, pp. 1097–1113, 2006.
- [16] A. Suppapitnarm and G. T. Parks, *Simulated Annealing: An Alternative Approach to True Multiobjective Optimization*, Morgan Kaufmann, 1999.
- [17] J. S. Yao and F. T. Lin, "Constructing a fuzzy flow-shop sequencing model based on statistical data," *International Journal of Approximate Reasoning*, vol. 29, no. 3, pp. 215–234, 2002.
- [18] J. S. Yao and K. Wu, "Consumer surplus and producer surplus for fuzzy demand and fuzzy supply," *Fuzzy Sets and Systems*, vol. 103, no. 3, pp. 421–426, 1999.
- [19] C. J. Lin and U. P. Wen, "A labeling algorithm for the fuzzy assignment problem," *Fuzzy Sets and Systems*, vol. 142, no. 3, pp. 373–391, 2004.
- [20] J. Xu, L. Yao, and X. Zhao, "A multi-objective chance-constrained network optimal model with random fuzzy coefficients and its application to logistics distribution center location problem," *Fuzzy Optimization and Decision Making*, vol. 10, no. 3, pp. 255–285, 2011.
- [21] J. Xu and Z. Zeng, "A discrete time optimal control model with uncertainty for dynamic machine allocation problem and its application to manufacturing and construction industries," *Applied Mathematical Modelling*, vol. 36, no. 8, pp. 3513–3544, 2012.
- [22] J. C. Bezdek, "Fuzzy models—what are they, and why?" *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 1, pp. 1–6, 1993.
- [23] D. Dubois and H. Prade, *Fuzzy Sets and Systems*, Academic Press, New York, NY, USA, 1980.
- [24] U. Thole, H. J. Zimmermann, and P. Zysno, "On the suitability of minimum and product operators for the intersection of fuzzy sets," *Fuzzy Sets and Systems*, vol. 2, no. 2, pp. 167–180, 1979.
- [25] B. Kosko, *Neural Networks and Fuzzy Systems*, Prentice Hall, Englewood Cliffs, NJ, USA, 1992.
- [26] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," *International Joint Conference on Artificial Intelligence*, vol. 14, pp. 1137–1145, 1995.
- [27] J. Xu and X. Zhou, *Fuzzy-Like Multiple Objective Decision Making*, Springer, 2011.
- [28] D. Dubois and H. Prade, "Operations on fuzzy numbers," *International Journal of Systems Science*, vol. 9, no. 6, pp. 613–626, 1978.
- [29] A. Kaufmann and M. M. Gupta, *Introduction to Fuzzy Arithmetic*, Van Nostrand Reinhold, New York, NY, USA, 1985.
- [30] J. F. Bard, "Some properties of the bilevel programming problem," *Journal of Optimization Theory and Applications*, vol. 68, no. 2, pp. 371–378, 1991.
- [31] P. Hansen, B. Jaumard, and G. Savard, "New branch-and-bound rules for linear bilevel programming," *Society for Industrial and Applied Mathematics*, vol. 13, no. 5, pp. 1194–1217, 1992.

- [32] O. Ben-Ayed and C. E. Blair, "Computational difficulties of bilevel linear programming," *Operations Research*, vol. 38, no. 3, pp. 556–560, 1990.
- [33] S. Kirkpatrick Jr., C. D. Gelatt,, and M. P. Vecchi, "Optimization by simulated annealing," *American Association for the Advancement of Science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [34] S. Kirkpatrick, "Optimization by simulated annealing: quantitative studies," *Journal of Statistical Physics*, vol. 34, no. 5-6, pp. 975–986, 1984.
- [35] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.
- [36] P. J. M. van Laarhoven, *Theoretical and Computational Aspects of Simulated Annealing*, vol. 51, Stichting Mathematisch Centrum Centrum voor Wiskunde en Informatica, Amsterdam, The Netherlands, 1988.
- [37] J. M. Steel, "A review of Laarhoven," *Journal of the American Statistical Association*, vol. 85, pp. 596–596, 1990.
- [38] A. Suppapitnarm, K. A. Seffen, G. T. Parks, and P. J. Clarkson, "Simulated annealing algorithm for multiobjective optimization," *Engineering Optimization*, vol. 33, no. 1, pp. 59–85, 2000.
- [39] B. Suman, "Study of simulated annealing based algorithms for multiobjective optimization of a constrained problem," *Computers and Chemical Engineering*, vol. 28, no. 9, pp. 1849–1871, 2004.
- [40] B. Suman and P. Kumar, "A survey of simulated annealing as a tool for single and multiobjective optimization," *Journal of the Operational Research Society*, vol. 57, no. 10, pp. 1143–1160, 2006.
- [41] B. Sanghamitra, S. Sriparna, M. Ujjwal, and D. Kalyanmoy, "A simulated annealing-based multiobjective optimization algorithm: AMOSA," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 3, pp. 269–283, 2008.
- [42] A. Sadegheih, "Scheduling problem using genetic algorithm, simulated annealing and the effects of parameter values on GA performance," *Applied Mathematical Modelling*, vol. 30, no. 2, pp. 147–154, 2006.
- [43] T. Xu, H. Wei, and G. Hu, "Study on continuous network design problem using simulated annealing and genetic algorithm," *Expert Systems with Applications*, vol. 36, no. 2, pp. 1322–1328, 2009.

*Research Article*

## **Dynamic Programming and Heuristic for Stochastic Uncapacitated Lot-Sizing Problems with Incremental Quantity Discount**

**Yuli Zhang,<sup>1</sup> Shiji Song,<sup>1</sup> Cheng Wu,<sup>1</sup> and Wenjun Yin<sup>2</sup>**

<sup>1</sup> Department of Automation, TNList, Tsinghua University, Beijing 100084, China

<sup>2</sup> Industry Solutions, IBM Research-China, Beijing 100193, China

Correspondence should be addressed to Shiji Song, shjis@mail.tsinghua.edu.cn

Received 30 January 2012; Revised 25 April 2012; Accepted 8 May 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Yuli 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.

The stochastic uncapacitated lot-sizing problems with incremental quantity discount have been studied in this paper. First, a multistage stochastic mixed integer model is established by the scenario analysis approach and an equivalent reformulation is obtained through proper relaxation under the decreasing unit order price assumption. The proposed reformulation allows us to extend the production-path property to this framework, and furthermore we provide a more accurate characterization of the optimal solution. Then, a backward dynamic programming algorithm is developed to obtain the optimal solution and considering its exponential computation complexity in term of time stages, we design a new rolling horizon heuristic based on the proposed property. Comparisons with the commercial solver CPLEX and other heuristics indicate better performance of our proposed algorithms in both quality of solution and run time.

### **1. Introduction**

The lot-sizing problems have been the subject of intensive research in the last few decades. The basic definition of single-item lot-sizing problems can be stated as follows: the order (production), inventory and backlog quantities in each period should be determined to meet the deterministic but dynamic demand over a finite time horizon. The objective is to minimize the total costs, which consist of the fixed setup cost, order cost and inventory cost. Different quantity discount policies, such as all-units quantity discount and incremental quantity discount, have been widely executed in practice and thus have also been introduced into the lot-sizing problems.

Although the deterministic planning and scheduling models have been intensively studied, in practice there are many different sources of uncertainties, such as customer

demand, production lead-time, and product price, which make information that will be needed in subsequent decision stages unavailable to the decision maker. In such cases, the solution provided by a deterministic model may be of little value in terms of applicability of the model's recommendations, see Beraldi et al. [1]. Thus the stochastic version of lot-sizing problems has been studied recently and with the advent of stochastic programming, the classical deterministic lot-sizing models have been extended to scenario-based multistage stochastic mixed integer programming.

Wagner and Whitin [2] first introduced the single-item dynamic lot-sizing problems without backlogging. They proposed a dynamic programming approach based on the Wagner-Whitin (W-W) property, that is, no production is undertaken if inventory is available. Because W-W property holds for the nondiscount problem, many heuristics have been developed based on this property; however, in the all-units quantity discount case, the property does not necessarily hold due to its discontinuous piecewise linear cost structure by Hu and Munson [3]. For the modified all-units discount problem, Chan et al. [4] demonstrated that a zero-inventory-ordering policy based on the W-W property exists, whose cost is no more than  $4/3$  times the optimal cost. Federgruen and Lee [5] characterized structural properties of optimal solutions for both all-units and incremental quantity discount, and proposed dynamic programming algorithms with complexity  $O(T^3)$  and  $O(T^2)$ , respectively, and with  $T$  being the number of periods in the planning horizon. For deterministic capacitated lot-sizing problem with general piecewise linear cost function, Shaw and Wagelmans [6] presented a dynamic programming procedure with complexity  $O(T^2\bar{q}\bar{d})$ , where  $T$  is the number of periods,  $\bar{d}$  is the average demand and  $\bar{q}$  is the average number of pieces required to represent the production cost function. Note that this pseudopolynomial time algorithm is based on the assumption that the demand is an integral value.

For the stochastic version problem, although Ahmed et al. [7] showed that the W-W property does not hold for the stochastic lot-sizing problems several modified W-W properties have been presented for different versions of the stochastic lot-sizing problems: Guan and Miller [8] studied the stochastic uncapacitated lot-sizing problems with uncertain parameters and introduced the production path property for the optimal solution. Further, the production path property was extended to the stochastic capacitated lot-sizing problems with backlogging in Guan and Miller [8] and the dynamic programming algorithms based on this property have been presented. For the stochastic lot-sizing problems with random lead times, Huang and Küçükyavuz [9] presented the Semi-Wagner-Whitin property under assumption that no order crosses in time.

Besides the algorithms based on the extended W-W properties, Lulli and Sen [10] and Ahmed et al. [7] have presented branch-and-price and branch-and-bound algorithms, respectively to solve the proposed multistage integer programming. Although such branch-and-bound-based (B&B-based) methods have broad application prospects for general integer programming, special structure properties of the stochastic lot-sizing problems have not been explored in order to design customized algorithms, and only computational results for small-size stochastic batch-sizing problems have been reported in Lulli and Sen [10]. Other heuristic methods, such as the fix and relax, have also been redesigned to solve particular stochastic lot-sizing problems, see [1, 11]. We also refer the reader to the recent literature review of [12–14].

To the best of our knowledge, little research has been reported on the stochastic lot-sizing problems with incremental quantity discount (SLP-IQD). However as it is reported in the survey by Munson and Rosenblatt [15], 83% of the buyers received quantity discounts

for most of the items they purchased and 37% of orders involved either the offer or receipt of some incremental quantity discounts; thus, the study on quantity discount is of great importance in practice. In this paper, a multistage stochastic mixed-integer programming model is established and under decreasing unit order price assumption, an equivalent relaxed formulation is obtained. The reformulation provides the possibility of extending the production path property for optimal solution of SLP-IQD. The extended production path property is not only a direct extension to the case with incremental quantity discount, but also provides a more accurate characterization for the optimal solution. Then, a backward dynamic programming algorithm has been developed. Although it can obtain optimal solutions in polynomial time in terms of the number of nodes, it has an exponential computational complexity in terms of the number of stages. Thus, a new rolling horizon heuristic which makes use of the extended production path property and has flexible parameters settings is presented in order to balance the desired solution quality and run time. Numerical experiments have been implemented to explore the proper parameters settings and validate the effectiveness of the proposed algorithms by comparison with the CPLEX 11.1 solver and other heuristics.

The remainder of the paper is organized as follows. In Section 2, we first introduce the deterministic formulation and then formulate the general multistage stochastic mixed integer model for the stochastic uncapacitated lot-sizing problems with incremental quantity discount (SULP-IQD). An equivalent reformulation is proposed under the decreasing unit order price assumption. In Section 3, the extended production path property is proven and a backward dynamic programming algorithm and a rolling horizon heuristic with flexible parameters settings are developed. Computational results are reported and discussed in Section 4. Section 5 presents conclusions.

## 2. Mathematical Model

### 2.1. Deterministic Lot-Sizing Problems with Incremental Quantity Discount

First, we will establish an mathematical model for the deterministic uncapacitated lot-sizing problems with incremental quantity discount (DULS-IQD). Considering a planning horizon of  $T$  time periods (stages), at each period  $t$ , the nonnegative demand  $d_t$ , variable inventory holding cost  $h_t$ , the variable setup cost  $c_t$ , and piecewise order cost function  $f_t(\cdot)$  at period  $t \in \{1, \dots, T\}$  are given. Variable  $s_t$  denotes the inventory quantity at the end of period  $t$ , and variable  $x_t$  and Boolean variable  $y_t$  denote the order quantity and fixed charge variable at period  $t$ , respectively.

The incremental quantity discount cost structure is given as follows:

$$\text{the unit order cost} = \begin{cases} p_{t,1} & \text{for the first } Q_{t,1} \text{ units,} \\ p_{t,2} & \text{for the next } Q_{t,2} - Q_{t,1} \text{ units,} \\ \vdots & \vdots \\ p_{t,K_t} & \text{for each unit in excess of } Q_{t,K_t-1} \text{ units,} \end{cases} \quad (2.1)$$

where  $K_t$  denotes the number of price intervals. Suppose that  $Q_{t,0} = 0$  and  $Q_{t,K_t} = +\infty$ . The *decreasing unit order price assumption*, that is,  $p_{t,k} > p_{t,k+1}$ ,  $k = 1, \dots, K_t - 1$  with

$0 = Q_{t,0} < Q_{t,1} < \dots < Q_{t,k} < Q_{t,k+1} < \dots < Q_{t,K_t} = +\infty$  always naturally holds in practice. The piecewise order cost function is given as

$$f_t(x_t) = \begin{cases} p_{t,1}x_t, & \text{if } Q_{t,0} \leq x_t < Q_{t,1}, \\ \sum_{j=1}^{k-1} \{(p_{t,j} - p_{t,j+1})Q_{t,j}\} + p_{t,k}x_t, & \text{if } Q_{t,k-1} \leq x_t < Q_{t,k}, \quad k = 2, \dots, K_t. \end{cases} \quad (2.2)$$

Thus, the DULS-IQD can be formulated as

$$\min : \sum_{t=1}^T \{f_t(x_t) + h_t s_t + c_t y_t\} \quad (2.3)$$

subject to

$$s_{t-1} + x_t - d_t = s_t, \quad t = 1, \dots, T, \quad (2.4)$$

$$x_t \leq M y_t, \quad t = 1, \dots, T, \quad (2.5)$$

$$x_t \geq 0, \quad s_t \geq 0, \quad s_0 = 0 \quad t = 1, \dots, T, \quad (2.6)$$

$$y_t \in \{0, 1\}, \quad t = 1, \dots, T. \quad (2.7)$$

The objective function (2.3) minimizes the sum of inventory cost, setup cost and piecewise order cost. Constraints (2.4) guarantee that the dynamic demands in every period are met. Constraints (2.5) are the capacity constraints, and here we assume that  $M$  is a sufficiently large upper bound on  $x_t$ . Constraints (2.6) ensure that there is no backlogging and order quantity variables are nonnegative. Without loss of generality, we suppose that there is no initial inventory. Setup variable  $y_t$  is defined as a binary variable in constraints (2.7).

## 2.2. Stochastic Lot-Sizing Problems with Incremental Quantity Discount

In this subsection, a stochastic model is established by scenario analysis approach. We assume that the problem parameters follow discrete time stochastic processes with a finite probability space and evolve as a multistage stochastic scenario tree. Let node  $n$  at stage  $t$  ( $t = 1, \dots, T$ ) represent all the realization of the system leading up to and including stage  $t$ , and set  $N$  denotes the set of all nodes. Let set  $L$  denote the set of leaf nodes. Thus a scenario represents a path from the root node (indexed as  $n = 0$ ) to a leaf node. The related notation is given as follows:

$a(n)$ : parent of node  $n$ ,

$t(n)$ : time stage of node  $n$ ,  $t(n) \in \{1, \dots, T\}$ ,

$D(n)$ : the set of all descendants of node  $n$  and  $n \in D(n)$ ,

$P(n, m)$ : the set of nodes on a path from node  $n$  to node  $m$ , where  $m \in D(n)$ , and we assume that  $P(n, m)$  includes  $n$  and  $m$ ,

$C(n)$ : children nodes of node  $n$ , that is,  $C(n) = \{m \in N : a(m) = n\}$ ,

$\theta_n$ : the likelihood (probability) assigned to node  $n$ ,  $\theta_n > 0$ .

Here using the above notation, the deterministic model in the above subsection can be extended to the stochastic environments by replacing the stage index  $t \in \{1, \dots, T\}$  with node index  $n \in N$ . The multistage stochastic problem formulation, denoted by SP, is given as follows:

$$(SP) \min : \sum_{n \in N} \{\theta_n \{f_n(x_n) + h_n s_n + c_n y_n\}\} \quad (2.8)$$

subject to

$$\begin{aligned} s_{a(n)} + x_n - d_n &= s_n, \quad n \in N, \\ x_n &\leq M y_n, \quad n \in N, \\ x_n &\geq 0, \quad s_n \geq 0, \quad s_{a(0)} = 0, \quad n \in N, \\ y_n &\in \{0, 1\}, \quad n \in N. \end{aligned} \quad (2.9)$$

An equivalent mixed integer programming formulation can be easily obtained by introducing auxiliary order quantity variables and corresponding Boolean variables. A group of variables are assigned for each order quantity interval as follows:

$$\begin{aligned} x_{n,k} &\geq 0, \quad n \in N, \quad k = 1, \dots, K_n, \\ y_{n,k} &= \begin{cases} 0, & x_{n,k} = 0, \\ 1, & Q_{n,k-1} < x_{n,k} \leq Q_{n,k}. \end{cases} \end{aligned} \quad (2.10)$$

For model brevity, we introduce the following notations:  $\bar{p}_{n,k} \triangleq \theta_n p_{n,k}$ ,  $\bar{h}_n \triangleq \theta_n h_n$ , and  $\bar{c}_{n,k} \triangleq \theta_n \{c_n + \sum_{j=1}^{k-1} \{(p_{n,j} - p_{n,j+1})Q_{n,j}\}\}$ . Note that the setup cost also increases correspondingly in index  $k$ , that is,  $\bar{c}_{n,k} < \bar{c}_{n,k+1}$  for each node  $n \in N$ . An equivalent MIP formulation, ESP1, is given as follows:

$$(ESP1) \min : \sum_{n \in N} \left\{ \sum_{k=1}^{K_n} \{\bar{p}_{n,k} x_{n,k} + \bar{c}_{n,k} y_{n,k}\} + \bar{h}_n s_n \right\} \quad (2.11)$$

subject to

$$s_{a(n)} + \sum_{k=1}^{K_n} x_{n,k} - d_n = s_n, \quad n \in N, \quad (2.12)$$

$$Q_{n,k-1}y_{n,k} \leq x_{n,k} \leq Q_{n,k}y_{n,k}, \quad n \in N, k = 1, \dots, K_n, \quad (2.13)$$

$$\sum_{k=1}^{K_n} y_{n,k} \leq 1, \quad n \in N, \quad (2.14)$$

$$x_{n,k} \geq 0, \quad s_n \geq 0, \quad s_{a(0)} = 0, \quad n \in N, k = 1, \dots, K_n, \quad (2.15)$$

$$y_{n,k} \in \{0, 1\}, \quad n \in N, k = 1, \dots, K_n. \quad (2.16)$$

**Proposition 2.1.** *Formulation ESP1 is equivalent to the original formulation SP.*

*Proof.* Suppose that  $\{x_{n,k}^*, y_{n,k}^*, s_n^* \mid n \in N, k = 1, \dots, K_n\}$  is an optimal solution of ESP1. For each  $n \in N$ , there exists at most one  $k_n \in \{1, \dots, K_n\}$  such that  $y_{n,k_n}^* = 1$  and  $M \geq x_{n,k_n}^* > 0$  by constraints (2.13), (2.14), and the optimality. Note that if  $\sum_{k'=1}^{K_n} x_{n,k'}^* = x_{n,k_n}^* = 0$ , then  $\sum_{k'=1}^{K_n} y_{n,k'}^* = y_{n,k_n}^* = 0$ . Thus, by checking constraints in SP, we construct a corresponding feasible solution of SP as  $\{\bar{x}_n^* = x_{n,k_n}^*, \bar{y}_n^* = y_{n,k_n}^*, \bar{s}_n^* = s_n^* \mid n \in N\}$ . By the definition of  $\bar{p}_{n,k}$ ,  $\bar{c}_{n,k}$  and  $\bar{h}_n$ , the constructed feasible solution of SP has the same objective cost to  $\{x_{n,k}^*, y_{n,k}^*, s_n^* \mid n \in N, k = 1, \dots, K_n\}$ .

For each optimal solution of SP,  $\{x_n^*, y_n^*, s_n^* \mid n \in N\}$ , we define the corresponding solution for ESP1 as follows: for given  $n \in N$ , if  $Q_{n,k-1} < x_n^* \leq Q_{n,k}$ , then  $\{\bar{x}_{n,k}^* = x_n^*, \bar{y}_{n,k}^* = 1, \bar{s}_n^* = s_n^*, \bar{x}_{n,k'}^* = \bar{y}_{n,k'}^* = 0, k' \neq k\}$  and if  $x_n^* = 0$ , then  $\{\bar{x}_{n,k}^* = 0, \bar{y}_{n,k}^* = 0, \bar{s}_n^* = s_n^*, k \in \{1, \dots, K_n\}\}$ . By checking its feasibility and the objective cost in ESP1, we conclude that the constructed solution is also feasible in ESP1 and has same objective cost to  $\{x_n^*, y_n^*, s_n^* \mid n \in N\}$ .  $\square$

Further, by relaxing the constraints on Boolean variables and order quantity variables in (2.13)-(2.14), the following formulation, denoted by ESP2, and proposition can be obtained.

$$(ESP2) \quad \min : \sum_{n \in N} \left\{ \sum_{k=1}^{K_n} \{ \bar{p}_{n,k} x_{n,k} + \bar{c}_{n,k} y_{n,k} \} + \bar{h}_n s_n \right\} \quad (2.17)$$

subject to

$$s_{a(n)} + \sum_{k=1}^{K_n} x_{n,k} - d_n = s_n, \quad n \in N,$$

$$x_{n,k} \leq y_{n,k} M, \quad n \in N, k = 1, \dots, K_n, \quad (2.18)$$

$$x_{n,k} \geq 0, \quad s_n \geq 0, \quad s_{a(0)} = 0 \quad n \in N, k = 1, \dots, K_n,$$

$$y_{n,k} \in \{0, 1\}, \quad n \in N, k = 1, \dots, K_n.$$

**Proposition 2.2.** Under the decreasing unit order price assumption, formulation ESP2 is equivalent to ESP1 and SP.

*Proof.* Because the feasible set of the relaxed formulation ESP2 contains the feasible set of ESP1, the optimal solutions of ESP2 must be proven feasible for ESP1. Let  $\{\tilde{x}_{n,k}^*, \tilde{y}_{n,k}^*, \tilde{s}_n^* \mid n \in N, k = 1, \dots, K_n\}$  be an optimal solution of ESP2. First note that if  $\tilde{y}_{n,k}^* = 1$ , we must have  $\tilde{x}_{n,k}^* > 0$  otherwise it contradicts the optimality since  $\bar{c}_{n,k} > 0$ .

Second, it is asserted that the relaxed constraints  $\sum_{k=1}^{K_n} y_{n,k} \leq 1, n \in N$  always hold for every optimal solution of ESP2. Suppose that  $\tilde{y}_{n,l}^* = \tilde{y}_{n,m}^* = 1, n \in N, 1 \leq l \neq m \leq K_n$ ; thus,  $\tilde{x}_{n,l}^* > 0$  and  $\tilde{x}_{n,m}^* > 0$ . Without loss of generality, suppose that  $m > l$  and  $Q_{n,r-1} \leq \tilde{x}_{n,l}^* + \tilde{x}_{n,m}^* < Q_{n,r}, 1 \leq r \leq K_n$ , then

$$\begin{aligned}
& \bar{c}_{n,r} + \bar{p}_{n,r} (\tilde{x}_{n,l}^* + \tilde{x}_{n,m}^*) \\
&= \theta_n \left\{ c_n + \sum_{j=1}^{r-1} \{(p_{n,j} - p_{n,j+1})Q_{n,j}\} + p_{n,r} (\tilde{x}_{n,l}^* + \tilde{x}_{n,m}^*) \right\} \\
&= \theta_n \left\{ \sum_{j=m}^{r-1} \{(p_{n,j} - p_{n,j+1})Q_{n,j}\} + p_{n,r} (\tilde{x}_{n,l}^* + \tilde{x}_{n,m}^*) - p_{n,m} (\tilde{x}_{n,l}^* + \tilde{x}_{n,m}^*) \right\} \\
&\quad + \bar{c}_{n,m} + \bar{p}_{n,m} (\tilde{x}_{n,l}^* + \tilde{x}_{n,m}^*) \\
&\leq \theta_n \left\{ \sum_{j=m}^{r-1} \{(p_{n,j} - p_{n,j+1})(\tilde{x}_{n,l}^* + \tilde{x}_{n,m}^*)\} + p_{n,r} (\tilde{x}_{n,l}^* + \tilde{x}_{n,m}^*) - p_{n,m} (\tilde{x}_{n,l}^* + \tilde{x}_{n,m}^*) \right\} \\
&\quad + \bar{c}_{n,m} + \bar{p}_{n,m} (\tilde{x}_{n,l}^* + \tilde{x}_{n,m}^*) \\
&= \bar{c}_{n,m} + \bar{p}_{n,m} (\tilde{x}_{n,l}^* + \tilde{x}_{n,m}^*) \\
&< \bar{c}_{n,m} + \bar{p}_{n,m} \tilde{x}_{n,m}^* + \bar{p}_{n,l} \tilde{x}_{n,l}^* \\
&< \bar{c}_{n,m} + \bar{p}_{n,m} \tilde{x}_{n,m}^* + \bar{c}_{n,l} + \bar{p}_{n,l} \tilde{x}_{n,l}^*, 
\end{aligned} \tag{2.19}$$

where  $\sum_{j=m}^{r-1} \{(p_{n,j} - p_{n,j+1})\} = p_{n,m} - p_{n,r}$  in the third inequality.

From the above analysis, a better solution can always be obtained by setting  $\tilde{x}_{n,r}^* = \sum_{j=1}^{K_n} \tilde{x}_{n,j}^*, \tilde{y}_{n,r}^* = 1$ , where  $Q_{n,r-1} \leq \sum_{j=1}^{K_n} \tilde{x}_{n,j}^* \leq Q_{n,r}$ , and  $\tilde{x}_{n,k}^* = 0, \tilde{y}_{n,k}^* = 0, k \neq r$ , which contradicts the optimality.

Third, we prove that the relaxed constraints  $Q_{n,k-1} y_{n,k} < x_{n,k} \leq Q_{n,k} y_{n,k}, n \in N, k = 1, \dots, K_n$  also hold for the optimal solution. Assume that  $\tilde{x}_{n,l}^* > 0$  and  $\tilde{x}_{n,m}^* = 0$  for all  $m \neq l$ , but

the inequality constraint  $Q_{n,l-1} \leq \tilde{x}_{n,l}^* \leq Q_{n,l}$  does not hold. Without loss of generality, suppose that  $Q_{n,r-1} < \tilde{x}_{n,l}^* \leq Q_{n,r} \leq Q_{n,l-1}$  where  $r \leq l - 1$ , then

$$\begin{aligned} \bar{c}_{n,l} + \bar{p}_{n,l} \tilde{x}_{n,l}^* &= \theta_n \left\{ c_n + \sum_{j=1}^{l-1} \{(p_{n,j} - p_{n,j+1})Q_{n,j}\} + p_{n,l} \tilde{x}_{n,l}^* \right\} \\ &= \bar{c}_{n,r} + \bar{p}_{n,r} \tilde{x}_{n,l}^* + \theta_n \left\{ \sum_{j=r}^{l-1} \{(p_{n,j} - p_{n,j+1})Q_{n,j}\} + (p_{n,l} - p_{n,r}) \tilde{x}_{n,l}^* \right\} \quad (2.20) \\ &> \bar{c}_{n,r} + \bar{p}_{n,r} \tilde{x}_{n,l}^*. \end{aligned}$$

Therefore, we reach a contradiction because we obtain a better solution by setting  $\hat{x}_{n,r}^* = \tilde{x}_{n,l}^*$ ,  $\hat{y}_{n,r}^* = 1$ , and  $\hat{x}_{n,k}^* = 0$ ,  $\hat{y}_{n,k}^* = 0$  for  $k \neq r$ . Since the optimal solution of ESP2 satisfies all the constraints of ESP1, and ESP2 is obtained from ESP1 by relaxing its constraints, this implies that ESP2 is equivalent to ESP1 and SP.  $\square$

### 3. Optimality Condition and Algorithms

In this section, we explore the property of the SULS-IDQ and design-efficient algorithms. It is necessary to highlight the differences between the deterministic problems and the stochastic problems. First, Ahmed et al. [7] and Huang and Küçükyavuz [9] have shown that W-W property does not hold for the stochastic version problems. The reason is that one production or positive order, which is made to exactly cover the demand along certain path, will inevitably influence all its descendant nodes. The violation happens when this positive order could not cover some nodes and their inventory level is nonzero. Thus, we develop an extended production-path property for the stochastic problem and design a dynamic programming algorithm. Second, since the number of nodes grows exponentially as the number of time stages increases for stochastic version problems, the traditional dynamic programming methods fail to run efficiently. So, we further design an approximate heuristic based on the proposed property to obtain good enough solutions efficiently in this section.

#### 3.1. Extended Production-Path Property

Formulation ESP2 presented in Section 2 is essential to handle the piecewise linear objective function and extend the production path property to the SULS-IDQ. The following proposition is not only a direct extension of Guan and Miller [8] when incremental quantity discount is provided, but also a more accurate characterization of the optimal solution. It shows that there always exists an optimal solution such that if a positive order is made at node  $n$ , the order quantity exactly covers the demand along the path from node  $n$  to a certain descendant node  $m$ ; moreover, no positive order will be made for these nodes along the path from node  $n$  to the parent node of  $m$ .

**Proposition 3.1** (extended production-path property). *For any instance of SULS-IDQ, there exists an optimal solution  $(x^*, y^*, s^*)$ , such that for each node  $n \in N$ , if  $x_{n,k}^* > 0$ , then there exist  $m \in D(n)$  and  $Q_{n,k-1} < d_{n,m} - s_{a(n)}^* \leq Q_{n,k}$  such that*

- (1)  $x_{n,i}^* = 0, y_{n,i}^* = 0$ , for all  $i = 1, \dots, K_n, i \neq k$ ;
- (2)  $x_{n,k}^* = d_{n,m} - s_{a(n)}^*, y_{n,k}^* = 1$ ;
- (3)  $x_{l,i}^* = 0, y_{l,i}^* = 0$ , for all  $l \in P(n, a(m)) \setminus \{n\}, i = 1, \dots, K_l$ .

Note that under assumption that all lead time is equal to 1 and by similar arguments, the second optimal condition can be regarded as an extension of the Semi-Wagner-Whitin Property in Huang and Küçükyavuz [9]; however, the third optimal condition provides new restrictions which narrow the scope of the expected optimal solutions. In the next section, an improved backward dynamic programming algorithm and a rolling horizon heuristic based on the extended production-path property will be presented.

For any optimal solution of ESP2,  $(x, y, s)$ , if  $x_{n,k} > 0$ , we introduce the following definition:

$$\begin{aligned} N_P(n) &= \left\{ m \mid m \in D(n), \sum_{k=1}^{K_l} x_{l,k} = 0, \forall l \in P(n, a(m)) \setminus \{n\}, \sum_{k=1}^{K_m} x_{m,k} > 0 \right\}, \\ N_{NP}(n) &= \left\{ m \mid m \in D(n) \cap L, \sum_{k=1}^{K_l} x_{l,k} = 0, \forall l \in P(n, m) \setminus \{n\} \right\}, \end{aligned} \quad (3.1)$$

where set  $N_P(n)$  contains the first node with positive order quantity after node  $n$ , and set  $N_{NP}(n)$  contains these leaf nodes from which to node  $n$  no positive order has been placed except node  $n$ . Note that if a certain positive order quantity is properly transferred between node  $n$  and all the nodes in set  $N_P(n)$ , the feasibility can be kept and only the cost for nodes in the following set is changed:

$$\phi(n) = \{m \mid m \in P(n, l), \forall l \in N_P(n) \cup N_{NP}(n)\}. \quad (3.2)$$

*Proof.* First, it is asserted that there exists at least an optimal solution by Weierstrass' theorem since the feasible set is compact (note that  $\sum_{k=1}^{K_n} x_{n,k}$  can be bounded by  $\max_{m \in D(n) \cup L} \{d_{n,m}\}$ ) and the objective function is continuous. Then for any given optimal solution of ESP2,  $(x^*, y^*, s^*)$ , if  $x_{n,k}^* > 0$ , the first part holds from Proposition 2.2. Next, we show that an optimal solution holding the above property can always be constructed from any given optimal solution of ESP2 by adjusting the variable's value.

We scan the optimal solution  $(x^*, y^*, s^*)$  from stage 1 to stage  $T$ . Assume that there exists a node  $n$  with  $x_{n,k}^* > 0$  not holding the property, but for nodes at stage  $t < t(n)$  the conclusion holds, the following approach adjusts variables' value assigned to nodes in  $D(n)$  to satisfy the property without changing variables' value before stage  $t(n)$ . For any feasible solution  $(x, y, s)$  of ESP2, we define the object cost function as

$$\mathcal{F}(x, y, s) = \sum_{n \in N} \left\{ \sum_{k=1}^{K_n} \left\{ \bar{p}_{n,k} x_{n,k} + \bar{c}_{n,k} y_{n,k} \right\} + \bar{h}_n s_n \right\}. \quad (3.3)$$

The objective cost for  $(x^*, y^*, s^*)$  is

$$\begin{aligned} & \mathcal{F}(x^*, y^*, s^*) \\ &= \sum_{m \in N \setminus \phi(n)} \left\{ \sum_{j=1}^{K_m} \left\{ \bar{p}_{m,j} x_{m,j}^* + \bar{c}_{m,j} y_{m,j}^* \right\} + \bar{h}_m s_m^* \right\} + \bar{p}_{n,k} x_{n,k}^* + \bar{c}_{n,k} y_{n,k}^* \\ &\quad + \bar{h}_n s_n^* + \sum_{m \in \phi(n) \setminus \{n\} \cup N_P(n)} \bar{h}_m s_m^* + \sum_{m \in N_P(n)} \left\{ \bar{p}_{m,j_m} x_{m,j_m}^* + \bar{c}_{m,j_m} y_{m,j_m}^* + \bar{h}_m s_m^* \right\}, \end{aligned} \quad (3.4)$$

where  $j_m$  is the unique positive order quantity at node  $m$ ,  $m \in N_P(n)$ , and the node set  $N$  is divided into four subsets:  $\{n\}$ ,  $N_P(n)$ ,  $\phi(n) \setminus \{n\} \cup N_P(n)$ , and  $N \setminus \phi(n)$ . For any node  $m \in \phi(n) \setminus \{n\} \cup N_P(n)$ , no order is placed.

Since  $x_{m,j_m}^* > 0$  for  $m \in N_P(n)$  (by definition) and  $s_m^* > 0$  for  $m \in N_{NP}(n)$  (otherwise it contradicts the assumption), we choose small positive scalar  $\varepsilon$  satisfying

$$0 < \varepsilon \leq x_{n,k}^*, \quad (3.5)$$

$$0 < \varepsilon \leq s_m^*, \quad \text{for } m \in N_{NP}(n), \quad (3.6)$$

$$0 < \varepsilon \leq x_{m,j_m}^*, \quad \text{for } m \in N_P(n), \quad (3.7)$$

such that the following solutions are feasible for ESP2

$$\begin{aligned} \bar{x}_{n,k}^* &= x_{n,k}^* - \varepsilon, & \bar{s}_m^* &= s_m^* - \varepsilon \quad \text{for } m \in \phi(n) \setminus N_P(n), \\ \bar{x}_{m,j_m}^* &= x_{m,j_m}^* + \varepsilon \quad \text{for } n \in N_P(n), \end{aligned} \quad (3.8)$$

$$\begin{aligned} \hat{x}_{n,k}^* &= x_{n,k}^* + \varepsilon, & \hat{s}_m^* &= s_m^* + \varepsilon \quad \text{for } m \in \phi(n) \setminus N_P(n), \\ \hat{x}_{m,j_m}^* &= x_{m,j_m}^* - \varepsilon \quad \text{for } n \in N_P(n) \end{aligned} \quad (3.9)$$

and keep the value of other variables unchanged. For given  $\varepsilon$  satisfying constraints (3.5)–(3.7), the objective costs of  $(\bar{x}^*, \bar{y}^*, \bar{s}^*)$  and  $(\hat{x}^*, \hat{y}^*, \hat{s}^*)$  are

$$\begin{aligned} & \mathcal{F}(\bar{x}^*, \bar{y}^*, \bar{s}^*) \\ &= \sum_{m \in N \setminus \phi(n)} \left\{ \sum_{j=1}^{K_m} \left\{ \bar{p}_{m,j} x_{m,j}^* + \bar{c}_{m,j} y_{m,j}^* \right\} + \bar{h}_m s_m^* \right\} + \bar{p}_{n,k} (x_{n,k}^* - \varepsilon) + \bar{c}_{n,k} y_{n,k}^* + \bar{h}_n (s_n^* - \varepsilon) \\ &\quad + \sum_{m \in \phi(n) \setminus \{n\} \cup N_P(n)} \bar{h}_m (s_m^* - \varepsilon) + \sum_{m \in N_P(n)} \left\{ \bar{p}_{m,j_m} (x_{m,j_m}^* + \varepsilon) + \bar{c}_{m,j_m} y_{m,j_m}^* + \bar{h}_m s_m^* \right\} \\ &= \mathcal{F}(x^*, y^*, s^*) - \bar{p}_{n,k} \varepsilon - \bar{h}_n \varepsilon - \sum_{m \in \phi(n) \setminus \{n\} \cup N_P(n)} \bar{h}_m \varepsilon + \sum_{m \in N_P(n)} \bar{p}_{m,j_m} \varepsilon \\ &= \mathcal{F}(x^*, y^*, s^*) - \bar{p}_{n,k} \varepsilon - \sum_{m \in \phi(n) \setminus N_P(n)} \bar{h}_m \varepsilon + \sum_{m \in N_P(n)} \bar{p}_{m,j_m} \varepsilon, \end{aligned}$$

$$\begin{aligned}
& \mathcal{F}(\hat{x}^*, \hat{y}^*, \hat{s}^*) \\
&= \sum_{m \in N \setminus \phi(n)} \left\{ \sum_{j=1}^{K_m} \left\{ \bar{p}_{m,j} x_{m,j}^* + \bar{c}_{m,j} y_{m,j}^* \right\} + \bar{h}_m s_m^* \right\} + \bar{p}_{n,k} (x_{n,k}^* + \varepsilon) + \bar{c}_{n,k} y_{n,k}^* + \bar{h}_n (s_n^* + \varepsilon) \\
&\quad + \sum_{m \in \phi(n) \setminus \{\{n\} \cup N_P(n)\}} \bar{h}_m (s_m^* + \varepsilon) + \sum_{m \in N_P(n)} \left\{ \bar{p}_{m,j_m} (x_{m,j_m}^* - \varepsilon) + \bar{c}_{m,j_m} y_{m,j_m}^* + \bar{h}_m s_m^* \right\} \\
&= \mathcal{F}(x^*, y^*, s^*) + \bar{p}_{n,k} \varepsilon + \bar{h}_n \varepsilon + \sum_{m \in \phi(n) \setminus \{\{n\} \cup N_P(n)\}} \bar{h}_m \varepsilon - \sum_{m \in N_P(n)} \bar{p}_{m,j_m} \varepsilon. \\
&= \mathcal{F}(x^*, y^*, s^*) + \bar{p}_{n,k} \varepsilon + \sum_{m \in \phi(n) \setminus N_P(n)} \bar{h}_m \varepsilon - \sum_{m \in N_P(n)} \bar{p}_{m,j_m} \varepsilon. \\
\end{aligned} \tag{3.10}$$

Note that the first equality comes from the definition of  $(\bar{x}^*, \bar{y}^*, \bar{s}^*)$ ,  $(\hat{x}^*, \hat{y}^*, \hat{s}^*)$ , and  $\mathcal{F}$  function. The second one is obtained by comparing with the value of  $\mathcal{F}(x^*, y^*, s^*)$  and the last one is obtained by rearranging the terms in the former, where  $\bar{h}_n \varepsilon + \sum_{m \in \phi(n) \setminus \{\{n\} \cup N_P(n)\}} \bar{h}_m \varepsilon = \sum_{m \in \phi(n) \setminus N_P(n)} \bar{h}_m \varepsilon$ .

Since  $(x^*, y^*, s^*)$  is an optimal solution of ESP2, thus we must have  $\bar{p}_{n,k} + \sum_{m \in \phi(n) \setminus N_P(n)} \bar{h}_m - \sum_{m \in N_P(n)} \bar{p}_{m,j_m} = 0$  and  $(\bar{x}^*, \bar{y}^*, \bar{s}^*)$ ,  $(\hat{x}^*, \hat{y}^*, \hat{s}^*)$  are also optimal solution for ESP2. Now we increase  $\varepsilon$  so that at least one of the following cases occur.

*Case 1.* If equality in (3.5) holds, then we have eliminated the undesired positive order at node  $n$  and  $(\bar{x}^*, \bar{y}^*, \bar{s}^*)$  will be used to construct an eventual solution by a similar method.

*Case 2.* If equality in (3.6) for a certain  $m$  holds, then there exists  $m \in D(n)$  such that  $\bar{x}_{n,k}^* = d_{n,m} - s_{a(n)}^*$  and  $\bar{x}_{l,k}^* = 0, \bar{y}_{l,k}^* = 0$  for all  $l \in P(n, a(m)) \setminus \{n\}, k = 1, \dots, K_l$ . Next,  $(\bar{x}^*, \bar{y}^*, \bar{s}^*)$  will be used to construct an eventual solution by a similar method.

*Case 3.* If equality in (3.7) for a certain  $m$  holds, then we apply the above analysis to  $(\hat{x}^*, \hat{y}^*, \hat{s}^*)$ . Since there are finite nodes in  $D(n)$ , eventually case 1 or case 2 will occur (in the worst case  $N_P(n) = \emptyset$  after finite steps).

Thus, the optimal solution holding the proposed property can always be constructed after finite steps.  $\square$

### 3.2. Dynamic Programming Algorithm

To recursively calculate the optimal solution, the following functions are introduced as in Guan and Miller [8]:

$H(n, s)$ : value function at node  $n$  when its initial inventory is  $s_{a(n)} = s$ , that is,  $H(n, s) = \min \{ \sum_{m \in D(n)} \{ \sum_{k=1}^{K_m} \{ \bar{p}_{m,k} x_{m,k} + \bar{c}_{m,k} y_{m,k} \} + \bar{h}_m s_m \} \}$  subject to constraints (2.18) and  $s_{a(n)} = s$ :

$H_P(n, s)$ : production value function at  $n$  when its initial inventory is  $s_{a(n)} = s$  and  $\sum_{k=1}^{K_n} x_{n,k} > 0$ :

$H_{NP}(n, s)$ : nonproduction value function at  $n$  when its initial inventory is  $s_{a(n)} = s$  and  $\sum_{k=1}^{K_n} x_{n,k} = 0$ .

```

For each stage  $t = 1$  to  $T$ 
  For each node  $n$  at stage  $t$ 
    For each possible initial inventory  $s_m = d_{0,m} - d_{0,a(n)} \geq 0$ ,  $m \in N$ 
      (if  $t = 1$ , set  $s_m = 0$ )
        step 1: Calculate  $H_P(n, s_m)$  by (3.11)
        step 2: Calculate  $H_{NP}(n, s_m)$  by (3.12) if  $d_{0,m} - d_{0,n} \geq 0$ ,
          otherwise  $H_{NP}(n, s_m) = +\infty$ 
        step 3: Calculate  $H(n)$  by (3.13)
    End For Iteration ( $s_m$ )
  End For Iteration ( $n$ )
End For Iteration ( $t$ )

```

**Algorithm 1:** Dynamic programming algorithm.

From Proposition 3.1, if a positive order quantity is made at node  $n$  when the initial inventory is  $s$ , then there exists a node  $j \in D(n)$ , such that  $x_n = d_{n,j} - s$ . Therefore, the following equations hold:

$$\begin{aligned}
H_P(n, s) &= \min_{j \in D(n): Q_{n,k-1} < d_{n,j} - s \leq Q_{n,k}} \left\{ \bar{c}_{n,k} + \bar{p}_{n,k}(d_{n,j} - s) + h_n(d_{n,j} - d_n) + \sum_{m \in C(n)} H(m, d_{n,j} - d_n) \right\}, \\
&\quad (3.11)
\end{aligned}$$

$$H_{NP}(n, s) = h_n(s - d_n) + \sum_{m \in C(n)} H(m, s - d_n), \quad (3.12)$$

$$H(n, s) = \min\{H_P(n, s), H_{NP}(n, s)\}. \quad (3.13)$$

To obtain the exact optimal solution of SP, it is not necessary to completely characterize the value function  $H(0, s)$ . We only need to calculate its values at possible positive discontinuous points. That is, with zero initial inventory assumption for node  $n \in N \setminus \{0\}$ , we only need to evaluate  $H(n, s)$  for  $s = d_{0,m} - d_{0,a(n)} \geq 0$ ,  $m \in N$  since  $d_{n,m} - d_n = d_{0,m} - d_{0,n}$  in (3.11). Thus, we give the following dynamic programming for SULS-IQD:

Without loss of generality, we assume  $|C(n)| = C$  for all  $n \in N$  in the following analysis. Dynamic programming demonstrates a straight method to obtain the optimal solution. Although Guan and Miller [8] showed that general SULS without incremental quantity discount can be solved in  $O(|V(0)|^2 \max\{\log V(0), C\})$ , the above algorithm runs in exponential time in terms of  $T$  since  $V(0) = (C^T - 1)/(C - 1)$ . The exponential computational time encourages us to develop a more effective algorithm for the problems with large numbers of stages (see Algorithm 1).

### 3.3. Rolling Horizon Heuristic

In dynamic lot-sizing and scheduling problems with a large planning horizon, rolling horizon heuristics have been developed to decompose the original large-scale problem into a set of smaller subproblems. See, for example, [11, 16]. In contrast with the existing heuristics, the

proposed RHH based on the extended production-path property stems from the proposed dynamic programming algorithm. In the heuristic, nonproduction strategy is developed to take advantage of the accurate characterization of the optimal solution, and then look-forward and look-backward heuristic strategies are developed to reduce the computation complexity of the complete enumeration in evaluation of the value function. The computation complexity is also analyzed to demonstrate its advantage for the problems with a large planning horizon.

### (1) Nonproduction Strategy

At each iteration for  $s_m$  in DP algorithm for given  $n$ , if  $m \in D(n) \setminus \{n\}$ , step 1 can be skipped and set  $H(n) = H_{NP}(n, s_m)$  since it violates part 3 of Proposition 3.1. In such case, even if  $H(n) = H_P(n, s_m) < H_{NP}(n, s_m)$ , and there exist optimal solutions with  $\sum_{k=1}^{K_n} x_{n,k} > 0$ , we will not lose the optimality because it is guaranteed that there exists an optimal solution satisfying the extended production path property in Proposition 3.1. The nonproduction strategy can be stated as

$$\text{if } m \in D(n) \setminus \{n\}, \text{ then skip step1 and set } H(n) = H_{NP}(n, s_m) \text{ at step3.} \quad (3.14)$$

### (2) Look-Forward Strategy

At step 1 for given node  $n$  at stage  $t$  and initial inventory  $s_m = d_{0,m} - d_{0,a(n)}$ , (3.11) requires calculating all the positive orders that cover demand from node  $n$  to a certain node  $j \in D(n)$ . These complete enumeration calculations are very time-consuming, thus we define the following look-forward strategy with parameter FT (forward time stage):

$$H_P(n, s) = \min_{j \in D_{FT}(n): Q_{n,k-1} \leq d_{n,j} - s \leq Q_{n,k}} \left\{ \bar{c}_{n,k} + \bar{p}_{n,k}(d_{n,j} - s) + h_n(d_{n,j} - d_n) + \sum_{m \in C(n)} H(m, d_{0,j} - d_{0,n}) \right\}, \quad (3.15)$$

where  $D_{FT}(n) = \{j \mid j \in D(n) \text{ and } t(j) - t(n) \leq FT\}$  denotes the set of node  $n$ 's descendants within FT generations. The problem is how to select proper FT. It is obvious that increasing FT will not only improve the quality of solution, but will also increase computation time. Thus, the quality of solution and run time can be balanced by properly selecting FT. The performance of the proposed RHH with different FT settings will be tested by numerical experiments in the next section.

### (3) Look-Backward Strategy

At step 2 for given node  $n$  at stage  $t$  and initial inventory  $s_m = d_{0,m} - d_{0,a(n)}$ , if stage  $t(m)$  is smaller than  $t$ , it is likely  $s_m - d_n = d_{0,m} - d_{0,n} < 0$  and in such cases step 2 can be skipped. For the same reason, if  $t - t(m)$  is larger than a certain value, we can expect  $s_m = d_{0,m} - d_{0,a(n)} < 0$

and this iteration for  $s_m$  can be skipped too. Look-backward strategy with parameter  $BF$  (backward time stage) can be stated as follows:

if  $t - t(m) > BF$ , then skip this iteration for  $s_m$ ;  
(3.16)  
else if  $t - t(m) = BF$ , skip step2 and set  $H(n) = H_P(n, s_m)$  at step 3,

where  $BF$  depends on the distribution of demand. We will set proper  $BF$  based on distribution of demand by numerical experiments in the next section.

In the above strategies, the major modification comes from the iteration for  $s_m$  and calculation of  $H_P(n)$  at step 1 for given node  $n$ . In order to give a brief characterization of the above heuristic rules and analyze its computational complexity, we give the following lemma.

**Lemma 3.2.** *For each node  $n \in N \setminus \{0\}$ , the rolling horizon heuristic with FT and BT only needs to evaluate the following values:*

$$\{H(n, d_{0,m} - d_{0,a(n)}) : d_{0,m} - d_{0,a(n)} \geq 0, m \in RH_{FT,BT}(n)\}, \quad (3.17)$$

where  $RH_{FT,BT}(n) = \cup_{m \in P(0, a(n)), t(n) - t(m) \leq FT + BT} D_{FT}(m)$ .

*Proof.* This lemma is proven by induction from nodes at stage  $t = 1$  to leaf nodes at stage  $T$ . Since positive order must be made at node 0 due to the zero initial inventory assumption, thus for node  $n$  at stage  $t(n) = 2$  by (3.15) possible initial inventory  $s_{a(n)}$  at node  $n$  could only be a certain value in set  $\{d_{0,m} - d_{0,a(n)} : d_{0,m} - d_{0,a(n)} \geq 0, m \in RH_{FT,BT}(n)\} = \{d_{0,m} - d_{0,0} : d_{0,m} - d_{0,0} \geq 0, m \in D_{FT}(0)\}$ . Suppose the lemma holds for node  $a(n)$  at stage  $t$ , where  $2 \leq t \leq T$ , that is, the set of initial inventory for node  $a(n)$  is given by  $\{d_{0,m} - d_{0,a(a(n))} : d_{0,m} - d_{0,a(a(n))} \geq 0, m \in RH_{FT,BT}(a(n))\}$ .

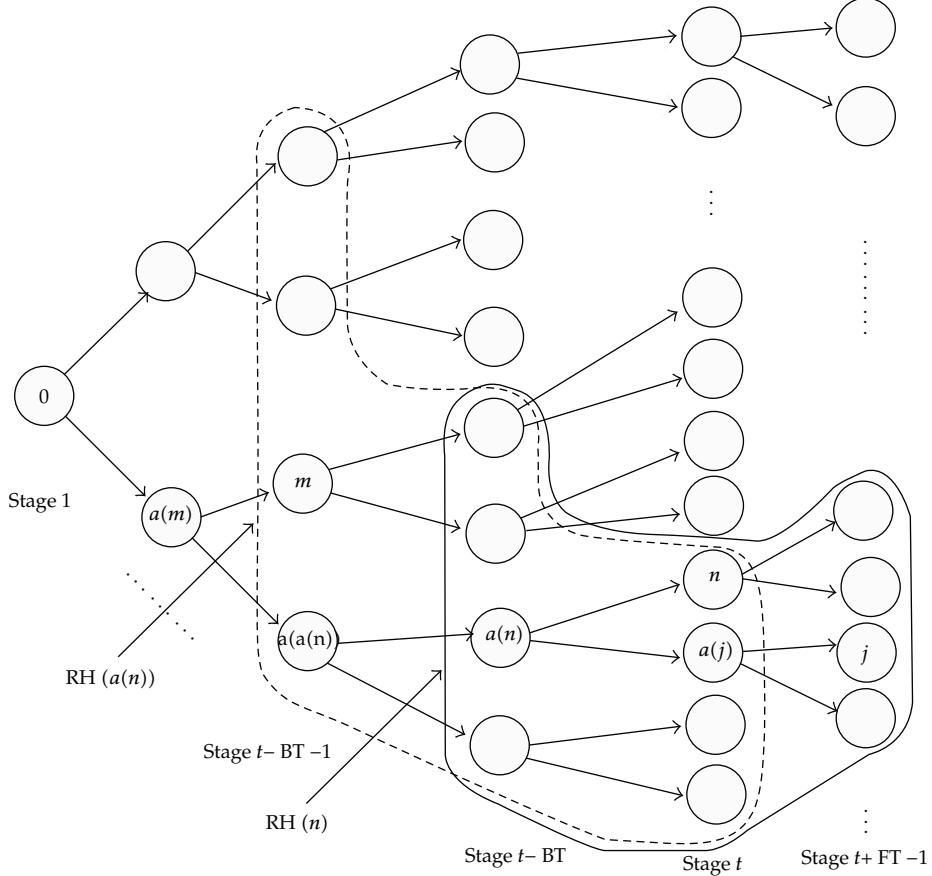
Now consider the possible initial inventory at node  $n$ .

*Case 1.* If a positive order has been made at node  $a(n)$ , by (3.15) in the look-forward strategy, the set of initial inventory at nodes  $n$  is given by  $\{d_{0,m} - d_{0,a(n)} : d_{0,m} - d_{0,a(n)} \geq 0, m \in D_{FT}(a(n))\}$ .

*Case 2.* if no order has been made at node  $a(n)$ , the set of possible initial inventory at node  $n$  is given by  $\{d_{0,m} - d_{0,a(n)} : d_{0,m} - d_{0,a(n)} \geq 0, m \in RH_{FT,BT}(a(n))\}$ . However, note that by look-backward strategy (3.16) we skip the calculation of  $H_{NP}(a(n), d_{0,m} - d_{0,a(n)})$  for those  $m$  such that  $t(a(n)) - t(m) = BF$ ; thus, in this case we only need to consider initial inventory value in set  $\{d_{0,m} - d_{0,a(n)} \geq 0, m \in \cup\{D_{FT}(m) : m \in P(0, a(a(n))), t(a(n)) - t(m) \leq FT + BT - 1\}\} = \{d_{0,m} - d_{0,a(n)} \geq 0, m \in \cup\{D_{FT}(m) : m \in P(0, a(a(n))), t(n) - t(m) \leq FT + BT\}\}$  at node  $n$ .

Combine the above two cases, the conclusion holds for node  $n$ .  $\square$

Figure 1 gives an intuitive example of a balanced scenario tree with  $C = 2$  at each node. For the rolling horizon heuristic with  $FT = 2$  and  $BT = 1$ , given  $RHH_{2,1}(a(n))$  for node  $a(n)$ , consider the initial inventory set for node  $n$ . If a positive order has been made at node  $a(n)$ ,  $\{d_{0,j} - d_{0,a(n)} : d_{0,j} - d_{0,a(n)} \geq 0, j \in D_{FT}(a(n))\}$  and  $t(j) = t(n) + FT - 1$  will be introduced by (3.15). By the look-backward strategy (3.16), we do not need to calculate  $H(n, s_m)$ , where  $t(m) = t(n) - BT - 1$ . Other values will be inherited if no order has been made at node  $a(n)$ .



**Figure 1:** Example of RHH with  $FT = 2$  and  $BT = 1$ .

Note that  $| RH_{FT,BT}(n) |$  is no more than  $(FT + BT)\mathcal{C}^{FT}$  where we assume that  $| C(n) | = \mathcal{C}$  for all  $n \in N$ .

By summarizing the above analysis, the rolling horizon heuristic is given as in Algorithm 2.

Next the computation complexity of RHH is analyzed. For given node  $n$  at stage  $t$ , there exist at most  $| RH_{FT,BT}(n) |$  possible initial inventories  $s_m$ . For each given  $s_m$ , it takes  $O(K_n | C(n) | | D_{FT}(n) |)$  time to complete calculation of  $H_P(n, s_m)$  at step 1. Step 2 and step 3 will be completed in  $O(C(n))$  time. Thus the total run time can be estimated as (assume that  $K_n = K$  for all  $n \in N$ )

$$\begin{aligned} \sum_{n \in N} \{ KC | RH_{FT,BT}(n) | | D_{FT}(n) | \} &\leq KC(FT + BT)\mathcal{C}^{FT} \sum_{n \in N} \{ | D_{FT}(n) | \} \\ &= KC(FT + BT)\mathcal{C}^{FT} \frac{\mathcal{C}^{FT+1} - 1}{\mathcal{C} - 1} | D(0) | \end{aligned}$$

```

For each stage  $t = 1$  to  $T$ 
  For each node  $n$  at stage  $t$ 
    For each initial inventory  $s_m \in \{d_{0,m} - d_{0,a(n)} \geq 0 : m \in \text{RH}_{\text{FT,BT}}(n)\}$ 
      Set  $H_P(n, s_m) = H_{NP}(n, s_m) = +\infty$ ; (if  $t = 1$ , set  $s_m = 0$ )
      step 1: If  $m \in D(n) \setminus \{n\}$ , go to step 3; otherwise calculate  $H_P(n, s_m)$ 
              by (3.15)
      step 2: If  $t - t(m) = \text{BF}$  or  $d_{0,n} - d_{0,m} < 0$ , goto step 3; otherwise
              calculate  $H_{NP}(n, s_m)$  by (3.12)
      step 3: Calculate  $H(n)$  by (3.13)
    End For Iteration ( $s_m$ )
  End For Iteration ( $n$ )
End For Iteration ( $t$ )

```

**Algorithm 2:** Rolling horizon heuristic (FT, BT).

$$\begin{aligned}
&= K(\text{FT} + \text{BT}) \frac{\mathcal{C}^{\text{FT}+1}}{\mathcal{C} - 1} \left\{ \mathcal{C}^{\text{FT}+1} - 1 \right\} |D(0)| \\
&\leq K(\text{FT} + \text{BT}) \mathcal{C}^{2(\text{FT}+1)} |D(0)|.
\end{aligned} \tag{3.18}$$

**Proposition 3.3.** For any instance of SULS-IDQ, the rolling horizon heuristic with parameter FT and BT runs in  $O(K(\text{FT} + \text{BT}) \mathcal{C}^{2(\text{FT}+1)} |D(0)|)$  time.

The above analysis can be applied to the dynamic programming algorithm, thus the total run time for DP is given by

$$\begin{aligned}
\sum_{n \in N} \{K\mathcal{C}|D(0)||D(n)|\} &= K\mathcal{C}|D(0)| \sum_{n \in N} \{|D(n)|\} \\
&= K\mathcal{C}|D(0)| \sum_{t=1}^T \mathcal{C}^{t-1} \left\{ 1 + \mathcal{C}^1 + \dots + \mathcal{C}^{T-t} \right\} \\
&= K|D(0)| \frac{\mathcal{C}}{\mathcal{C} - 1} \left\{ T\mathcal{C}^T - \frac{\mathcal{C}^T - 1}{\mathcal{C} - 1} \right\}.
\end{aligned} \tag{3.19}$$

By comparison of computational complexity in (3.18) and (3.19), we observe that RHH works more efficiently for large  $T$ . Effective parameters settings of RHH will be further explored by numerical experiments in the next section.

## 4. Computational Experiment

In this section, the computational results on both DP and RHH are reported. In the computational analysis, we first concentrate on identifying proper settings of parameters FT and BT for RHH by comparison of its relative error gap and run time with DP's. Then, DP and RHH are compared with the CPLEX solver and other heuristics for the lot-sizing problems with fixed charge.

**Table 1:** Problem instances.

Instance	$C$	$T$	$N$	Int.	Cont.
$G1$	2	10	1023	3069	7161
$G2$	2	11	2047	6141	8188
$G3$	2	12	4095	12285	16380
$G4$	3	7	1093	3279	4372
$G5$	3	8	3280	9840	13120
$G6$	3	9	9841	29523	39564
$G7$	4	6	1365	4095	5460
$G8$	4	7	5461	16383	21844

**Table 2:** Problem parameters.

Parameter	Distribution
$\bar{h}_{n,k}$	Uniform (1, 2)
$\bar{p}_{n,1}$	Uniform (1, 2)
$\bar{c}_{n,1}$	Uniform (3, 5)
$\bar{Q}_{n,k}$	Uniform (5, 15)
$d_{n,k}$	Truncated normal (8, $\sigma^2$ )

#### 4.1. Problem Instance

In order to evaluate performance of the proposed DP and RHH, and explore the proper parameters settings of FT and BT for RHH, 8 groups of problem instances are generated by varying the number of stages from 5 to 12 and the number of branches from 2 to 4. The number of incremental quantity discount intervals is fixed to 3. Table 1 gives the number of branches  $C$ , the number of stages  $T$ , the number of nodes  $N$ , and the number of integer and continuous variables in formulation ESP2 for each group of problem instances. Other parameters are generated randomly and we assume that they are a sequence of i.i.d. random variables which are subject to truncated normal distribution or uniform distribution. We report in Table 2 part of the problem parameters and the variance of  $d_{n,k}$  will be further defined by  $\sigma = 8 * cv$  (coefficient of variation) in the following subsection. Due to decreasing unit order price assumption, the unit order price and setup cost at node  $n$  is generated by  $\bar{p}_{n,k} = (1 - \alpha)\bar{p}_{n,k-1}$  and  $\bar{c}_{n,k} = \bar{c}_{n,k-1} + \alpha\bar{p}_{n,k-1}\bar{Q}_{n,k-1}$ , where  $\alpha$  is the discount factor and we set  $\alpha = 0.05$ . Our experiments are conducted on a workstation clocked at 2.33 GHz and equipped with 11.9 GB of core memory.

#### 4.2. Numerical Results

In order to evaluate the performance of the proposed RHH method with different parameters, we define two different implementations of RHH and the problem parameter  $cv$  varies from 0.05 to 0.25 for each group of problems. RHH<sub>1</sub> with parameters FT =  $[T/3]$  and BT = 1 has been designed to obtain good enough solution in a short time, while RHH<sub>2</sub> with parameters FT =  $[T/2]$  and BT = 2 has been designed to obtain better solutions, where  $[x]$  denotes the smallest integer larger than  $x$ . Table 3 gives the optimal objective function values and CPU

**Table 3:** Performance of RHH with different (FT, BT).

Instance	cv	RHH <sub>1</sub>		RHH <sub>2</sub>		DP	
		RE (%)	CPU (sec)	RE (%)	CPU (sec)	Value	CPU (sec)
G1 – 1	0.05	0.289	0.016	0.0	0.094	12968.28	0.100
G1 – 2	0.15	0.168	0.015	0.0	0.110	12991.93	0.985
G1 – 3	0.25	0.135	0.015	0.0	0.110	13082.23	0.969
G2 – 1	0.05	0.039	0.079	0.0	0.219	25694.23	4.672
G2 – 2	0.15	0.015	0.079	0.0	0.219	26111.22	4.672
G2 – 3	0.25	0.005	0.078	0.0	0.218	26447.11	4.703
G3 – 1	0.05	0.009	0.172	0.0	1.109	51470.76	22.047
G3 – 2	0.15	0.008	0.172	0.0	1.109	51722.50	22.109
G3 – 3	0.25	0.006	0.156	0.0	1.110	52772.90	22.172
G4 – 1	0.05	2.431	0.047	0.509	0.156	8322.42	0.656
G4 – 2	0.15	1.529	0.031	0.195	0.125	8998.02	0.641
G4 – 3	0.25	1.391	0.047	0.158	0.125	9126.63	0.640
G5 – 1	0.05	2.877	0.140	0.635	0.516	24898.40	7.344
G5 – 2	0.15	1.869	0.125	0.409	0.516	26456.16	7.359
G5 – 3	0.25	1.212	0.141	0.190	0.500	27716.36	7.375
G6 – 1	0.05	2.950	0.438	0.214	5.937	74120.22	81.484
G6 – 2	0.15	1.879	0.453	0.064	5.969	79096.31	81.563
G6 – 3	0.25	1.434	0.453	0.054	6.047	82641.24	81.844
G7 – 1	0.05	7.894	0.031	1.270	0.093	7814.61	0.828
G7 – 2	0.15	5.590	0.016	0.783	0.093	8551.74	0.813
G7 – 3	0.25	4.350	0.015	0.435	0.094	9024.62	0.812
G8 – 1	0.05	1.697	0.437	0.247	2.219	30614.77	16.969
G8 – 2	0.15	1.001	0.453	0.075	2.234	33982.89	16.938
G8 – 3	0.25	0.597	0.437	0.053	2.234	36099.06	17.032

time in seconds for DP, and the percentage relative error (RE(%)) and CPU time for RHH<sub>1</sub> and RHH<sub>2</sub>, where

$$RE = \frac{\text{Value of RHH} - \text{Value of DP}}{\text{Value of DP}} \times 100\%. \quad (4.1)$$

For RHH<sub>1</sub>, the computational results show that RHH<sub>1</sub> is able to solve all the instances in the shortest time with no more than 3% relative error except, for instance, G7. The solution quality and run time are further improved when the number of stages  $T$  becomes larger and the worst results of G7 can be interpreted as the look-forward time horizon being too small to obtain near optimal solution. For RHH<sub>2</sub>, the relative error has been further decreased compared with RHH<sub>1</sub>. Although the run time has been increased, RHH<sub>2</sub> still has advantage in run time compared with DP especially for large problem instances. We also note that as the coefficient of variation increases, the solutions quality and run time have been constantly improved for all problem instances.

Then in the second experiment, we concentrate on the comparison of solution quality and computation time with standard MIP solver CPLEX (version 11.1) and another heuristic dynamic slope scaling procedure (DSSP). DSSP proposed by Kim and Pardalos [17] is a

**Table 4:** Comparison with DSSP and CPLEX.

Instance	RHH		DSSP		CPLEX		DP	
	RE (%)	CPU (sec)	RE (%)	CPU (sec)	RE (%)	CPU (sec)	Value	CPU (sec)
G1 – 1	0.0	0.094	2.225	1.062	11.881	5.110	11452.16	0.985
G1 – 2	0.0	0.109	2.299	0.907	10.441	5.063	11653.62	0.984
G1 – 3	0.0	0.094	2.581	1.062	9.555	5.015	11685.286	0.984
G2 – 1	0.0	0.516	1.986	2.172	0.619	25.750	22602.25	4.719
G2 – 2	0.0	0.516	2.583	2.438	3.800	24.093	23408.59	4.719
G2 – 3	0.0	0.500	3.040	2.094	6.103	24.110	23259.73	4.719
G3 – 1	0.0	1.140	1.947	4.578	0.645	111.891	45820.49	22.187
G3 – 2	0.0	1.125	2.845	4.250	1.162	114.657	46665.20	22.172
G3 – 3	0.0	1.125	3.104	4.406	0.858	113.641	47492.82	22.218
G4 – 1	0.704	0.140	0.721	1.015	3.744	3.391	7285.71	0.656
G4 – 2	0.110	0.140	0.893	0.828	15.302	3.375	7916.02	0.656
G4 – 3	0.035	0.140	1.089	0.735	13.440	3.297	8279.95	0.641
G5 – 1	0.867	0.515	0.484	3.313	3.334	37.516	21619.82	7.453
G5 – 2	0.174	0.500	0.632	2.796	1.199	38.265	23767.47	7.422
G5 – 3	0.198	0.516	1.163	3.110	0.814	37.703	24543.10	7.438
G6 – 1	0.202	6.063	0.401	8.781	0.800	414.046	65133.43	82.640
G6 – 2	0.068	6.109	0.747	11.453	1.879	412.860	70243.82	82.375
G6 – 3	0.022	6.110	1.034	11.157	0.843	419.937	74203.32	82.563
G7 – 1	0.201	0.360	0.716	1.078	4.300	4.250	6681.52	0.828
G7 – 2	0.0	0.359	0.889	0.859	1.742	4.172	7480.93	0.812
G7 – 3	0.0	0.344	1.368	1.188	1.188	4.2347	7980.60	0.828
G8 – 1	0.367	2.204	0.429	4.578	3.918	86.125	26885.65	17.265
G8 – 2	0.082	2.172	0.579	4.812	0.741	87.125	30213.60	17.234
G8 – 3	0.0	2.171	1.060	5.594	1.111	89.468	32503.12	17.157

newly developed effective heuristic algorithm that provides good-quality solution to the concave piecewise linear optimization problem and among the heuristics it works well in practice [18]. The initial solutions and updating schemes are implemented in accordance with the recommendations in [18] and the stopping criterion is given as follows: if  $\|x^k - x^{k-1}\| \leq \varepsilon$  or the iteration reaches MaxIteration, then it stops, where MaxIteration = 20. To compare the solution quality for given time limit, we set the time limit of CPLEX solver to 5 time of the total run time of DP. We report in Table 4 the percentage relative error and CPU time of RHH (its parameters are set according to RHH<sub>2</sub>), DSSP and CPLEX and the optimal objective function value and CPU time of DP. Table 4 shows that the proposed RHH outperforms the DSSP in both the quality of solution and run time for almost all the test instances, and the CPLEX solver fails to find optimal solution within the given time.

In summary, the proposed DP can solve the SULP-IQD efficiently compared with the standard CPLEX solver and by properly setting the parameters, we obtain effective RHH which outperforms the DSSP heuristic for the tested instances. The computational results also show that RHH performs better for problem instances with a larger number of stages and high coefficient of variation.

## 5. Conclusion

In this paper, we study the stochastic uncapacitated lot-sizing problems with incremental quantity discount where the uncertain parameters are supposed to evolve as discrete time stochastic processes. First, we establish the original stochastic formulation by scenario analysis approach. Another two formulations are built by introducing auxiliary variables and relaxing certain constraints. Then, it is proven that under the decreasing unit order price assumption, the relaxed formulation is equivalent to the original one. Based on this reformulation, the extended production-path property is presented for the SULP-IQD and it enhances the ability to further refine the desired optimal solution by providing a more accurate characterization.

To obtain the exact optimal solution, a dynamic programming algorithm is developed. Although the dynamic programming algorithm has the polynomial-time computational complexity in terms of the number of nodes, it runs exponentially in terms of the number of stages. Thus, a new rolling horizon heuristic is further designed which contains three types of strategies to reduce the computational time. The nonproduction strategy is designed based on the accurate characterization of the optimal solution, and the look-forward and look-backward strategy is developed to overcome the complete enumeration calculations in the production and nonproduction value function. Numerical experiments are carried out to identify proper parameters settings of the proposed RHH and to evaluate the performance of the proposed algorithms by comparison with the CPLEX solver and DSSP heuristic. The computational results of a large group of problem instances with different parameters setting suggest that DP outperforms the CPLEX solver in run time required for obtaining optimal solution and the proposed RHH demonstrates satisfactory run time and solution quality compared with DSSP heuristic; moreover, as the computational complexity analysis suggests, the performance of RHH is better for problems with a greater number of stages.

Since the main difficulties for the stochastic lot-sizing problems stem from the setup cost and uncertain parameters, it will be an area of future research to analyze the properties of the problem and present effective algorithms for the stochastic lot-sizing problems with complex setup requirements, such as setup carryovers by Buschkuhl et al. [14] and sequence-dependent setup costs by Beraldi et al. [1].

## Acknowledgments

The paper is supported by the National Nature Science Foundation of China (NSFC no. 60874071 and 60834004) and Research Fund for the Doctoral Program of Higher Education of China (RFDP no. 20090002110035).

## References

- [1] P. Beraldi, G. Ghiani, A. Grieco, and E. Guerriero, "Fix and relax heuristic for a stochastic lot-sizing problem," *Computational Optimization and Applications*, vol. 33, no. 2-3, pp. 303–318, 2006.
- [2] H. M. Wagner and T. M. Whitin, "Dynamic version of the economic lot size model," *Management Science*, vol. 5, pp. 89–96, 1958.
- [3] J. Hu and C. L. Munson, "Dynamic demand lot-sizing rules for incremental quantity discounts," *Journal of the Operational Research Society*, vol. 53, no. 8, pp. 855–863, 2002.
- [4] L. M. A. Chan, A. Muriel, Z. J. Shen, and D. S. Levi, "On the effectiveness of zero-inventory-ordering policies for the economic lot-sizing model with a class of piecewise linear cost structures," *Operations Research*, vol. 50, no. 6, pp. 1058–1067, 2002.

- [5] A. Federgruen and C. Y. Lee, "The dynamic lot size model with quantity discount," *Naval Research Logistics*, vol. 37, no. 5, pp. 707–713, 1990.
- [6] D. X. Shaw and A. P. M. Wagelmans, "An algorithm for single-item capacitated economic lot sizing with piecewise linear production costs and general holding costs," *Management Science*, vol. 44, no. 6, pp. 831–838, 1998.
- [7] S. Ahmed, A. J. King, and G. Parija, "A multi-stage stochastic integer programming approach for capacity expansion under uncertainty," *Journal of Global Optimization*, vol. 26, no. 1, pp. 3–24, 2003.
- [8] Y. P. Guan and A. J. Miller, "Polynomial-time algorithms for stochastic uncapacitated lot-sizing problems," *Operations Research*, vol. 56, no. 5, pp. 1172–1183, 2008.
- [9] K. Huang and S. Küçükyavuz, "On stochastic lot-sizing problems with random lead times," *Operations Research Letters*, vol. 36, no. 3, pp. 303–308, 2008.
- [10] G. Lulli and S. Sen, "A branch-and-price algorithm for multistage stochastic integer programming with application to stochastic batch-sizing problems," *Management Science*, vol. 50, no. 6, pp. 786–796, 2004.
- [11] P. Beraldi, G. Ghiani, E. Guerriero, and A. Grieco, "Scenario-based planning for lot-sizing and scheduling with uncertain processing times," *International Journal of Production Economics*, vol. 101, no. 1, pp. 140–149, 2006.
- [12] D. Quadt and H. Kuhn, "Capacitated lot-sizing with extensions: a review," *A Quarterly Journal of Operations Research*, vol. 6, no. 1, pp. 61–83, 2008.
- [13] J. Raf and D. Zeger, "Modeling industrial lot sizing problems: a review," *International Journal of Production Research*, vol. 46, no. 6, pp. 1619–1643, 2008.
- [14] L. Buschkuhl, F. Sahling, S. Helber, and H. Tempelmeier, "Dynamic capacitated lot-sizing problems: a classification and review of solution approaches," *OR Spectrum*, vol. 32, no. 2, pp. 231–261, 2010.
- [15] C. L. Munson and M. J. Rosenblatt, "Theories and realities of quantity discounts: an exploratory study," *Production and Operations Management*, vol. 7, no. 4, pp. 352–369, 1998.
- [16] P. Beraldi, G. Ghiani, A. Grieco, and E. Guerriero, "Rolling-horizon and fix-and-relax heuristics for the parallel machine lot-sizing and scheduling problem with sequence-dependent set-up costs," *Computers and Operations Research*, vol. 35, no. 11, pp. 3644–3656, 2008.
- [17] D. Kim and P. M. Pardalos, "A solution approach to the fixed charge network flow problem using a dynamic slope scaling procedure," *Operations Research Letters*, vol. 24, no. 4, pp. 195–203, 1999.
- [18] S. Lawphongpanich, "Dynamic slope scaling procedure and lagrangian relaxation with subproblem approximation," *Journal of Global Optimization*, vol. 35, no. 1, pp. 121–130, 2006.

*Research Article*

## **Sparse Signal Recovery via ECME Thresholding Pursuits**

**Heping Song<sup>1</sup> and Guoli Wang<sup>2</sup>**

<sup>1</sup> School of Computer Science and Telecommunication Engineering, Jiangsu University, Zhenjiang 212013, China

<sup>2</sup> School of Information Science and Technology, Sun Yat-Sen University, Guangzhou 510006, China

Correspondence should be addressed to Heping Song, hepingsong@gmail.com

Received 17 February 2012; Revised 24 April 2012; Accepted 8 May 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 H. Song and G. Wang. 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 emerging theory of compressive sensing (CS) provides a new sparse signal processing paradigm for reconstructing sparse signals from the undersampled linear measurements. Recently, numerous algorithms have been developed to solve convex optimization problems for CS sparse signal recovery. However, in some certain circumstances, greedy algorithms exhibit superior performance than convex methods. This paper is a followup to the recent paper of Wang and Yin (2010), who refine BP reconstructions via iterative support detection (ISD). The heuristic idea of ISD was applied to greedy algorithms. We developed two approaches for accelerating the ECME iteration. The described algorithms, named ECME thresholding pursuits (EMTP), introduced two greedy strategies that each iteration detects a support set  $I$  by thresholding the result of the ECME iteration and estimates the reconstructed signal by solving a truncated least-squares problem on the support set  $I$ . Two effective support detection strategies are devised for the sparse signals with components having a fast decaying distribution of nonzero components. The experimental studies are presented to demonstrate that EMTP offers an appealing alternative to state-of-the-art algorithms for sparse signal recovery.

### **1. Introduction**

Sparsity exploiting has recently received a great amount of attention in the applied mathematics and signal processing community. Sparse signal processing algorithms have been developed for various applications. A recent Proceedings of the IEEE special issue on applications of sparse representation and compressive sensing devoted to this topic. Some of the exciting developments addressed in [1–7]. Compressed sensing, also known as compressive sensing, compressive sampling (CS) [8, 9], has been the subject of significant activity in sparse signal processing where one seeks to recover efficiently a sparse unknown signal vector of dimension  $n$  via a much smaller number ( $m$ ) of undersampled linear

measurements. For  $k$ -sparse unknown signal  $x_0 \in \mathbb{R}^n$ , the sparse signal recovery is intimately related to solving an underdetermined system of linear equations  $y = Ax_0$  with sparseness constraint

$$(P_{\ell_0}): \min_x \|x\|_{\ell_0} \quad \text{s.t. } Ax = y, \quad (1.1)$$

where  $x$  is true signal to be recovered,  $\|x\|_{\ell_0}$  is used to denote the number of nonzero components of  $x$ ,  $A \in \mathbb{R}^{m \times n}$  is the measurement matrix, and  $y \in \mathbb{R}^m$  is the measurement vector. The key insight in the pioneering work on CS [8, 9] is to replace  $\ell_0$  by  $\ell_1$  for the  $(P_{\ell_0})$  problem due to nonconvexity and combinatorial effect. In [10], it is the basis pursuit problem

$$(\text{BP}): \min_{x_0} \|x_0\|_{\ell_1} \quad \text{s.t. } Ax = y. \quad (1.2)$$

Hereafter, numerous researchers developed various computational algorithms for sparse signal recovery [11, 12]. Generally, there are three major classes of computational algorithms: convex relaxation, bayesian inference, and greedy pursuit. Convex relaxation replaces the combinatorial problem  $(P_{\ell_0})$  with a convex optimization problem (BP), such as basis pursuit [10], basis pursuit denoising [13], the least absolute shrinkage and selection operator (LASSO) [14], and least angle regression (LARS) [15]. Bayesian inference derives the maximum a posteriori estimator from a sparse prior model, such as sparse bayesian learning (SBL) [16, 17] and bayesian compressive sensing (BCS) [18, 19]. Another popular approach is to use greedy algorithms which iteratively refine a sparse solution by successively selecting one or more elements. These algorithms include matching pursuit (MP) [20], orthogonal matching pursuit (OMP) [21–23], subspace pursuit (SP) [24], compressive sampling matching pursuit (CoSaMP) [25], and iterative thresholding algorithms [26, 27].

Iterative hard thresholding (IHT) [27] is a simple and powerful approach for sparse recovery. Recently, an alternative algorithm has been present to alleviate the convergence speed issue of IHT in [28, 29]. Qiu and Dogandzic [28, 29] derived an expectation conditional maximization either (ECME) iteration from a probabilistic framework based on the Gaussian-distributed signals with unknown variance and proposed an acceleration method, termed double overrelaxation (DORE) thresholding scheme, to improve the convergence speed of the ECME algorithm. In addition, Qiu and Dogandzic [28, 29] further proposed an automatic double overrelaxation (ADORE) thresholding method for conditions that the underlying sparsity level is not available. As in study [30], Wang and Yin presented an iterative support detection (ISD) algorithm to refine the failed reconstructions by thresholding the solution of a truncated (BP) problem.

Inspired by the theoretical and empirical evidence of favorable recovery performance of ECME [29] and ISD [30], we combine ECME [29] and ISD [30] to devise novel sparse signal recovery methods, dubbed as ECME thresholding pursuits (EMTP). EMTP detects a support set  $I$  using an ECME iteration and estimates the reconstructed signal by solving a truncated least-squares problem on the support set  $I$ , and it iterates these two steps for a small number of times. We present two effective support detection strategies (hard thresholding and dynamic thresholding) for the sparse signals with components having a fast-decaying distribution of nonzeros (called fast decaying signals in [30]), which include sparse Gaussian signals, sparse Laplacian signals, and certain power-law decaying signals.

This paper considers the iterative greedy algorithms and abstracts them into two types [31], One-Stage Thresholding (OST) and Two-Stage Thresholding (TST), as discussed in

Sections 2.1 and 2.2. Then, we review the initial work of Iterative Support Detection (ISD) in Section 2.3. In Section 3, we describe the proposed approaches. After that, Section 4 details the experimental results. Finally, we conclude this paper in Section 5.

### 1.1. Notation

We introduce the notation used in this paper.

- (i)  $x^{(t)}$ : the algorithms described in this paper are iterative and the reconstructed signal  $x$  in current iteration  $t$  is denoted as  $x^{(t)}$ . The same convention is used for other vectors and matrices.
- (ii)  $I, A_I$ : index set  $I$ , the matrix  $A_I$  denotes the submatrix of  $A$  containing only those columns of  $A$  with indexes in  $I$ . The same convention is used for vectors.
- (iii)  $[1, n] \setminus I$ : the complement of set  $I$  in set  $\{1, 2, \dots, n\}$ .
- (iv)  $\text{supp}(x)$ : the support set of a vector  $x$ , that is, the index set corresponding to the nonzeros of  $x$ ,  $\text{supp}(x) = \{i : x_i \neq 0\}$ .
- (v)  $H_k(x)$ : the hard thresholding that sets all but the largest in magnitude  $k$  elements of a vector  $x$  to zero.
- (vi)  $|x|, \|x\|_{\ell_p}, x^T$ : the absolute value,  $\ell_p$  norm, and transpose of a vector  $x$ , respectively.
- (vii)  $A^\dagger$ : the Moore-Penrose pseudoinverse of matrix  $A \in \mathbb{R}^{m \times n}$ .  $A^\dagger = A^T(AA^T)^{-1}$  for  $m \leq n$ ;  $A^\dagger = (A^T A)^{-1} A^T$  for  $m \geq n$ .

## 2. Related Works

### 2.1. One-Stage Thresholding

Qiu and Dogandzic derived an expectation conditional maximization either (ECME) iteration based on a probabilistic framework [29]

$$x^{(t)} = H_k \left( x^{(t-1)} + A^T (AA^T)^{-1} (y - Ax^{(t-1)}) \right). \quad (2.1)$$

Note that ECME iteration reduces to IHT step when the measurement matrix  $A$  has orthonormal rows (i.e.,  $(AA^T)$  is the identity matrix). These one-stage thresholding algorithms (e.g., IHT [27] and ECME [29]) are guaranteed to recover sparse signals and converge with limited iterations. However, OST is not the empirical choice for practical applications due to slow convergence. To this end, Qiu and Dogandzic proposed an acceleration method, termed double overrelaxation (DORE) thresholding scheme [28, 29], to improve the convergence speed of the ECME algorithm. DORE utilizes two overrelaxation steps:

$$\begin{aligned} x_1^{(t)} &= x^{(t)} + \alpha_1 (x^{(t)} - x^{(t-1)}), \\ x_2^{(t)} &= x_1^{(t)} + \alpha_2 (x_1^{(t)} - x^{(t-2)}), \end{aligned} \quad (2.2)$$

where  $\alpha_1, \alpha_2$  are the line search parameters. Finally, an additional hard thresholding step

$$x^{(t)} = H_k(x_2^{(t)}) \quad (2.3)$$

ensures that the resulting signal is guaranteed to be  $k$ -sparse. In addition, Qiu and Dogandzic further presented an automatic double overrelaxation (ADORE) thresholding method for conditions that the underlying sparsity level is not available.

## 2.2. Two-Stage Thresholding

The algorithms described in this paper fall into the category of a general family of iterative greedy pursuit algorithms. Following [31], we adopt the name “Two-Stage Thresholding” (TST). Considering a CS recovery problem, the sparse recovery algorithms aim to detect the support set  $I$  and approximate  $y$  using

$$y = A_I x_I. \quad (2.4)$$

Starting with initial solution  $x = 0$ , TST iterates between 2 main steps:

*Step 1* (support detection). Detect the support set  $I$  of the signal  $x$ , that is, select atoms of measurement matrix  $A$  which have been used to generate  $y$ ; in other words, determine active atoms in sparse representation of a signal  $x$ . In some literature, this step also is called basis selection or atom selection.

*Step 2* (signal estimation). Update the signal  $x$  using the least-squares solution on the detected support set  $I$ .  $x_I = \arg \min_z \{ \|y - A_I z\|_2^2, \text{supp}(z) \subseteq I \}, x_{[1,n] \setminus I} = 0$ .

Many algorithms (e.g., Orthogonal Matching Pursuit (OMP) [23], Subspace Pursuit (SP) [24], Compressed Sensing Matching Pursuit (CoSaMP) [25], and Gradient Pursuits (GP) [32]) developed various approaches for Step 1 or Step 2.

## 2.3. Iterative Support Detection

Considering the failed reconstructions of BP, Wang and Yin [30] proposed an algorithmic framework to refine the BP constructions, called iterative support detection (ISD). ISD alternates between two steps: support detection and signal reconstruction. Initialize the detected support  $I = \emptyset$  and set the iteration number  $t = 0$ ; ISD iteratively calls the following steps:

(1) signal reconstruction:

solve the truncated BP with  $T = I^C$ :

$$x^{(t)} = \arg \min_x \|x_T\|_1 \quad \text{s.t. } y = Ax; \quad (2.5)$$

(2) support detection:

detect support set  $I$  using  $x^{(t)}$  as the reference.

The reliability of ISD relies on the support detection. Wang and Yin devised several detection strategies for the sparse signals with components having a fast decaying distribution of nonzero components (called fast decaying signals [30]). One of the detection strategies is based on thresholding (we use ISD defined by (2.6) to refer the implementation algorithm in the following context):

$$I^{(t)} = \left\{ i : |x_i^{(t)}| > \beta^t \max |x^{(t)}| \right\}, \quad \beta \in (0, 1). \quad (2.6)$$

### 3. ECME Thresholding Pursuits

In this section, we describe our proposed approaches, named ECME Thresholding Pursuits (EMTP), that combine OST and TST using the heuristic idea of ISD. The detailed description of the proposed algorithms are presented as follows

*Step 1* (initialization). We have the following.

- Initialize the reconstruction signal  $x^{(0)} = 0$ ,
- initialize residual signal  $r = y$ ,
- initialize support set  $I^{(0)} = \emptyset$ ,
- set the iteration counter  $t = 1$ .

*Step 2* (support detection). We have the following.

Update signal approximation:

$$x^{(t)} = x^{(t-1)} + A^\dagger r^{(t-1)}. \quad (3.1)$$

Detect the support set  $I$ :

- strategy 1: hard thresholding  $I^{(t)} = \text{supp}(H_k(x^{(t)}))$ ;
- strategy 2: dynamic thresholding  $I^{(t)} = \{i : |x_i^{(t)}| > \beta^t \max |x^{(t)}|\}$ .

*Step 3* (signal estimation). We have the following.

Estimate the signal:

$$\begin{aligned} x_{I^{(t)}}^{(t)} &= A_{I^{(t)}}^\dagger y, \\ x_{[1,n] \setminus I^{(t)}}^{(t)} &= 0. \end{aligned} \quad (3.2)$$

Update the residual:

$$r^{(t)} = y - Ax^{(t)}. \quad (3.3)$$

**Input:** Measurement matrix  $A$ , measurements  $y$ , sparsity level  $k$   
**Output:** The reconstructed signal  $x$

- (1) **Initialization:**
- (2)  $t = 1$  //iteration number
- (3)  $x^{(0)} = \mathbf{0}$  //initial signal
- (4)  $r^{(0)} = y$  //initial residual
- (5)  $I^{(0)} = \emptyset$  //initial support set
- (6) **while** halting criterion false **do**
- (7)    $x^{(t)} = H_k(x^{(t-1)} + A^\dagger r^{(t-1)})$
- (8)    $I^{(t)} = \{i : x_i^{(t)} \neq 0\}$
- (9)    $x_{I^{(t)}}^{(t)} = A_{I^{(t)}}^\dagger y$
- (10)    $x_{[1,n] \setminus I^{(t)}}^{(t)} = 0$
- (11)    $r^{(t)} = y - Ax^{(t)}$
- (12)    $t = t + 1$
- (13) **end while**
- (14) **return**  $x$

**Algorithm 1:** EMTP- $k$  algorithm.

**Input:** Measurement matrix  $A$ , measurements  $y$ , thresholding parameter  $\beta$   
**Output:** The reconstructed signal  $x$

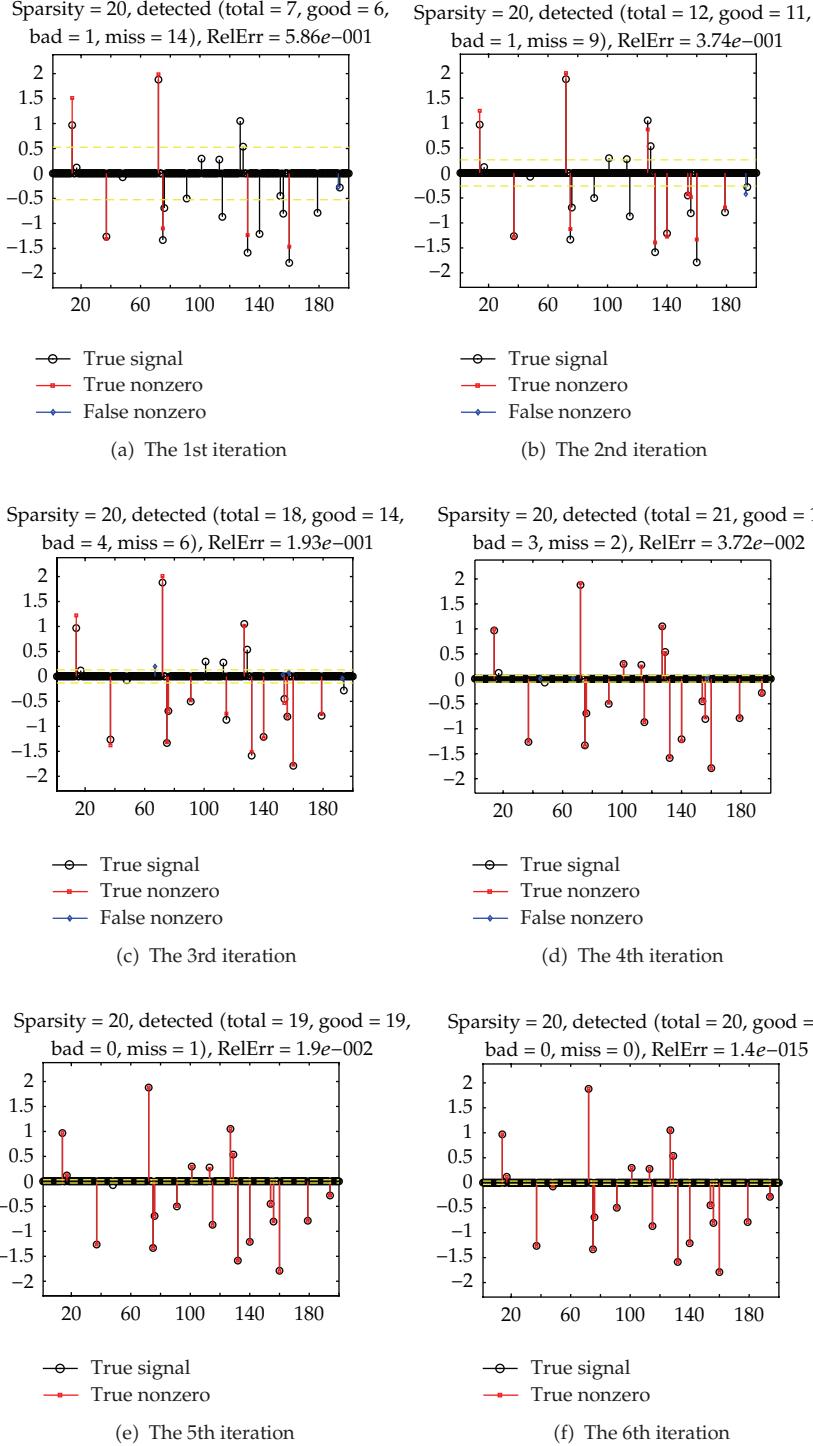
- (1) **Initialization:**
- (2)  $t = 1$  //iteration number
- (3)  $x^{(0)} = \mathbf{0}$  //initial signal
- (4)  $r^{(0)} = y$  //initial residual
- (5)  $I^{(0)} = \emptyset$  //initial support set
- (6) **while** halting criterion false **do**
- (7)    $x^{(t)} = x^{(t-1)} + A^\dagger r^{(t-1)}$
- (8)    $I^{(t)} = \{i : |x_i^{(t)}| > \beta^t \max |x|\}$
- (9)    $x_{I^{(t)}}^{(t)} = A_{I^{(t)}}^\dagger y$
- (10)    $x_{[1,n] \setminus I^{(t)}}^{(t)} = 0$
- (11)    $r^{(t)} = y - Ax^{(t)}$
- (12)    $t = t + 1$
- (13) **end while**
- (14) **return**  $x$

**Algorithm 2:** EMTP- $\beta$  algorithm.

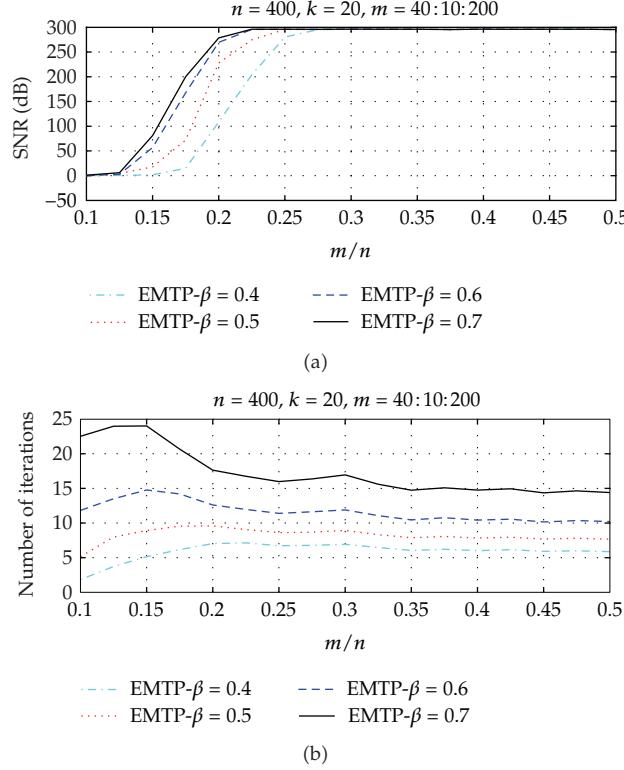
*Step 4* (halting). Check the stopping condition and, if it is not yet time to stop, increment  $t = t + 1$  and return to Step 2. If it is time to stop, the recovered signal  $x$  has nonzero entries in support set  $I^{(t)}$  and corresponding support vector lies in  $x_{I^{(t)}}^{(t)}$ .

We present two thresholding proposals and corresponding algorithm EMTP- $k$  (shown as Algorithm 1) and EMTP- $\beta$  (shown as Algorithm 2), where  $\beta \in (0, 1)$  is the thresholding parameter.

*Remarks.* (1) EMTP updates the leading elements of  $x^{(t)}$  using a least-squares solution on the support set  $I$ . However, DORE updates all the elements of  $x^{(t)}$  using double overrelaxation



**Figure 1:** An EMTP- $\beta$  demo that recovers a Gaussian-distributed sparse vector with  $n = 200, m = 80, k = 20, \beta = 0.5$ .

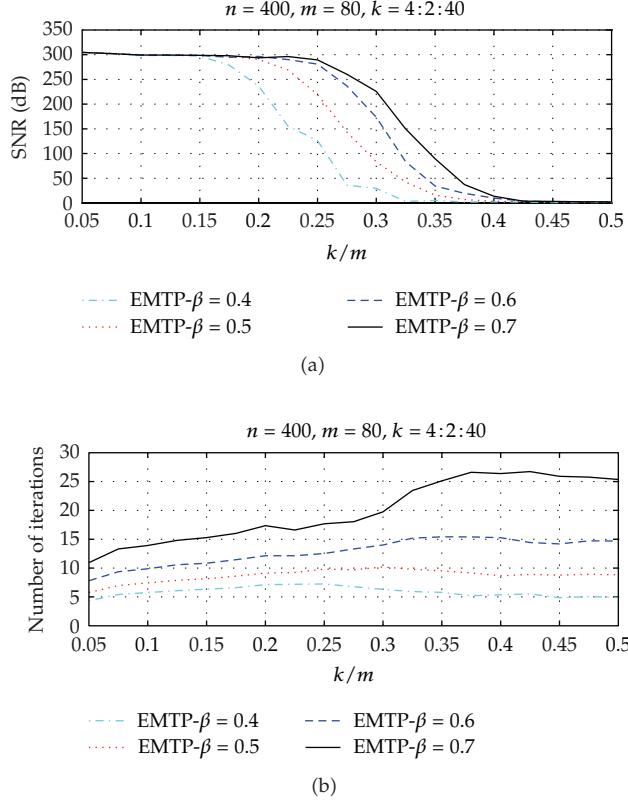


**Figure 2:** Influence of thresholding parameter  $\beta$  for sparse Gaussian signals with  $n = 400$ ,  $k = 20$ : comparisons in terms of SNR (dB) and number of iterations.

steps. Finally, DORE needs a hard thresholding to ensure the new approximation is  $k$ -sparse. EMTP- $\beta$  demands no prior knowledge about the underlying sparsity level  $k$ .

(2) EMTP is a special case of TST and utilizes OST in the support detection stage. It is different from OMP, SP, and CoSaMP. When the ECME iteration reduces to IHT step, that is, the reference for support detection  $x^{(t)} = x^{(t-1)} + A^T r^{(t-1)}$  is the gradient descent step for least-squares, OMP, SP and CoSaMP use the correlation between the residual signal and the atoms of the measurement matrix  $A$  (i.e.,  $A^T r^{(t-1)}$ ) to detect support set.  $A^T r^{(t-1)}$  is the negative gradient for least squares. It is clear that ECME iteration provides the estimation for the underlying sparse signal. We can employ the heuristic idea of ISD to devise various support detection strategies depending on the underlying sparse signal distribution. The dynamic thresholding method can be performed without the sparsity level  $k$ . EMTP directly detects the support of the underlying sparse signal by referencing the ECME iteration while OMP, SP, and CoSaMP augment the support by picking out the leading values of the negative gradient. OMP each step spots one index into the support, so it requires more iterations than EMTP. SP and CoSaMP spot several indexes into the support, so they need an additional step (i.e., orthogonal projection) to make sure the recovered signal is  $k$ -sparse. Like ISD, EMTP has an appealing self-correction capacity. An EMTP demo is presented in Figure 1.

(3) Like other greedy pursuits such as Subspace Pursuit [24] and CoSaMP [25], EMTP- $k$  fixes the cardinality of support set  $I$  and removes previous false detections. However, EMTP- $\beta$  refines the support set  $I$  which is not necessarily increasing and nested over the iterations.



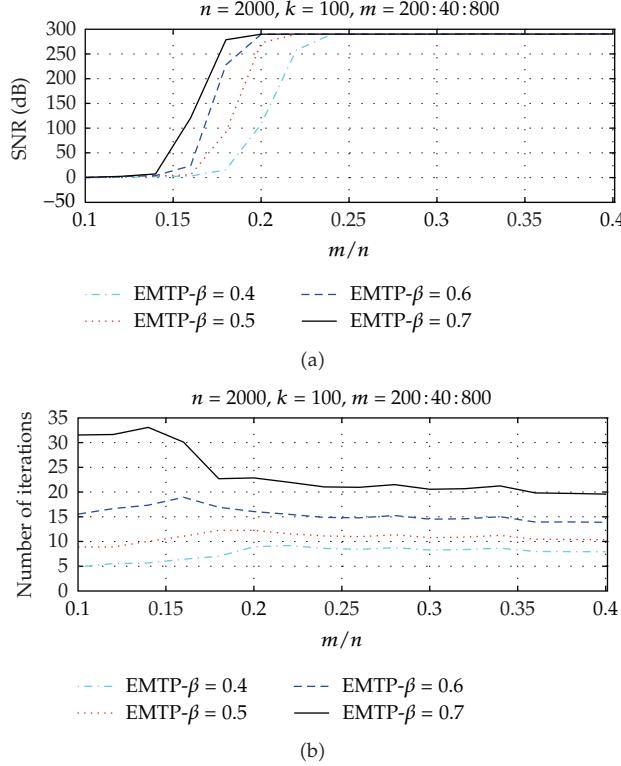
**Figure 3:** Influence of thresholding parameter  $\beta$  for sparse Gaussian signals with  $n = 400$ ,  $m = 80$ : comparisons in terms of SNR (dB) and number of iterations.

(4) The dynamic thresholding strategy used in EMTP- $\beta$  is inspired by ISD [30]. It finds significant nonzeros by comparing a threshold rather than maintaining fixed number ( $k$ ) of items. It is appealing for the conditions that the underlying sparsity level  $k$  is not available.

(5) EMTP resembles ISD [30]. EMTP and ISD have the same idea in support detection step with iterative behavior. However, the support detection step is based on different sparse recovery methods, ECME and BP, respectively. EMTP updates the reconstruction signal using a least-squares solution on the detected support set  $I$ . However, ISD iteratively refines the BP solution on the complement of the detected support set  $I$ .

(6) EMTP- $\beta$  with large  $\beta$  obtains high-quality reconstruction from a small number of measurements. However, because support set  $I$  grows slowly, EMTP- $\beta$  takes a larger number of iterations (to be discussed in Section 4). EMTP- $k$  wrongly detected elements can often be pruned out in later iterations.

(7) Like ISD (as discussed in [30]), EMTP- $\beta$  only performs well for fast decaying signals. It does not work on sparse signals that decay slowly or have no decay at all (e.g., trinary and binary sparse signals). EMTP- $k$  performs worse for the sparse signals that nonzero elements have similar magnitude [31, 33]. However, we can apply EMTP to non-fast-decaying signals via linear or nonlinear premapping [34, 35].



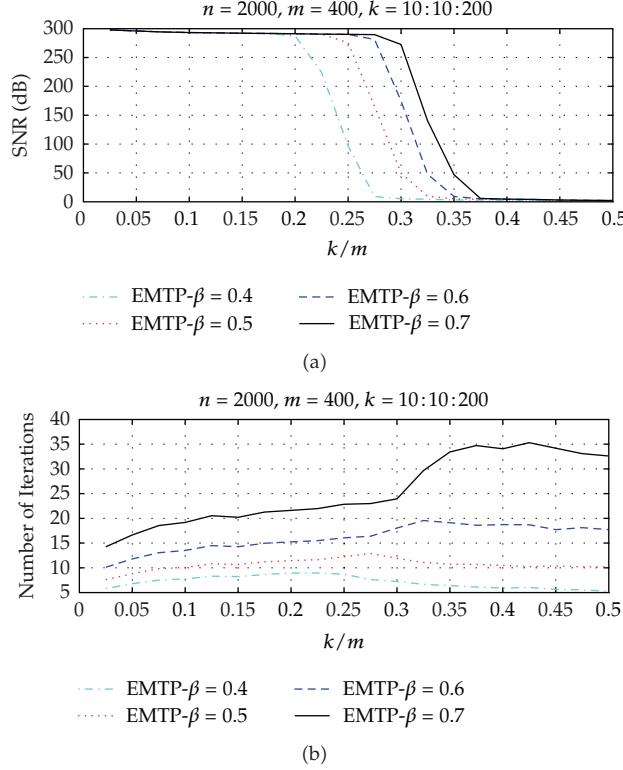
**Figure 4:** Influence of thresholding parameter  $\beta$  for sparse Gaussian signals with  $n = 2000$ ,  $k = 100$ : comparisons in terms of SNR (dB) and number of iterations.

### 3.1. Complexity Analysis

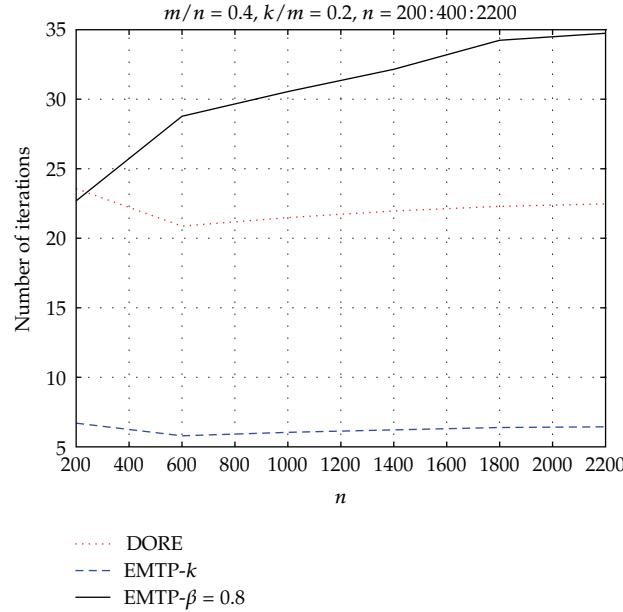
Like DORE algorithm, the basic operation is matrix-vector multiplication and sorting operation. Assume that the pseudoinverse matrix  $A^\dagger$  is precomputed (with the cost of  $\mathcal{O}(m^2n + mn)$ ). Line 7 in Algorithms 1 and 2 for updating the ECME iteration requires  $\mathcal{O}(mn)$ . The computation complexity of thresholding operation (line 8 in Algorithms 1 and 2) is  $\mathcal{O}(n \log k)$ . Line 9 involves the partial least-square solver, costing  $\mathcal{O}(k^2m + km)$ . The step for updating residual requires an additional matrix-vector operation, costing  $\mathcal{O}(km)$ . To summarize, one EMTP iteration costs  $\mathcal{O}(mn + n \log k + k^2m + 2km)$ , which is significantly less than the complexity of one DORE step (with the cost of  $\mathcal{O}(2n \log k + 2m^2n + 3mn)$ ).

## 4. Experimental Results

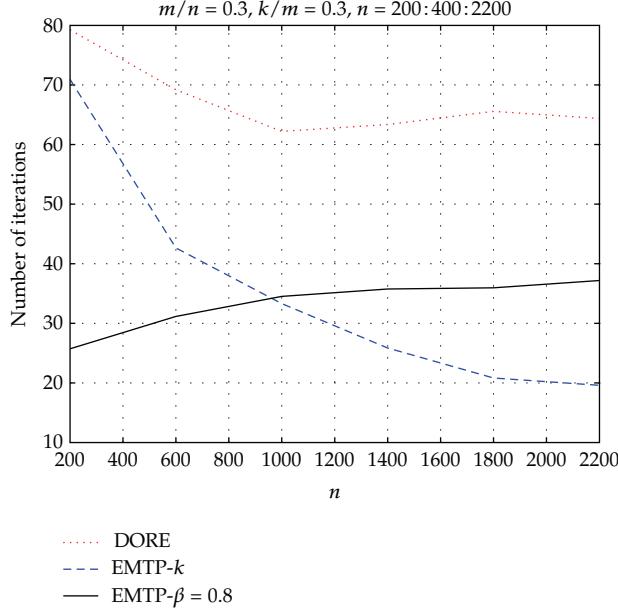
To assess the performance of the proposed approaches, we conduct numerical experiments on computer simulations. All algorithms were implemented and tested in MATLAB v7.6 running on Windows XP with 2.53 GHz Intel Celeron CPU and 2GB of memory. We compared the proposed approaches with the accelerated ADORE/DORE approaches. The code of ADORE/DORE is available on the authors homepage (<http://home.eng.iastate.edu/~ald/DORE.htm>). We initialized  $x$  with zero sparse signal. The search length of ADORE was set to 1. All results were averaged



**Figure 5:** Influence of thresholding parameter  $\beta$  for sparse Gaussian signals with  $n = 2000$ ,  $m = 400$ : comparisons in terms of SNR (dB) and number of iterations.



**Figure 6:** Number of iterations as a function of problem size with fixed ratios of  $m/n = 0.4$ ,  $k/m = 0.2$ ,  $n = 200:400:2200$ .



**Figure 7:** Number of iterations as a function of problem size with fixed ratios of  $m/n = 0.3, k/m = 0.3, n = 200:400:2200$ .

over 100 times Monte Carlo problem instances. We used our unoptimized code (<http://cs-notes.googlecode.com/files/EMTP.zip>). The least-squares solution  $x = \text{argmin}\|y - Ax\|_2^2$  was implemented using MATLAB pseudoinverse function by  $x = \text{pinv}(A)*y$ . The MATLAB code was partially adapted from [30, 33, 36]. In all the experiments, the measurement matrix  $A$  was generated by uniform spherical ensemble (USE), that is, we generate the measurement matrix by sampling each entry independently from the standard norm distribution and then normalize each column to have unit norm. The MATLAB code is given by

```
A = randn(m,n); A = A./repmat(sqrt(sum(A.^2)), [m 1]).
```

The underlying  $k$ -sparse vectors were generated by randomly selecting a support set of size  $k$  and each entry in the support set is sampled uniformly from a specific distribution. The sparse signals were generated in MATLAB by

```
x = zeros(n,1); support = randperm(n); x(support(1:k)) = v;
```

and the nonzeros  $v$  generated by following code.

The sparse Gaussian signals were generated in MATLAB by

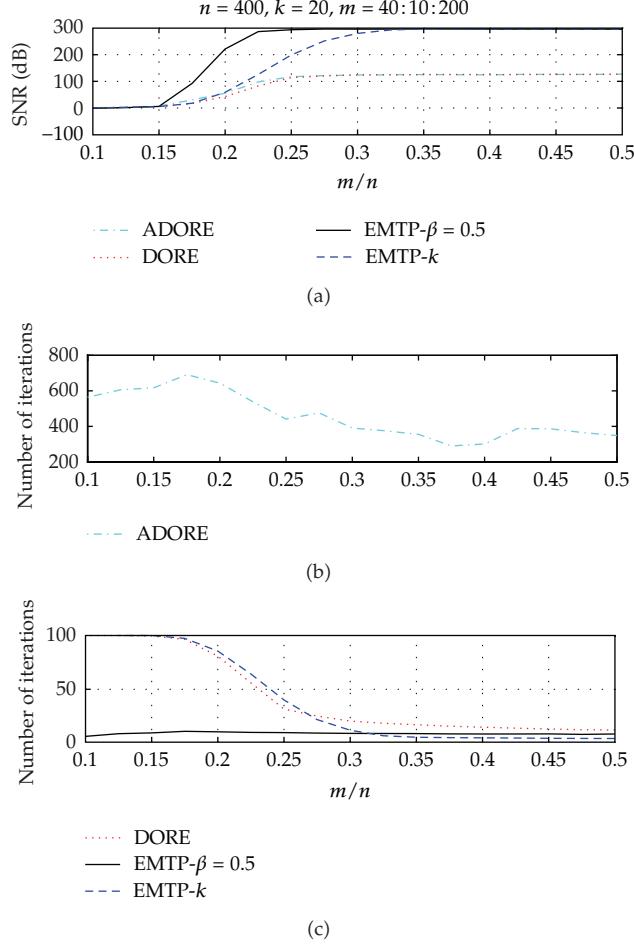
```
v = randn(k,1);
```

The sparse Laplacian signals were generated by

```
z = rand([k,1]);
v = zeros([k,1]);
in = z <= 0.5;
ip = z > 0.5;
v(in) = 1/lambda * log(2 * z(in));
v(ip) = -1/lambda * log(2 * (1 - z(ip))).
```

The power-law decaying signals were generated by

```
v = sign(randn(k,1)).* ((1:k).^\wedge(-1/lambda)').
```



**Figure 8:** Sparse Gaussian signals with  $n = 400, k = 20$ : comparisons in terms of SNR (dB) and number of iterations.

The variable lambda controls the rate of decay. We set `lambda = 10` and `lambda = 0.5` for sparse Laplacian signals and power-law decaying signals, respectively.

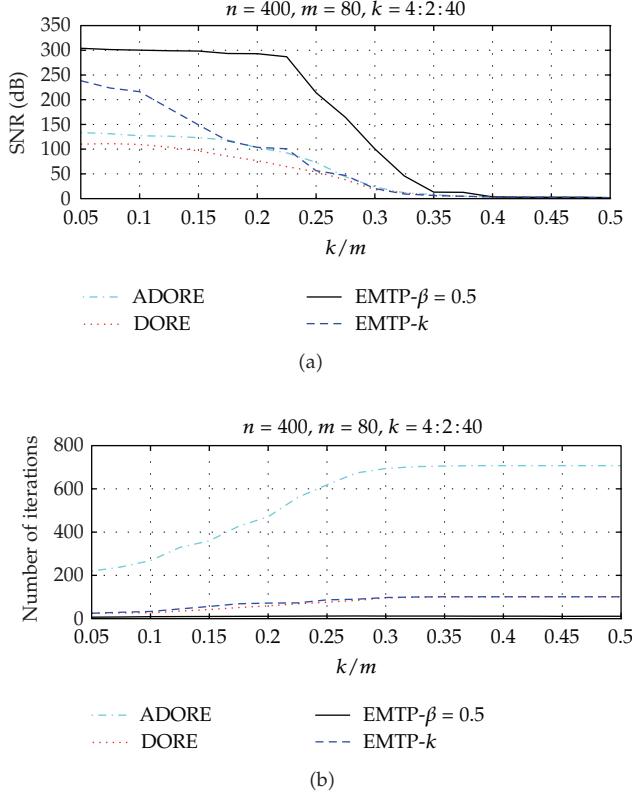
For fair comparison, we stopped iterations once the relative error fell below a certain convergence tolerance or the number of iterations is greater than 100. The convergence tolerance is given by

$$\frac{\|y - r^{(t)}\|_2}{\|y\|_2} \leq 10^{-6}. \quad (4.1)$$

We empirically evaluate reconstruction performance in terms of signal-to-noise ratio (SNR) and number of iterations. The SNR is defined as

$$\text{SNR(dB)} = -20\log_{10} \frac{\|x_0 - x\|_2}{\|x_0\|_2}, \quad (4.2)$$

where  $x_0$  is the true signal and  $x$  is the recovered signal.

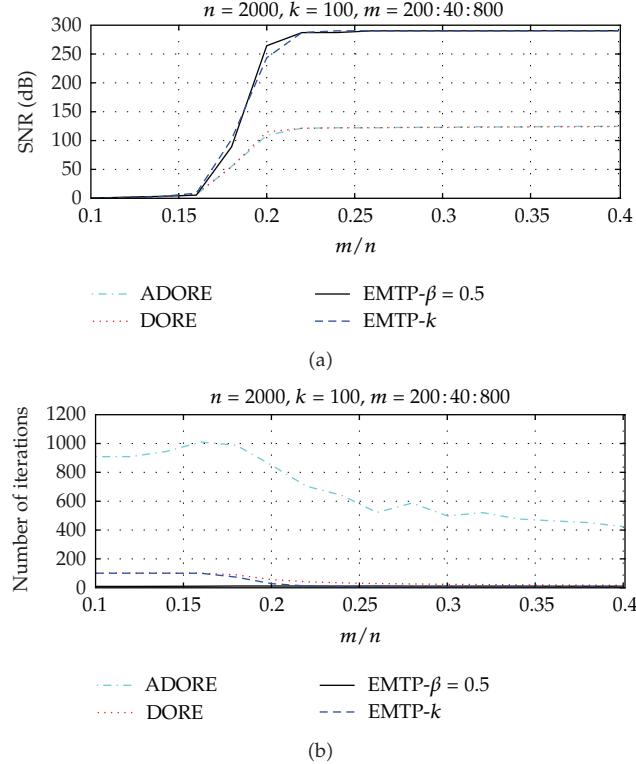


**Figure 9:** Sparse Gaussian signals with  $n = 400$ ,  $m = 80$ : comparisons in terms of SNR (dB) and number of iterations.

#### 4.1. Influence of Thresholding Parameter

We evaluated the influence of thresholding parameter  $\beta$  by varying  $\beta$  from 0.4 to 0.7. First, we fixed signal dimension  $n = 400$ , the underlying sparsity level  $k = 20$ , and varied the number of measurements. The plot of influence of thresholding parameter  $\beta$  for sparse Gaussian signals was presented in Figure 2. It is clear that larger thresholding parameter achieves better recovery performance with fewer measurements. However, the cost is more iterations. Large  $\beta$  is particularly competitive when the number of measurements is fairly small. It is worth noting that the number of iterations is smaller than the underlying sparsity level  $k$  for exact recovery, especially for large  $k$ . As shown in Figure 2, EMTP- $\beta$  achieves the SNR (dB) around 300, that is, the relative error is almost as low as the double precision. Second, we tested the comparisons by fixing the number of measurements  $m = 80$ , as shown in Figure 3, respectively. Finally, we enlarged the dimension of signals ( $n = 2000$ ), as shown in Figures 4 and 5. The latter tests come to similar conclusions as the first test.

We fixed the ratios of  $m/n$ ,  $k/m$  and present plots cases for the number of iterations as a function of problem size  $n$ . Given sufficient measurements, as shown in Figure 6 with  $m/n = 0.4$ ,  $k/m = 0.2$ ,  $n = 200:400:2200$ , EMTP- $k$  requires significantly fewest iterations and EMTP- $\beta$  requires relatively more iterations than DORE. It is clear that the number of iterations for EMTP- $k$  and DORE is stable to the problem size, and EMTP- $\beta$  needs acceptably a few more iterations with increasing the problem size. For another case, as shown in Figure 7

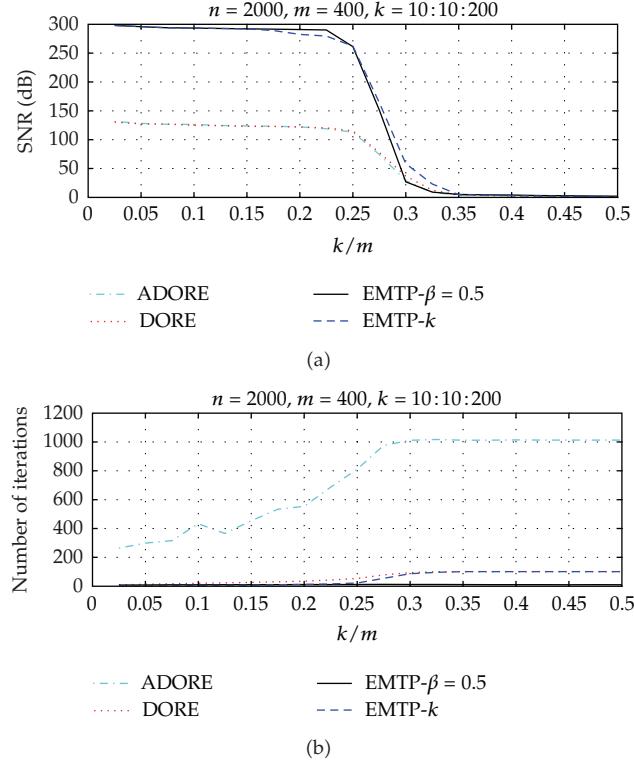


**Figure 10:** Sparse Gaussian signals with  $n = 2000, k = 100$ : comparisons in terms of SNR (dB) and number of iterations.

with  $m/n = 0.3, k/m = 0.3, n = 200:400:2200$ , DORE requires more iterations, EMTP- $k$  needs significantly fewer iterations with increasing the problem size, and EMTP- $\beta$  requires relatively stable iterations.

#### 4.2. Comparisons with the Accelerated OST

We present comparisons in terms of SNR (dB) and number of iterations. We omit the comparisons with  $\ell_1$  minimization methods and some other OST/TST methods, as they have already been investigated in [31, 33]. First, we fixed signal dimension  $n = 400$ , the underlying sparsity level  $k = 20$ , and varied the number of measurements. The result was plotted in Figure 8. Next, we tested the comparisons by fixing the number of measurements  $m = 80$ , as shown in Figure 9. Finally, the test set used larger dimension signals ( $n = 2000$ ). The corresponding results were depicted in Figures 10 and 11. Figures 10, and 11 show the average SNR (dB) (a) and number of iterations (b) for each indeterminacy level  $m/n$ . EMTP achieves significantly larger SNR (dB) than ADORE/DORE. For exact recovery, EMTP obtains the relative error almost as low as the double precision. EMTP- $\beta$  constantly needs small number of iterations and EMTP- $k$  needs the smallest number of iterations for exact recovery. However, ADORE needs hundreds of iterations. Figure 8 was zoomed in for better illustration. Figures 8, 9, 10 and 11 show the average SNR (dB) and number of iterations for each sparsity level  $k/m$ . For larger-dimension signals ( $n = 2000$ ), the results are depicted in

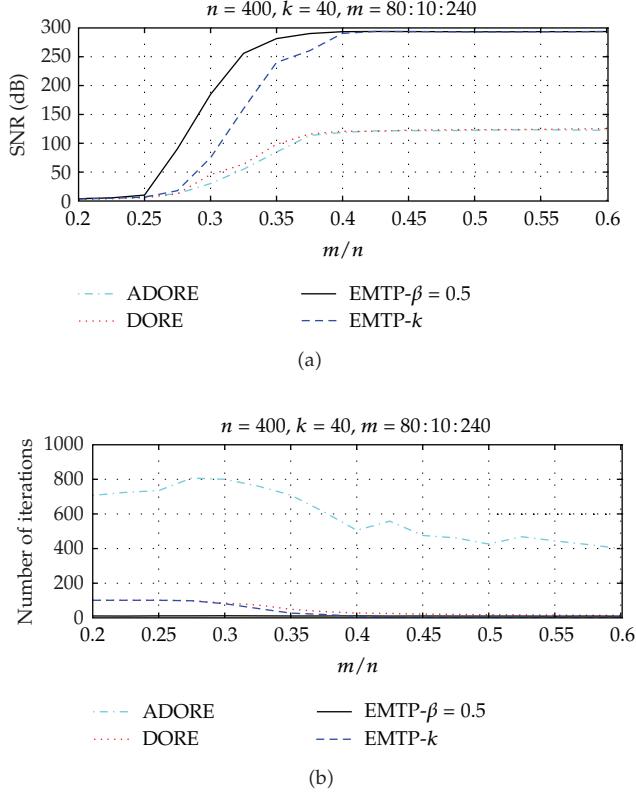


**Figure 11:** Sparse Gaussian signals with  $n = 2000, m = 400$ : comparisons in terms of SNR (dB) and number of iterations.

Figures 15 and 16. The latter tests come to similar conclusions as the first test. These figures do not fully capture the performance of EMTP- $\beta$  since we only set  $\beta = 0.5$ ; however EMTP- $\beta$  achieved almost superior performance than ADORE/DORE.

### 4.3. Performance for Other Fast-Decaying Signals

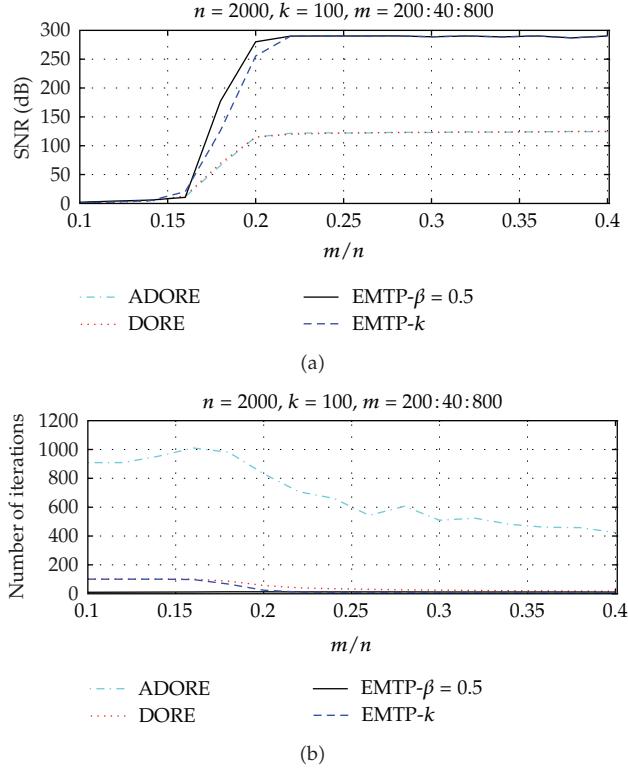
As discussed in [30, 33], thresholding-based sparse recovery methods achieve superior performance only if the nonzero elements of signals have a fast-decaying distribution. We present empirical studies on other fast-decaying signals. Sparse Laplacian signals containing  $k = 20$  nonzero elements for dimension  $n = 400$  were tested and the corresponding result was plotted in Figure 12. For larger dimension signals ( $n = 2000$ ), the results are shown in Figure 13. Power-law decaying signals containing  $k = 20$  nonzero elements for dimension  $n = 400$  were tested and the corresponding result was plotted in Figure 14. For larger-dimension signals ( $n = 2000$ ), the results are depicted in Figure 15. ADORE/DORE were derived from a probabilistic framework based on Gaussian distribution. Surprisingly, they can also work for fast-decaying signals, as depicted in Figures 12, 13, 14, and 15. The conclusions for sparse Gaussian signals also can be generalized for sparse Laplacian signals and power-law decaying signals. Power-law decaying signals for low indeterminacy level  $m/n$  make an exception that EMTP- $\beta$  achieves inferior performance.



**Figure 12:** Sparse Laplacian signals with  $n = 400$ ,  $k = 20$ : comparisons in terms of SNR (dB) and number of iterations.

#### 4.4. Phase Transitions

Following [31, 33], we present numerical comparisons in terms of phase transitions. For the sparse signal recovery from compressed measurements, the indeterminacy  $\delta = m/n$  defines the undersampling of a vector in compressed measurements. Let  $\rho = k/m$  be a normalized measure of the sparsity. We fix the problem size  $n = 400$  and test 100 Monte Carlo realizations of each sparse vector at each pair of  $(\rho, \delta)$ . We test a grid of  $16 \times 16$  linearly spaced  $(\rho, \delta)$  combinations with  $\rho, \delta$  varying from 0.05 to 0.5 in 16 steps. For each sparsity and indeterminacy pair, then, we find the probability of exact recovery with defining exact recovery when  $\|x_0 - x\|_2 / \|x\|_2 < 0.01$ . For each problem indeterminacy, we interpolate the results over all sparsities to find where successful recovery occurs with a probability of 0.5. As a result, we obtain a phase transition plot showing the boundary above which most recoveries fail and below which most recoveries succeed. Figure 16 presents the recovery rates of DORE, SP, and EMTP- $\beta$  for Gaussian-distributed sparse vectors as a function of problem sparsity and four problem indeterminacies from the thickest to thinnest lines. In Figure 16, we observe EMTP- $\beta$  outperforms SP, and DORE in recovery rates. Figure 17 shows the phase transitions of DORE, SP and EMTP- $\beta$  for compressively sensed sparse vectors sampled from Gaussian distribution. Figure 17 indicates that these transitions obey the following hierarchy in recovery performance: EMTP- $\beta$  > SP > DORE.



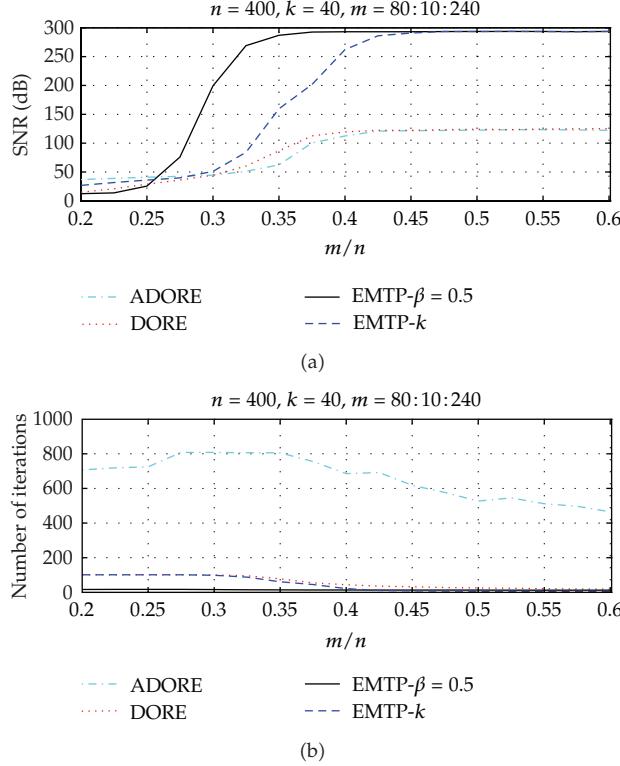
**Figure 13:** Sparse Laplacian signals with  $n = 2000, k = 100$ : comparisons in terms of SNR (dB) and number of iterations.

#### 4.5. Summary

To sum up, we have compared EMTP with ADORE/DORE. EMTP has significant advantages over the accelerated OST in terms of SNR and number of iterations. EMTP can significantly reduce the number of iterations required and achieves significantly higher SNR. For low indeterminacy level  $m/n$ , EMTP requires fewer measurements. EMTP- $\beta$  can work with no prior knowledge of the underlying sparsity level  $k$  yet achieves recovery performance better than ADORE. Furthermore, we compared EMTP with the state-of-the-art greedy algorithm SP in terms of phase transitions. Among all methods, EMTP- $\beta$  appeared to be the best.

### 5. Conclusions and Future Work

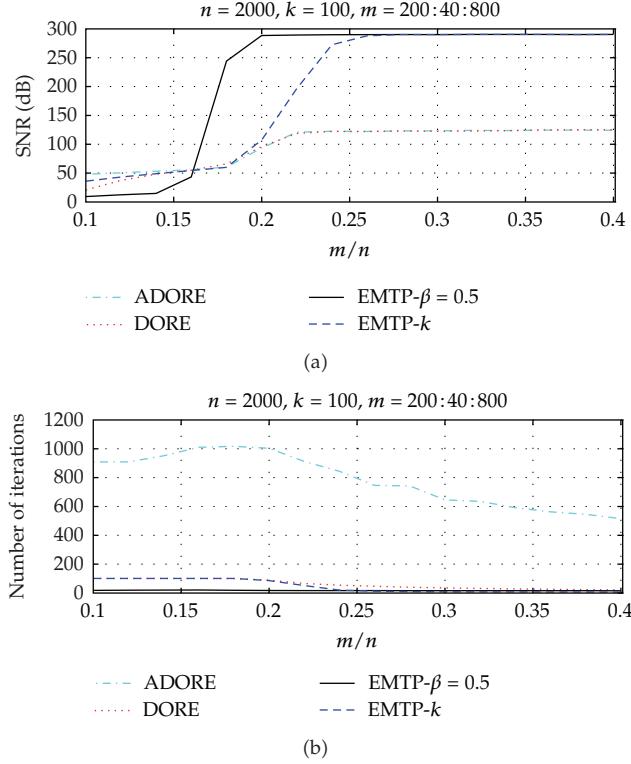
In this paper, we propose ECME thresholding pursuits (EMTP) for sparse signal recovery. EMTP detects a support set  $I$  using the ECME iteration and estimates the reconstructed signal by solving a truncated least-squares problem on the support set  $I$ , and it iterates these two steps for a small number of times. We present two effective support detection strategies (hard thresholding and dynamic thresholding) for the sparse signals with components having a fast-decaying distribution of nonzero components. The experimental studies are presented to demonstrate that EMTP offers an attractive alternative to state-of-the-art algorithms for sparse signal recovery. EMTP can significantly reduce the number of iterations required. We



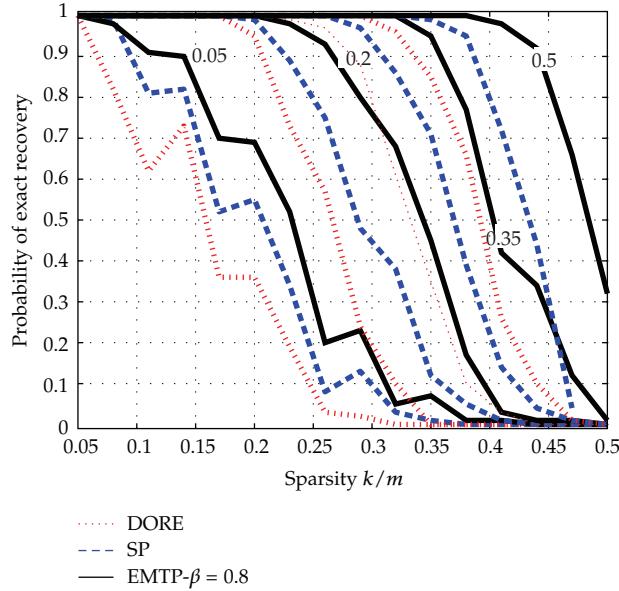
**Figure 14:** Power-law decaying signals with  $n = 400$ ,  $k = 20$ : comparisons in terms of SNR (dB) and number of iterations.

then turn to the problem of reducing the computational cost of each iteration, especially for large scale applications. Future research includes three directions.

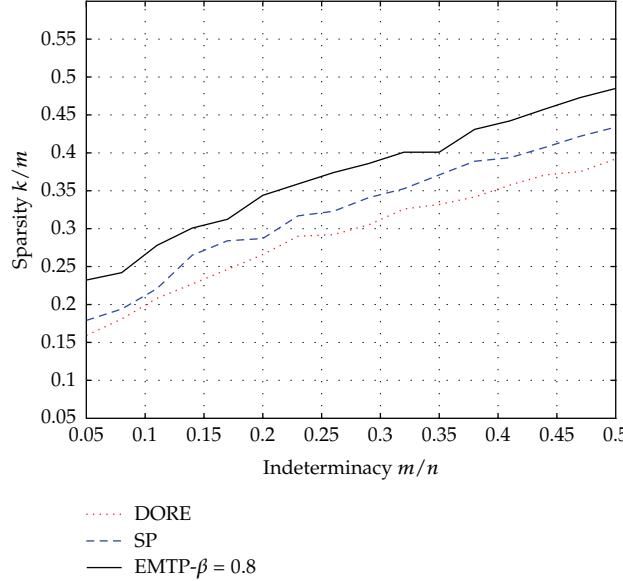
- (1) EMTP requires the precomputation and storage of the pseudoinverse matrix  $A^\dagger$ . So it is the advisable choice that replacing ECME iteration by IHT step in the support detection stage. IHT does not require the computation and storage of the pseudoinverse matrix  $A^\dagger$ , which leads to significant advantage for large scale applications.
- (2) In signal estimation stage, that is, update reconstructed signal by solving least-squares problem, EMTP uses the orthogonal projection, that is, by calculating  $x_I = A_I^\dagger y$  where  $A_I^\dagger$  is the pseudoinverse of  $A_I$ . We will approximate the orthogonal projection efficiently by gradient pursuits [32].
- (3) This paper devotes efforts to devise computational algorithms which are experimentally reproducible and provides empirical studies as useful guidelines for practical applications. Further, future investigations will test other OST methods as the reference and sophisticated thresholding methods for support detection.



**Figure 15:** Power-law decaying signals with  $n = 2000, k = 100$ : comparisons in terms of SNR (dB) and number of iterations.



**Figure 16:** Probability of exact recovery for DORE, SP, and EMTP as a function of problem sparsity and four problem indeterminacies from the thickest to thinnest lines:  $\delta = m/n = 0.05, 0.2, 0.35, 0.5$ .



**Figure 17:** Comparison of phase transitions of DORE, SP, and EMTP for Gaussian-distributed sparse vectors.

## Acknowledgment

This work was supported by the National Science Foundation of China under Grant no. 61074167, 61170126, and the Scientific Research Foundation for Advanced Talents by the Jiangsu University, no. 12JDG050.

## References

- [1] M. Elad, M. A. T. Figueiredo, and Y. Ma, "On the role of sparse and redundant representations in image processing," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 972–982, 2010.
- [2] M. J. Fadili, J. L. Starck, J. Bobin, and Y. Moudden, "Image decomposition and separation using sparse representations: an overview," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 983–994, 2010.
- [3] M. D. Plumley, T. Blumensath, L. Daudet, R. Gribonval, and M. E. Davies, "Sparse representations in audio and music: from coding to source separation," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 995–1005, 2010.
- [4] L. C. Potter, E. Ertin, J. T. Parker, and M. Çetin, "Sparsity and compressed sensing in radar imaging," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1006–1020, 2010.
- [5] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031–1044, 2010.
- [6] A. Y. Yang, M. Gastpar, R. Bajcsy, and S. S. Sastry, "Distributed sensor perception via sparse representation," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1077–1088, 2010.
- [7] R. Robucci, J. D. Gray, L. K. Chiu, J. Romberg, and P. Hasler, "Compressive sensing on a CMOS separable-transform image sensor," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1089–1101, 2010.
- [8] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.
- [9] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [10] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

- [11] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Review*, vol. 51, no. 1, pp. 34–81, 2009.
- [12] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 948–958, 2010.
- [13] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, vol. 24, no. 2, pp. 227–234, 1995.
- [14] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, no. 1, pp. 267–288, 1996.
- [15] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.
- [16] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, no. 3, pp. 211–244, 2001.
- [17] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153–2164, 2004.
- [18] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2346–2356, 2008.
- [19] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 53–63, 2010.
- [20] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.
- [21] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proceedings of the 27th Asilomar Conference on Signals, Systems & Computers*, vol. 1, pp. 40–44, Pacific Grove, Calif., USA, November 1993.
- [22] J. A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.
- [23] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.
- [24] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.
- [25] D. Needell and J. A. Tropp, "CoSaMP: iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.
- [26] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.
- [27] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.
- [28] K. Qiu and A. Dogandzic, "Double overrelaxation thresholding methods for sparse signal reconstruction," in *Proceedings of the 44th Annual Conference on Information Sciences and Systems (CISS '10)*, Princeton, NJ, USA, March 2010.
- [29] K. Qiu and A. Dogandzic, "Sparse signal reconstruction via ECME hard thresholding," *IEEE Transactions on Signal Processing*, vol. 60, no. 9, pp. 4551–4569, 2012.
- [30] Y. Wang and W. Yin, "Sparse signal reconstruction via iterative support detection," *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 462–491, 2010.
- [31] A. Maleki and D. L. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE Journal on Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 330–341, 2010.
- [32] T. Blumensath and M. E. Davies, "Gradient pursuits," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2370–2382, 2008.
- [33] B. Sturm, "A study on sparse vector distributions and recovery from compressed sensing," <http://arxiv.org/abs/1103.6246>.
- [34] X. Zhang, J. Wen, Y. Han, and J. Villasenor, "An improved compressive sensing reconstruction algorithm using linear/non-linear mapping," in *Proceedings of the Information Theory and Applications Workshop (ITA '11)*, pp. 146–152, San Diego, Calif., USA, February 2011.
- [35] X. Zhang, Z. Chen, J. Wen, J. Ma, Y. Han, and J. Villasenor, "A compressive sensing reconstruction algorithm for trinary and binary sparse signals using pre-mapping," in *Proceedings of the Data Compression Conference (DCC '11)*, pp. 203–212, Snowbird, Utah, USA, March 2011.
- [36] V. Stodden, L. Carlin, D. Donoho et al., "SparseLab: seeking sparse solutions to linear systems of equations," Tech. Rep., Stanford University, 2011.

*Research Article*

## **Variable Neighborhood Search for Parallel Machines Scheduling Problem with Step Deteriorating Jobs**

**Wenming Cheng,<sup>1</sup> Peng Guo,<sup>1</sup> Zeqiang Zhang,<sup>1</sup>  
Ming Zeng,<sup>1</sup> and Jian Liang<sup>2</sup>**

<sup>1</sup> School of Mechanical Engineering, Southwest Jiaotong University, Chengdu 610031, China

<sup>2</sup> School of Mechanical Engineering and Automation, Xihua University, Chengdu 610039, China

Correspondence should be addressed to Peng Guo, pengguo318@gmail.com

Received 20 February 2012; Revised 27 April 2012; Accepted 8 May 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Wenming Cheng 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 many real scheduling environments, a job processed later needs longer time than the same job when it starts earlier. This phenomenon is known as scheduling with deteriorating jobs to many industrial applications. In this paper, we study a scheduling problem of minimizing the total completion time on identical parallel machines where the processing time of a job is a step function of its starting time and a deteriorating date that is individual to all jobs. Firstly, a mixed integer programming model is presented for the problem. And then, a modified weight-combination search algorithm and a variable neighborhood search are employed to yield optimal or near-optimal schedule. To evaluate the performance of the proposed algorithms, computational experiments are performed on randomly generated test instances. Finally, computational results show that the proposed approaches obtain near-optimal solutions in a reasonable computational time even for large-sized problems.

### **1. Introduction**

Scheduling is a form of decision making that plays a crucial role in manufacturing and service systems. It began to be taken seriously in manufacturing at the beginning of 20th century, and since then has been received the attention of many researchers for years. In the traditional scheduling problems, most research in the literature was usually conducted under the assumption that the processing time of a job is known in advance and remains constant throughout the whole decision process. However, there are many practical situations where the processing times of jobs are not constant but increasing overtime such that the later a job starts, the longer it takes to process. This phenomenon is known as scheduling

with deteriorating jobs to many industrial applications, such as equipment maintenance, steel production, medical emergency, firefighting, and other problems. In these cases, the corresponding scheduling problem was first studied by Browne and Yechiali [1] and J. N. D. Gupta and S. K. Gupta [2]. They assumed that the processing time of all jobs is a function of their starting point ( $p_i = a_i + \alpha_i \times s_i$ ), where  $a_i$  is the normal processing time,  $\alpha_i$  is a deterioration rate, and  $s_i$  is the starting time of job  $i$ . From then onwards, more papers considering scheduling jobs with deterioration effects have been published recently [3–19]. Comprehensive reviews and discussions of different models and problems with time-dependent processing times were given by [20, 21].

Most of the current researches in deteriorating jobs scheduling problems assume that the job processing times are linear functions of their starting times. However, in many practical situations, if some jobs fail to be processed prior to a prespecified deteriorating date, then the jobs will require extra time for successful completion. As a result, the original schedule may become inapplicable under the new environment. This motivates the research of the scheduling problem with piecewise-deteriorating jobs. There are often two types of scheduling problems relevant to the piecewise-deteriorating model in the literature. The first one is step-deterioration scheduling problem [22]. Another one is piecewise-linear-deteriorating scheduling problem [23]. In this paper, we study the first kind of scheduling problem. Considering a set of  $n$  jobs with deteriorating date  $d_i$  for job  $J_i$ , the processing time  $p_i$  for job  $J_i$  is  $a_i$  if it is started before the deteriorating date and  $a_i + b_i$  if the starting time  $s_i > d_i$ , where  $a_i$  is the normal processing time,  $b_i$  is a deteriorating penalty. The objective is to schedule the  $n$  jobs on parallel machines to minimize the sum of job completion times. The scheduling problem with step-deterioration jobs is NP-complete even if in single machine environment [24]. As for such a problem over a certain size, it is handicapped to render an optimal schedule within a reasonable run time, depending on exact methods. Hence, some approximate solution techniques are employed to yield an optimal or near-optimal schedule for such problems.

In this paper, a modified weighted combination search algorithm (MWCSA) is firstly proposed to obtain near optimal solution. Subsequently, variable neighborhood search (VNS) algorithm is employed to yield the better schedule for the problem under consideration. VNS is one of the modern metaheuristic techniques that use systematic changes of the neighborhood structure within a local search to solve optimization problem [25, 26]. Owing to incorporating a lot desirable properties for a metaheuristics such as simplicity, efficiency, effectiveness, generality, and so forth, VNS has been widely used to combinatorial optimization problems in recent years. In addition, to improve the performance of VNS, the proposed heuristic MWCSA is utilized to produce initial solution for the VNS. The effectiveness of the proposed approaches is demonstrated by computational results based on a large set of randomly generated test instances.

This paper is organized as follows: Section 2 reviews the relevant literature in this area of scheduling. Section 3 gives a brief description of the problem under consideration and formulates the corresponding mixed integer programming model. A detailed description of the heuristic MWCSA and VNS algorithm are elaborated in Section 4. The performances of the VNS and the heuristic algorithms are shown in Section 5. Conclusions are given in Section 6.

## 2. Literature Review

We discuss related work with respect to the scheduling problem with piecewise-deteriorating jobs and with regard to VNS methods in scheduling, especially for parallel machines.

Since many different problems are encountered in production and service environments where the processing time of jobs or tasks can be modeled as the piecewise-deteriorating function, many active researchers in the field of scheduling have perceived the promising advantages of considering the piecewise-deteriorating jobs in scheduling problems. Kunnathur and Gupta [27] firstly assumed that job processing times are piecewise-linear function of their starting times with two pieces and addressed five heuristics and two optimizing algorithms based on dynamic programming and branch and bound techniques. However, Sundararaghavan and Kunnathur [22] considered that the processing time can be modeled by step function and gave some solvable cases for minimizing the sum of the weighted completion times in single machine scheduling. In addition, Mosheiov [28] studied the problem of stepwise deteriorating jobs for minimizing makespan, proved that the case of single machine step-deterioration is NP-complete, and suggested heuristic methods in single and multimachine scheduling environments.

Cheng and Ding [24] studied a piecewise model where the job processing time deteriorates as a step function if its starting time exceeds a given threshold, and presented NP-complete proofs for minimizing makespan, flow time, and weighted flow time in single machine scheduling. Jeng and Lin [29] presented a pseudopolynomial time dynamic programming algorithm for a single-machine scheduling problem. Then, they described an efficient branch and bound algorithm based on two dominance rules and a lower bound for deriving optimal solution. Subsequently, a single machine scheduling problem of minimizing the total completion time was solved by branch and bound algorithms [30]. Owing to the complexity of the problem, He et al. [31] proposed a weight combination search algorithm to yield a near-optimal solution. They also developed a branch and bound algorithm incorporated with several properties and a lower bound to obtain the optimal solution. Afterwards, Layegh et al. [32] applied a memetic algorithm based on three dominance properties to minimize the total weighted completion time on a single machine under step-deterioration.

With regards to the piecewise-linear-deterioration model, Kovalyov and Kubiak [33] investigated that the job processing time can be expressed by a piecewise-linear-increasing function with three pieces and presented a fully polynomial approximation scheme for minimizing makespan of single machine scheduling problem. Then Kubiak and van de Velde [23] proved that the problem is NP-hard, and proposed a branch and bound algorithm that solves instances with up to 100 jobs in a reasonable amount of time. Alternatively, the processing job times was expressed in a piecewise-linear-decreasing function of their start times. The single machine scheduling problem was researched by Cheng et al. [34]. Moslehi and Jafari [35] dealt with the same problem proposed by [33] with the minimization of the number of tardy jobs. Owing to the complexity of the studied problem, they proposed a heuristic algorithm with  $O(n^2)$  and a branch and bound algorithm.

As far as reviewed, there are some literature focused on single machine scheduling problem with piecewise-deteriorating jobs. However, to the best of our knowledge, parallel machines scheduling problem with piecewise deteriorating jobs has not been considered in the existing literature. And it still lacks some effective approaches to solve the intractable problem, especially metaheuristics. Therefore, we intend to propose a VNS approach for parallel machines scheduling problem with step-deteriorating jobs.

VNS approaches have successfully applied to solve several scheduling problems. Gupta and Smith [36] described a new hybrid of VNS for single machine total tardiness scheduling with sequence-dependence setups. Paula et al. [37] applied a VNS algorithm to solve large instances of parallel machines scheduling problems with sequence-dependent

times. Anghinolfi and Paolucci [38] contributed to the design of a hybrid meta-heuristic approach which integrates several features from tabu search (TS), simulated annealing (SA), and VNS for a generalized parallel machine total tardiness scheduling problem. Moreover, Driessel and Mönch [39] proposed several variants of VNS schemes for the same problem considered precedence constraints to minimize the total weighted tardiness of all jobs. Behnamian et al. [40] proposed a hybrid meta-heuristic method which combines some advantages of ant colony optimization, SA and VNS for the minimization of makespan in parallel machine scheduling problem with sequence-dependent times. C.-L. Chen and C.-L. Chen [41] proposed several hybrid metaheuristics for unrelated parallel machine scheduling with sequence dependent setup times by integrating VNS and TS principles. Furthermore, some job shop scheduling problems were solved by VNS [42–45]. A latest systemic survey of state-of-the-art development of VNS can be found in [46].

### 3. Problem Formulation

The problem under consideration is to schedule  $n$  independent jobs with step-deterioration effects, noncommon deadlines and varying processing times on  $m$  identical parallel machines. The jobs are available for processing at time zero and the machines are available in the whole process. No job preemption is permitted. For each job  $J_i$ , there is a normal processing time  $a_i$  and associated with a deteriorating date  $d_i$ . If the starting point of its process is less than or equal to its deteriorating date, then it only requires a normal processing time  $a_i$ . Otherwise, it requires an extra processing time  $b_i$ , which is called the deteriorating penalty. Thus, the actual processing time  $p_i$  of job  $J_i$  depends on its starting time  $s_i$  and deteriorating date  $d_i$ , and can be defined as a step-function:  $p_i = a_i$  if  $s_i \leq d_i$ ;  $p_i = a_i + b_i$ , otherwise. Without loss of generality, it is assumed that parameters  $a_i$ ,  $b_i$ , and  $d_i$  are all integers. Let  $C_i$  denote the completion time of job  $J_i$ . The goal of the problem is to find a schedule such that the total completion time, or the sum of completion times, of all jobs is minimized. This problem is denoted as  $Pm/p_i = a_i$  or  $a_i + b_i, d_i/\sum C_i$  by adopting the standard three-field notation.

For convenience, a job is called early if its starting time is before or at its deteriorating date; tardy, otherwise. For schedule  $S$ , the objective value is denoted by  $Z(S)$ . In addition, let  $M$  be a sufficiently large positive number. Before formulating the proposed mixed integer programming model, the following variables have to be defined.

$x_{ijk}$ : Binary, set to 1 if job  $i$  is immediately followed by job  $j$  in sequence on machine  $k$ ; 0, otherwise ( $1 \leq i, j \leq n, 1 \leq k \leq m$ )

$y_{ik}$ : Binary, set to 1 if job  $i$  is assigned to machine  $k$ ; 0, otherwise ( $1 \leq i \leq n, 1 \leq k \leq m$ ).

Based on the definitions and notation described above, the considered problem can now be formulated as 0-1 mixed integer programming model, as shown below.

$$\text{Minimize } Z(S) = \sum_{i=1}^n C_i. \quad (3.1)$$

Subject to:

$$p_i = \begin{cases} a_i, & s_i \leq d_i \\ a_i + b_i, & \text{otherwise} \end{cases} \quad \forall i = 1, \dots, n, \quad (3.2)$$

$$\sum_{i=1}^n x_{0ik} = 1 \quad \forall k = 1, \dots, m, \quad (3.3)$$

$$\sum_{i=1}^n x_{i(n+1)k} = 1 \quad \forall k = 1, \dots, m, \quad (3.4)$$

$$\sum_{i=0, i \neq j}^n x_{ijk} = y_{jk} \quad \forall j = 1, \dots, n, \quad \forall k = 1, \dots, m, \quad (3.5)$$

$$\sum_{j=1, j \neq j}^{n+1} x_{ijk} = y_{ik} \quad \forall i = 1, \dots, n, \quad \forall k = 1, \dots, m, \quad (3.6)$$

$$C_i \geq p_i + M(x_{0ik} - 1) \quad \forall i = 1, \dots, n, \quad \forall k = 1, \dots, m, \quad (3.7)$$

$$C_j \geq C_i + p_j + M(x_{ijk} - 1) \quad \forall i = 1, \dots, n, \quad \forall j = 1, \dots, n, \quad \forall k = 1, \dots, m, \quad (3.8)$$

$$s_i \geq M(x_{0ik} - 1) \quad \forall i = 1, \dots, n, \quad \forall k = 1, \dots, m, \quad (3.9)$$

$$s_j \geq C_i + M(x_{ijk} - 1) \quad \forall i = 1, \dots, n, \quad \forall j = 1, \dots, n, \quad \forall k = 1, \dots, m, \quad (3.10)$$

$$\sum_{k=1}^m y_{ik} = 1, \quad \forall i = 1, \dots, n, \quad (3.11)$$

$$x_{ijk}, y_{ik} \in \{0, 1\}, \quad \forall i = 1, \dots, n, \quad \forall j = 1, \dots, n, \quad \forall k = 1, \dots, m. \quad (3.12)$$

In the above mathematical model, the objective (3.1) minimizes the total completion time. Constraints (3.2) depict the processing time of the step-deteriorating jobs. Constraints (3.3) and (3.4) ensure that only one job can be processed at the first and the last position on each machine. Constraints (3.5) and (3.6) guarantee that each job is scheduled only once and processed by at most one machine. Constraints (3.7) define the completion time of the first job assigned to a machine. Constraints (3.8) represent that the completion time of a job in sequence on each machine will be at least equal to the sum of the completion time of the preceding job and the processing time of the present job, if the job is immediately scheduled after the previous job. Constraints (3.9) define the starting time of the first job assigned to a machine. Constraints (3.10) state that the starting time of a job in sequence on each machine is greater than or equal to the completion time of the preceding job. Constraints (3.11) confirm that each job is only processed by exactly one machine. Constraints (3.12) specify that the decision variable  $x$  and  $y$  is binary over all domains.

This described problem is NP-complete because the single machine problem  $1/p_i = a_i$  or  $a_i + b_i, d_i / \sum C_i$  has proven to be NP-complete by [24]. There exists no polynomial time algorithm for the exact solution of the considered problem. Therefore, we have to look for some efficient heuristic or meta-heuristic algorithms to yield near optimal solutions of large-sized problems in a reasonable computational time.

## 4. Heuristic Solution Approaches

In this section, we start with presenting one property. Then we modify a heuristic proposed in a literature. Subsequently, some detailed descriptions of VNS meta-heuristic are given.

Since the identical parallel machines scheduling problem  $Pm//\sum C_i$  is solvable in  $O(N \log N)$  time using the generalized shortest processing time (GSPT) rule [47]. Therefore, we can also use the same method to determine the sequence of early jobs and tardy jobs assigned to a given machine for minimizing the total completion time. Based on the GSPT, we obtain the following property directly.

*Property 1.* On each machine of an optimal solution for the considered problem, early jobs and tardy jobs are sequenced in the nondecreasing order of  $a_i$  and  $a_i + b_i$ , respectively.

*Proof.* The proof is straightforward from the SPT rule on  $a_i$  and  $a_i + b_i$ , respectively.  $\square$

### 4.1. Modified Weight Combination Search Approach

He et al. [31] proposed weight combination search approach (WCSA) to solve single machine total completion time scheduling problem with step-deteriorating jobs. The performance of the WCSA is relatively excellent compared to the proposed branch and bound algorithm within 24 jobs. Based on the good performance of the algorithm, it is modified to adapt the parallel machine model by incorporating Property 1. But the ranges of the weights in the literature are not suitable for parallel machine scheduling. For specifying the suitable ranges of the weights we carried out the preliminary tests of 30 10-job instances and found that the linear combination of  $a_i$ ,  $d_i$ , and  $b_i$  will acquire better results if  $\omega_1 \in [0.4, 0.75]$ ,  $\omega_2 \in [0.2, 0.5]$  and  $\omega_3 = 1 - \omega_1 - \omega_2$ . The procedure of the modified WCSA (MWCSA) is designed as show in Algorithm 1.

From Algorithm 1, all jobs are sorted in non-decreasing order of their normal processing times  $a_i$ . The first  $m$  jobs are assigned to  $m$  machines, respectively. Then the last  $n-m$  jobs are arranged in a non-decreasing order of the weight combination of processing time, deteriorating date and deterioration penalty. Of course, if the completion time of the last scheduled job is greater than the maximal deteriorating date of the unscheduled jobs, then the remaining jobs are indexed in the order of sum of their normal processing times and deterioration penalties.

### 4.2. Variable Neighborhood Search Metaheuristic

VNS, a new local search technique, attempts to escape from local optimum by exploring more than one type of neighborhood search structure (NSS) during the course of the algorithm. In practice, VNS is very similar to iterated local search (ILS). Owing to iterating over one constant type of neighborhood structure, ILS is easy to stick in local optima: the single move required to improve the solution cannot escape from the found local optima, and even lead to a deterioration of the solution quality. The disadvantage is also existed in some other metaheuristics, such as simulated annealing, and tabu search. However, VNS makes full use of some NSSs until some stopping criterion is met. By exploring various NSSs, we expect VNS to enjoy a systematic diversification mechanism. Besides this diversification mechanism, the other reasons to high acceptability and popularity of VNS among researchers are due

**Procedure:** the modified WCSA for the parallel machine scheduling problem

**Inputs:**  $n, m, a_i, d_i, b_i$  for  $i = 1, \dots, n$

**Output:** the near optimal scheme  $S_{opt}$  and the associated total completion time  $TC$

**Begin**

- let  $N_0 = \{J_1, J_2, \dots, J_n\}$  be the sequence that sorts all the jobs by ascending order of their normal time  $a_i$ ,
- set  $TC = \text{infinity}$ , and  $S_{opt_k} = \Phi$  for  $k = 1, \dots, m$  % initialize the  $TC$  and  $S_{opt}$
- set var =  $\max\{2, [n/m]\}$  % initialize the ranges of  $l_1$  and  $l_2$
- for** ( $l_1 = 1; l_1 \leq \text{var}; l_1++$ )

  - $\omega_1 = 0.4 + (0.75 - 0.4) \times ((l_1 - 1) / (\text{var} - 1))$
  - for** ( $l_2 = 1; l_2 \leq \text{var}; l_2++$ )

    - $\omega_2 = 0.2 + (0.5 - 0.2) \times ((l_2 - 1) / (\text{var} - 1))$
    - $\omega_3 = 1 - \omega_1 - \omega_2$
    - set  $N = N_0$
    - set  $S = \Phi$  for  $k = 1, \dots, m$  % initialize the scheme  $S$
    - set  $C^{\text{mac}}(k) = 0$ , for  $k = 1, \dots, m$  % initialize the completion time of all machines
    - set  $C^{\text{job}}(i) = 0$ , for  $i = 1, \dots, n$  % initialize the completion time of all jobs
    - for** ( $i = 1; i \leq m; i++$ )

      - select the machine  $h$  that has the least completion time in all machines
      - select the job  $J_h$  that has the least normal time from  $N$
      - $S_h = S_h \cup \{J_h\}$
      - $C^{\text{mac}}(h) = C^{\text{mac}}(h) + a(J_h)$
      - $C^{\text{job}}(J_h) = C^{\text{mac}}(h)$
      - delete job  $J_h$  from  $N$

    - end for**
    - for** ( $j = m + 1; j \leq n; j++$ )

      - select the machine  $f$  that has the least completion time in all machines
      - if** ( $C^{\text{mac}}(f) > \max\{d(J_r), (J_r \in N)\}$ ) % tardy jobs are sequenced in the nondecreasing order of  $a_i + b_i$

        - select the job  $J_f$  with the smallest  $a(J_f) + b(J_f)$  from  $N$
        - $S_f = S_f \cup \{J_f\}$
        - $C^{\text{mac}}(f) = C^{\text{mac}}(f) + a(J_f) + b(J_f)$
        - $C^{\text{job}}(J_f) = C^{\text{mac}}(f)$
        - delete job  $J_f$  from  $N$

      - else** % arrange the jobs in a ascending order of the weight of combination

        - set  $N' = \{d(J_r) \geq C^{\text{mac}}(f), (J_r \in N)\}$
        - select job  $J_f$  with the smallest  $\omega_1 \times a(J_f) + \omega_2 \times d(J_f) - \omega_3 \times b(J_f)$  from  $N'$
        - $S_f = S_f \cup \{J_f\}$
        - $C^{\text{mac}}(f) = C^{\text{mac}}(f) + a(J_f)$
        - $C^{\text{job}}(J_f) = C^{\text{mac}}(f)$
        - delete job  $J_f$  from  $N$

      - end if**

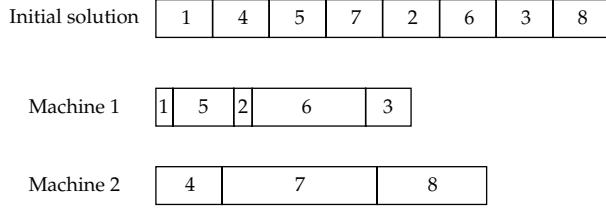
    - end for**
    - $TC_{\text{temp}} = \sum\{C^{\text{job}}(i) = 0, \text{ for } i = 1, \dots, n\}$
    - if**  $TC_{\text{temp}} < TC$  % update the scheme  $S_{opt}$  and the assisted completion time  $TC$

      - $TC = TC_{\text{temp}}$
      - $S_{opt} = S$
      - end if**

    - end for**
    - end for**

End


**Algorithm 1:** Procedure of MWCSA.



**Figure 1:** Sequence in one solution and its schedule.

to conceptual simplicity to understand and implement and the high flexibility and brilliant adaptability to different problems. Consequently, we intend to use it to solve our problem.

Since the VNS was proposed in 1997, some variants of VNS are developed by Hansen and Mladenović [25]. Variable neighborhood descent (VND), as one of these variants of VNS, performs change of neighborhoods in a deterministic way and adopts the first improvement rather than the best improvement. In the subsequent subsection, we are going to employ a VNS with five powerful NSSs based on insertion, swap, and inversion under the framework of VND.

#### 4.2.1. Encoding, Decoding, and Initialization Schemes

Encoding scheme is a procedure to make a solution recognizable for an algorithm. Permutation list is a common used encoding scheme for combinatorial optimization problems. Since the parallel machines scheduling problem is a combination of machine assignment and operation sequencing decisions, we construct five neighborhood structures that expediently use insertion, swap, and inversion operations based on permutation list. A good initial solution can reduce considerably the computational time. According to observing the processing sequence of jobs, we find that a job with the smallest normal processing time should be processed at first, and a job with the largest penalty should be processed at last. Therefore, we can obtain the initial solution by arranging jobs in the non-decreasing order of the ratio of  $a_i/b_i$ . The sample sequencing method is called smallest rate first (SRF). A permutation with all jobs sequence can express the order in which the jobs are processed. To decode a permutation list into a schedule, we use LIST method. For one solution sequence, whenever a machine is freed the job that is chosen from the sequence in succession is put on the machine.

The procedures of encoding and decoding a candidate solution are illustrated according to an example. Consider a problem with 8 jobs and 2 machines. The parameters of all jobs are given in Table 1. The initial solution is obtained by using SRF, as shown in Figure 1. The job that is selected from the solution sequence one by one is processed by the machine with the smallest completion time. The schedule is also depicted in Figure 1. We calculated that the total completion processing time is 1113.

#### 4.2.2. Neighborhood Search Structures

The main purpose of applying a neighborhood structure is to produce a neighboring solution from the current solution via making some changes in it. A variety of neighborhood structures have been applied to scheduling problems. Most of these neighborhood structures are based

**Table 1:** The parameters of all jobs in the example under consideration.

Parameter	Job number							
	$J_1$	$J_2$	$J_3$	$J_4$	$J_5$	$J_6$	$J_7$	$J_8$
$a_i$	10	13	28	55	63	81	90	95
$d_i$	14	48	21	52	55	60	36	7
$b_i$	14	3	5	41	35	16	47	2

```

Procedure: NSS1
improvement = yes
while (improvement = yes) do
    improvement = no
    i = 1
    while i ≤ n do      (n is the number of jobs)
        select randomly one job p without repletion from the current solution x
        x' = swap job i and job p on their positions.
        if Z(x') < Z(x) do
            x = x'
            i = n
            improvement = yes
        end if
        i = i + 1
    end while
end while

```

**Algorithm 2:** Procedure of neighborhood structure NSS<sub>1</sub>.

on insertion or swap operations. In addition, the inversion operation has been reported to be significantly surpassing all the modified forms of crossover of genetic algorithm especially tailored to deal with combinational problems [48]. Therefore, we define five types of neighborhood search structures based on insertion, swap, and inversion.

To generate a neighboring solution, NSS<sub>1</sub> makes some slight changes in the candidate solution by swapping the positions of all jobs in the sequence with one selected job at random and without repetition. All jobs are selected one after another without repetition in a random order. If we observe the first improvement, the associated sequence is accepted and the procedure restarts. According to the NSS<sub>1</sub>, some available neighboring searches are found and used to improve the candidate solution. The whole procedure repeats so long as no improvement is obtained through swap all jobs with another randomly selected job. The procedure of NSS<sub>1</sub> is described in Algorithm 2 in detail. After implementing all swap operations, we believe that there is little hope for further improvement just by swapping two jobs on their positions. Hence, it necessitates consider another neighborhood structure to escape from this local optimum of the NSS<sub>1</sub>. Subsequently, we need to introduce another NSS to search potential improvement based on insertion moves.

In our NSS<sub>2</sub>, the procedure is similar to the NSS<sub>1</sub>. The only difference between them is their move pattern that NSS<sub>2</sub> changes the position of one job based on insertion neighborhood. A job is removed from the sequence at random and without repetition, and then relocated another random selected position. The other procedure is the same as the

```

Procedure: NSS3
i = 1
while i ≤ n do
    j = j + 1
    while j ≤ n do
        select randomly two jobs p and q from the current solution x
        x' = swap the two selected jobs on their positions
        if Z(x') < Z(x) do
            x = x'
            j = n
            i = n
        end if
        j = j + 1
    end while
    i = i + 1
end while

```

**Algorithm 3:** Procedure of neighborhood structure NSS<sub>3</sub>.

NSS<sub>1</sub>. To avoid duplicated description, it is unnecessary to go into these details here. After relocating all jobs, we feel that the algorithm cannot rely solely on single move to improve the quality of the current solution. Hence, we need to introduce other neighborhood search structure to generate more complex neighbors than the above two structures.

In the NSS<sub>3</sub>, the number of randomly selected jobs is 2. The manner of choosing these 2 jobs is all the combinations of two-out-of-*n* jobs. Swap the selected jobs on their positions and observe if the solution is improved. Once observing the first improvement, the associated solution is accepted and the procedure restarts. If not, this search strategy repeats for the subsequent combinations. Algorithm 3 illustrated the whole procedure of the NSS<sub>3</sub>. By the same way of switching from NSS<sub>1</sub> to NSS<sub>2</sub>, the double insertions are employed in the fourth neighborhood structure NSS<sub>4</sub>. In the NSS<sub>4</sub>, the number of removed jobs is set equal to 2. To relocate them, the two jobs are reinserted into two new randomly selected positions. The other procedure is the same to the NSS<sub>3</sub>.

To improve further the search performance of the proposed VNS, an inversion operator is embedded into the NSS<sub>5</sub> when the foregoing neighborhood search methods terminate. In the NSS<sub>5</sub>, it randomly selects two positions, known as the points of inversion, and inverts the sequence between these positions. The inversion procedure is repeated  $\varphi$  times for producing a new job sequence. Owing to the impact of the parameter  $\varphi$  on neighborhood search, it must be tuned. The inversion operation is unidirectional, that is, only the inverted sequence is improved, the corresponding solution is accepted to replace the incumbent one. The procedure of is described in Algorithm 4.

The general outline of the proposed VNS is shown in Algorithm 5. According to using the above neighborhood search structures, the proposed VNS can be quick to yield the near optimal solution. If the neighborhood search structure continues to increase the number of selected jobs, the quality of the algorithm will be not improved according to our primary experiments.

```

Procedure:  $NSS_5$ 
for  $i = 1$  to  $\varphi$  do
    select two jobs  $p$  and  $q$  at random from the current solution  $x$ 
     $x' =$  invert partial jobs between job  $p$  and job  $q$ 
    if  $Z(x') < Z(x)$  do
         $x = x'$ 
    end if
end for

```

**Algorithm 4:** Procedure of neighborhood structure  $NSS_5$ .

```

Procedure: the proposed VNS algorithm
Initialization: Define the set of neighborhood structures  $NSS_k$ , for  $k = 1, \dots, k_{\max}$ ;
    find a initial solution  $x$  by SRF heuristic;
    choose a stopping criterion.
 $k = 1$ 
while the stopping criterion is not met do
    perform the neighborhood search structure  $NSS_k$ 
    if  $x$  is improved do
        continue the search with  $NSS_k$ 
    else
         $k = k \bmod k_{\max} + 1$ 
    end if
end while

```

**Algorithm 5:** Procedure of the proposed VNS algorithm.

## 5. Computational Analysis

To test the performance of our approaches, computational experiments were carried out for the  $Pm/p_i = a_i$  or  $a_i + b_i$ ,  $d_i/\sum C_i$  problem. More specifically, we also carried out comprehensive computational and statistical tests to assess the performance of MWCSA, and VNS. Firstly, we gave a detailed description of the different instance sets that we have employed. After reporting the results, we carried out comprehensive statistical analyses in order to soundly test the significance of the reported results.

Because there is not a widely available set of benchmark instances for the problem under study, data for the test instances are generated randomly. The normal processing times ( $a_i$ ) are randomly generated from an integer uniform distribution on  $U[1, 100]$ . The deterioration penalties ( $b_i$ ) are randomly picked from an integer uniform distribution on  $U[1, 100 \times \beta]$ , where  $\beta = 0.5$ . Let  $D_f = \sum_{i=1}^{f \times n} a_i/m$ ,  $0 \leq f \leq 1$ . The deteriorating dates  $d_i$  are randomly selected from three intervals  $U(0, D_{0.5}]$ ,  $U[D_{0.5}, D]$ , and  $U(0, D]$ . In this paper, we have divided the benchmark instances into 2 different types according to the number of jobs. For small-sized instances, we test all combinations of  $n = \{6, 8, 10\}$  and  $m = \{2, 3\}$ . For large-sized instances, we use the combinations of  $n = \{20, 40, 60, 80, 100\}$  and  $m = \{2, 4, 6, 8, 10\}$  to evaluate the proposed algorithms. Since the deteriorating date may be generated from three different intervals, we totally have  $6 \times 3 + 25 \times 3 = 93$  instances.

In order to validate the performance of the proposed approaches, attempts are made to solve the MIP model presented in Section 3 using the software IBM-ILOG CPLEX 12.3 solvers. The software is run on a PC with Intel Pentium Dual-Core 2.60 GHz CPU and 2 GB RAM. Because the problem under consideration is NP-complete, it is impossible to obtain optimal solutions by using some polynomial time algorithms. Therefore, only small-sized instances can be solved to optimality using CPLEX. For the small-sized instances, we use relative percentage deviation (RPD) as a performance measure to compare those results of the CPLEX software, the MWCSA and the VNS. RPD is obtained by given formula below:

$$\text{RPD} = \frac{Z(\text{Alg}) - Z(\text{OPL})}{Z(\text{OPL})} \times 100, \quad (5.1)$$

where  $Z(\text{Alg})$  is the total completion time of the solution obtained by a given algorithm and instance and  $Z(\text{OPL})$  is the total completion time of the optimal schedule given by CPLEX. The SRF (small rate first) of Section 4 is also brought into the large-sized instances for comparison. Our purpose to utilize SRF is to use its result as upper bound for a given instance to evaluate the VNS. In addition, the MWCSA is used to obtain the initial solution of the VNS for improving the quality of solutions. The hybrid algorithm, denoted by VNS + MWCSA, is employed for the large-sized instances.

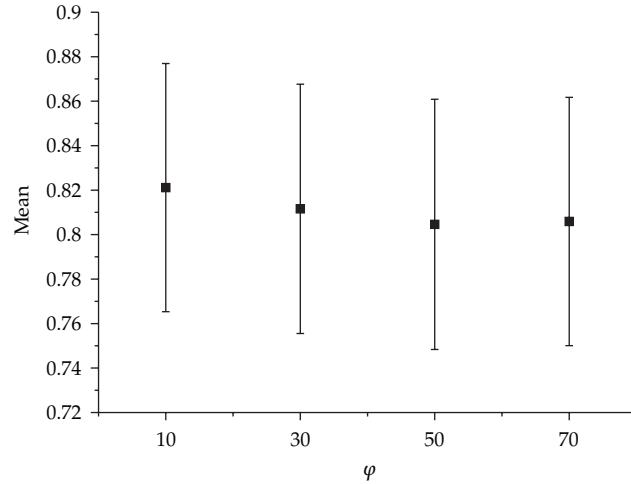
All proposed algorithms in the previous sections were coded in MATLAB 7.11 and implement on the same PC. According to the preliminary tests, the stopping criterion used when testing all instances with the proposed algorithms is set to the maximum iterations time fixed to 200. Thus, there is only one parameter ( $\varphi$ ) is tuned in our proposed VNS. The considered candidate levels are 10, 30, 50, and 70. A set of 15 large-sized instances are randomly generated when the number of jobs is fixed to 100. All the 15 instances are solved by these VNSs with different parameter candidate levels. The results were analyzed by the one-way ANOVA test. The means plot and Fisher LSD intervals for the 4 levels of parameter  $\varphi$  are shown in Figure 2. As we can see,  $\varphi$  of 50 and 70 provide the better results among all levels. However, it needs more computational time when  $\varphi$  is 70. Therefore, the most reasonable value of parameter  $\varphi$  is 50.

CPLEX, SRF, and MWCSA are deterministic and only one run is necessary. The VNS and VNS + MWCSA are stochastic and we need to run some replicates in order to better assess the results. We run each algorithm five times. The listed objective value and computational time are the means of five reported results.

Table 2 provides a comparison of the solutions to the mixed integer programming model generated by the CPLEX and those solutions provided by execution of the proposed MWCSA and VNS algorithms. All small-sized instances were solved to optimality by CPLEX as shown in Table 1. The computational time of the CPLEX increases rapidly as the instance become larger. When the number of jobs is equal to 10, solutions to MIP models require far longer than 1 hour. From Table 1, the longest computational time of CPLEX is 144761.59 seconds (40.21 hours). It is impractical in scheduling for such a long time. In addition, we found that CPLEX is not greatly affected by the distribution of the deteriorating dates almost. While the deteriorating dates are randomly picked from the interval  $U[D_{0.5}, D]$ , all corresponding instances are relatively easy. This is because the number of deteriorating jobs is less. It is worthwhile to note that the MWCSA behaves well for all small-sized instances and gives an average result of 0.62% deviation from the optimal solution. Specially, eleven of these instances can be solved optimally by the proposed heuristic. Its execution time is far

**Table 2:** Comparison of CPLEX results with the proposed algorithms for small-sized instances.

$n$	$m$	CPLEX		MWCSA		VNS	
		Objective value (gap)	Time (sec)	Objective value	RPD	Objective value	RPD
6	2	582	1.48	<b>582</b>	0.00%	<b>582</b>	0.00%
	3	360	2.47	370	2.78%	<b>360</b>	0.00%
8	2	859	103.55	<b>859</b>	0.00%	<b>859</b>	0.00%
	3	513	122.20	<b>513</b>	0.00%	<b>513</b>	0.00%
10	2	1284	24632.09	1308	1.87%	<b>1284</b>	0.00%
	3	1190	144761.59	1216	1.59%	<b>1190</b>	0.00%
*the deteriorating date with the interval $U(0, D_{0.5})$							
6	2	430	1.50	<b>430</b>	0.00%	<b>430</b>	0.00%
	3	392	2.05	<b>392</b>	0.00%	<b>392</b>	0.00%
8	2	835	80.70	<b>835</b>	0.00%	<b>835</b>	0.00%
	3	590	233.25	<b>590</b>	0.00%	<b>590</b>	0.00%
10	2	1186	17676.28	<b>1186</b>	0.00%	<b>1186</b>	0.00%
	3	794	62828.11	799	0.63%	<b>794</b>	0.00%
*the deteriorating date with the interval $U[D_{0.5}, D]$							
6	2	446	1.63	<b>446</b>	0.00%	<b>446</b>	0.00%
	3	335	1.92	<b>335</b>	0.00%	<b>335</b>	0.00%
8	2	934	102.61	939	0.54%	<b>934</b>	0.00%
	3	486	205.08	491	1.03%	<b>486</b>	0.00%
10	2	1161	22216.36	1193	2.76%	<b>1161</b>	0.00%
	3	638	51367.00	<b>638</b>	0.00%	<b>638</b>	0.00%
*the deteriorating date with the interval $U(0, D]$							
Average					0.62%		0.00%

**Figure 2:** Means plot and Fisher LSD intervals for the different levels of parameter  $\varphi$  (the significance level  $\alpha$  is 0.05).

**Table 3:** Test results of the algorithms on the instances with the interval  $U(0, D_{0.5}]$ .

Problem number	<i>n</i>	<i>m</i>	SRF		MWCSA		VNS		VNS + MWCSA		
			Objective value (A)	Objective value (B)	(B)/(A)	Objective value (C)	(C)/(A)	Time (sec)	Objective value (D)	(D)/(A)	Time (sec)
1	20	2	5859	5062	86.40%	5001	85.36%	3.71	5001	85.36%	3.90
2		4	3274	2589	79.08%	2587	79.02%	3.58	2583	78.89%	3.77
3		6	1792	1676	93.53%	1655	92.35%	3.65	1655	92.35%	3.71
4		8	1299	1247	96.00%	1199	92.30%	3.68	1199	92.30%	3.73
5		10	1324	1309	98.87%	1269	95.85%	3.71	1267	95.69%	3.76
6	40	2	17056	14013	82.16%	13361	78.34%	23.00	13307	78.02%	23.73
7		4	11530	9614	83.38%	9513	82.51%	23.57	9508	82.46%	24.38
8		6	7833	6890	87.96%	6800	86.81%	23.05	6797	86.77%	23.56
9		8	4684	4299	91.78%	4201	89.69%	23.60	4198	89.62%	23.67
10		10	3858	3465	89.81%	3434	89.01%	24.15	3428	88.85%	24.71
11	60	2	47917	41030	85.63%	39204	81.82%	73.04	39149	81.70%	71.14
12		4	28547	22651	79.35%	22234	77.89%	72.34	22184	77.71%	67.62
13		6	17378	13616	78.35%	13504	77.71%	70.08	13474	77.53%	70.38
14		8	12577	10288	81.80%	10122	80.48%	69.37	10121	80.47%	72.68
15		10	10346	8695	84.04%	8573	82.86%	71.59	8548	82.62%	68.72
16	80	2	73047	66321	90.79%	58736	80.41%	157.50	58392	79.94%	181.86
17		4	41378	33779	81.64%	32734	79.11%	160.57	32687	79.00%	132.71
18		6	27955	22144	79.21%	21679	77.55%	152.83	21671	77.52%	142.25
19		8	20569	16200	78.76%	16055	78.05%	157.20	16013	77.85%	136.14
20		10	15479	12720	82.18%	12569	81.20%	160.32	12516	80.86%	152.90
21	100	2	126101	108828	86.30%	98779	78.33%	288.47	98026	77.74%	290.95
22		4	58800	46684	79.39%	45214	76.89%	295.09	44858	76.29%	277.00
23		6	46629	35717	76.60%	34678	74.37%	307.01	34466	73.92%	240.82
24		8	35233	27602	78.34%	27275	77.41%	286.20	27143	77.04%	263.09
25		10	29084	22906	78.76%	22753	78.23%	299.92	22629	77.81%	303.41
Average			84.40%			82.14%			81.93%		

less than 1 second so that its log is not necessary. It can also be noticed that the solutions from the VNS algorithm are optimal for these small-sized instances. Meanwhile, the computational time of these instances given by VNS are not more than 1 second. This means that the VNS algorithm is effective in solving the scheduling problem under study and has significantly better optimization performance than the others.

Tables 3, 4, and 5 show the results of large-sized instances where the deteriorating dates are generated from three different intervals. In order to compare the performance of the proposed algorithms, the ratios of the results of a given algorithm to the values obtained by SRF were calculated out. For the instances with the deteriorating dates generated from interval  $U(0, D_{0.5}]$ , VNS behaves slightly better than MWCSA. The difference in average rate between two algorithms is only 2.26%. For other two types of instances with intervals  $U[D_{0.5}, D]$  and  $U(0, D]$ , the VNS and VNS + MWCSA strikingly outperformed the MWCSA heuristic because of its larger search space. As a rule of thumb, a given algorithm takes possession of the better initial solution and should yield better result. However, the hybrid

**Table 4:** Test results of the algorithms on the instances with the interval  $U[D_{0.5}, D]$ .

Problem number	$n$	$m$	SRF		MWCSA		VNS		VNS + MWCSA		
			Objective value (A)	Objective value (B)	(B)/(A)	Objective value (C)	(C)/(A)	Time (sec)	Objective value (D)	(D)/(A)	Time (sec)
1	20	2	5065	4936	97.45%	4576	90.35%	3.68	4576	90.35%	3.81
2		4	2405	2186	90.89%	2145	89.19%	3.66	2145	89.19%	3.70
3		6	1420	1298	91.41%	1296	91.27%	3.67	1296	91.27%	3.76
4		8	1172	1146	97.78%	1146	97.78%	3.73	1146	97.78%	3.75
5		10	1567	1528	97.51%	1528	97.51%	3.74	1528	97.51%	3.73
6	40	2	17081	18299	107.13%	15326	89.73%	23.57	15316	89.67%	26.33
7		4	10157	9559	94.11%	9133	89.92%	24.56	9133	89.92%	24.69
8		6	5847	5040	86.20%	4944	84.56%	24.00	4943	84.54%	23.70
9		8	3708	3376	91.05%	3334	89.91%	23.87	3334	89.91%	24.17
10		10	4141	3772	91.09%	3733	90.15%	24.67	3733	90.15%	24.67
11	60	2	37202	40606	109.15%	32409	87.12%	74.86	32406	87.11%	85.56
12		4	18904	18275	96.67%	16831	89.03%	73.84	16830	89.03%	76.31
13		6	14321	13618	95.09%	13122	91.63%	72.86	13113	91.56%	74.14
14		8	9488	8980	94.65%	8792	92.66%	77.12	8792	92.66%	75.39
15		10	10032	9203	91.74%	8991	89.62%	77.29	8988	89.59%	78.05
16	80	2	61930	68725	110.97%	54652	88.25%	163.31	54651	88.25%	205.40
17		4	27137	28577	105.31%	24790	91.35%	167.83	24776	91.30%	169.78
18		6	29003	26914	92.80%	25022	86.27%	175.63	25021	86.27%	177.12
19		8	17291	15715	90.89%	15037	86.96%	162.31	15032	86.94%	164.65
20		10	12900	12215	94.69%	11860	91.94%	168.39	11855	91.90%	169.90
21	100	2	110757	128605	116.11%	98384	88.83%	286.03	98358	88.81%	386.87
22		4	50773	49696	97.88%	44153	86.96%	310.59	44152	86.96%	315.32
23		6	30745	31697	103.10%	27834	90.53%	319.17	27833	90.53%	319.76
24		8	26562	25254	95.08%	23576	88.76%	320.56	23573	88.75%	288.13
25		10	24330	22469	92.35%	21508	88.40%	318.57	21506	88.39%	327.75
Average			97.24%			89.95%			89.93%		

algorithm, with longer run times, does not perform noticeably better than the VNS. It is chiefly because the VNS carries out enough neighborhood searches before the stopping criterion meets. The rates of three different types of instances were plotted into smooth curves shown in Figures 3, 4, and 5. It can be seen from those figures that the performance of MWCSA heuristic is not particularly robust as regards the distribution of the deteriorating date. Alternatively, VNS and VNS + MWCSA are statistically better than MWCSA for three types of instances.

## 6. Conclusions

This paper considers the identical parallel machines scheduling problem with step-deteriorating jobs. The processing time of each job is a step function of its starting time and

**Table 5:** Test results of the algorithms on the instances with the interval  $U(0, D]$ .

Problem number	$n$	$m$	SRF		MWCSA		VNS		VNS + MWCSA		
			Objective value (A)	Objective value (B)	(B)/(A)	Objective value (C)	(C)/(A)	Time (sec)	Objective value (D)	(D)/(A)	Time (sec)
1	20	2	4670	4681	100.24%	4103	87.86%	3.65	4100	87.79%	3.75
2		4	3762	3034	80.65%	2981	79.24%	3.70	2980	79.21%	3.69
3		6	2179	2077	95.32%	2048	93.99%	3.69	2048	93.99%	3.77
4		8	1166	1126	96.57%	1070	91.77%	3.66	1067	91.51%	3.62
5		10	1056	1037	98.20%	1037	98.20%	3.73	1037	98.20%	3.74
6	40	2	19603	19675	100.37%	17212	87.80%	23.44	17174	87.61%	25.33
7		4	9990	9373	93.82%	8871	88.80%	23.67	8866	88.75%	23.89
8		6	7720	6942	89.92%	6786	87.90%	24.45	6786	87.90%	24.49
9		8	6087	5481	90.04%	5442	89.40%	23.81	5433	89.26%	24.45
10		10	5067	4356	85.97%	4321	85.28%	24.70	4321	85.28%	24.33
11	60	2	40513	40049	98.85%	33051	81.58%	69.64	33014	81.49%	80.71
12		4	19380	17741	91.54%	15364	79.28%	69.82	15358	79.25%	75.96
13		6	15350	12530	81.63%	11737	76.46%	75.05	11722	76.36%	72.58
14		8	11725	10139	86.47%	9889	84.34%	74.55	9874	84.21%	73.80
15		10	9949	8786	88.31%	8699	87.44%	73.75	8685	87.30%	72.12
16	80	2	74532	75705	101.57%	61269	82.20%	152.41	61263	82.20%	199.07
17		4	41890	37588	89.73%	32807	78.32%	171.50	32798	78.30%	170.17
18		6	27990	23587	84.27%	21803	77.90%	155.46	21754	77.72%	160.16
19		8	19278	16845	87.38%	16162	83.84%	161.03	16117	83.60%	158.46
20		10	16916	14285	84.45%	13804	81.60%	169.19	13802	81.59%	170.30
21	100	2	108067	107291	99.28%	81743	75.64%	281.41	81426	75.35%	373.51
22		4	59031	56349	95.46%	48136	81.54%	294.00	48135	81.54%	325.10
23		6	43204	38473	89.05%	34873	80.72%	297.22	34774	80.49%	323.49
24		8	30485	26263	86.15%	23977	78.65%	295.55	23957	78.59%	308.85
25		10	19559	17986	91.96%	16919	86.50%	305.73	16853	86.16%	304.20
Average			91.49%			84.25%			84.15%		

a deteriorating date that is individual to all jobs. The problem is to determine the allocation of jobs to machines as well as the sequence of the jobs assigned to each machine for the criteria of minimizing the total completion time. A mathematical model for this problem has been formulated. Since the problem under study is NP-complete, it is impossible to solve large-sized instances to optimality. To solve the tackled problem, a heuristic MWCSA and a VNS are proposed to obtain the near optimal solutions. In order to further improve the quality of solution, the heuristic MWCSA has been hybridized with the VNS algorithm and implemented to provide a good initial solution. Numerical experiments are conducted on small- and large-sized instances. Computational results show that MWCSA produces some good solutions compared to CPLEX, but the performance is greatly affected by the distribution of the deteriorating date. In contrast, VNS and VNS + WMCSA are robust as regards three types of instances. Therefore, fairly good solutions can be obtained by the proposed methods within reasonable amount of time.

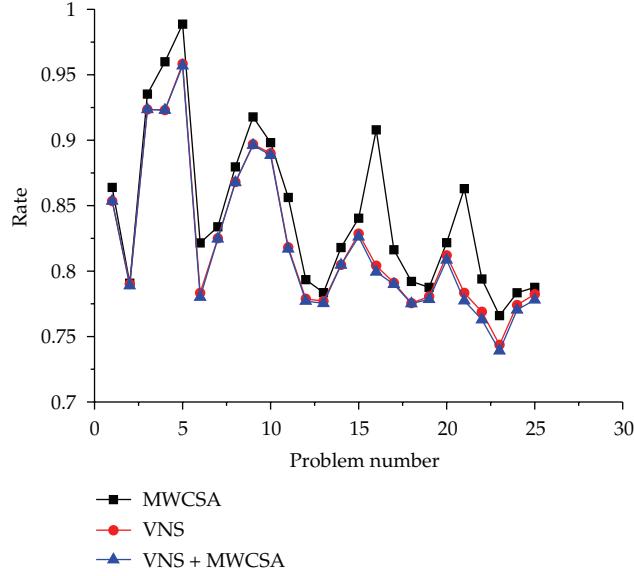


Figure 3: Comparison of rates between the three algorithms for the case with the interval  $U(0, D_{0.5}]$ .

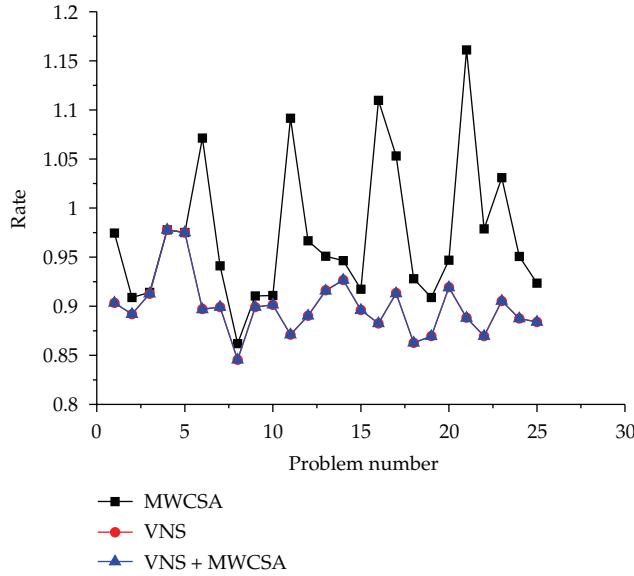


Figure 4: Comparison of rates between the three algorithms for the case with the interval  $U[D_{0.5}, D]$ .

For further study, it is worth considering the setup times between jobs for the problem of scheduling jobs with piecewise-deterioration on multimachine. In addition, other efficient constructive heuristics and neighborhood properties are worthwhile to investigate for improving the quality of solutions.

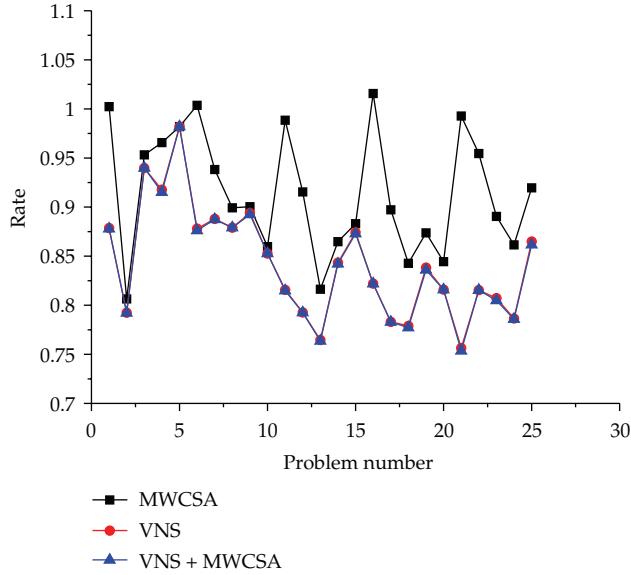


Figure 5: Comparison of rates between the three algorithms for the case with the interval  $U(0, D]$ .

## Acknowledgments

This work is partially supported by the National Natural Science Foundation of China under Grant no. 51175442, the Youth Foundation for Humanities and Social Sciences of Ministry of Education of China under Grant no. 12YJCZH296, the Ph.D. Programs Foundation of Ministry of Education of China under Grant no. 200806131014, and the Fundamental Research Funds for the Central Universities under Grant nos. SWJTU09CX022, 2010ZT03.

## References

- [1] S. Browne and U. Yechiali, "Scheduling deteriorating jobs on a single processor," *Operations Research*, vol. 38, no. 3, pp. 495–498, 1990.
- [2] J. N. D. Gupta and S. K. Gupta, "Single facility scheduling with nonlinear processing times," *Computers and Industrial Engineering*, vol. 14, no. 4, pp. 387–393, 1988.
- [3] A. Bachman and A. Janiak, "Minimizing maximum lateness under linear deterioration," *European Journal of Operational Research*, vol. 126, no. 3, pp. 557–566, 2000.
- [4] A. Bachman, A. Janiak, and M. Y. Kovalyov, "Minimizing the total weighted completion time of deteriorating jobs," *Information Processing Letters*, vol. 81, no. 2, pp. 81–84, 2002.
- [5] C.-C. Wu, W.-C. Lee, and Y.-R. Shiao, "Minimizing the total weighted completion time on a single machine under linear deterioration," *International Journal of Advanced Manufacturing Technology*, vol. 33, no. 11-12, pp. 1237–1243, 2007.
- [6] J.-B. Wang, L. Lin, and F. Shan, "Single-machine group scheduling problems with deteriorating jobs," *International Journal of Advanced Manufacturing Technology*, vol. 39, no. 7-8, pp. 808–812, 2008.
- [7] J.-B. Wang, C. T. Ng, and T. C. E. Cheng, "Single-machine scheduling with deteriorating jobs under a series-parallel graph constraint," *Computers & Operations Research*, vol. 35, no. 8, pp. 2684–2693, 2008.
- [8] M. M. Mazdeh, F. Zaerpour, A. Zareei, and A. Hajinezhad, "Parallel machines scheduling to minimize job tardiness and machine deteriorating cost with deteriorating jobs," *Applied Mathematical Modelling*, vol. 34, no. 6, pp. 1498–1510, 2010.

- [9] L. Sun, L. Sun, K. Cui, and J. B. Wang, "A note on flow shop scheduling problems with deteriorating jobs on no-idle dominant machines," *European Journal of Operational Research*, vol. 200, no. 1, pp. 309–311, 2010.
- [10] D. Wang and J. B. Wang, "Single-machine scheduling with simple linear deterioration to minimize earliness penalties," *International Journal of Advanced Manufacturing Technology*, vol. 46, no. 1–4, pp. 285–290, 2010.
- [11] J.-B. Wang, "Flow shop scheduling with deteriorating jobs under dominating machines to minimize makespan," *International Journal of Advanced Manufacturing Technology*, vol. 48, no. 5–8, pp. 719–723, 2010.
- [12] C.-M. Wei and J.-B. Wang, "Single machine quadratic penalty function scheduling with deteriorating jobs and group technology," *Applied Mathematical Modelling*, vol. 34, no. 11, pp. 3642–3647, 2010.
- [13] X. Huang and M.-Z. Wang, "Parallel identical machines scheduling with deteriorating jobs and total absolute differences penalties," *Applied Mathematical Modelling*, vol. 35, no. 3, pp. 1349–1353, 2011.
- [14] J.-B. Wang and C. Wang, "Single-machine due-window assignment problem with learning effect and deteriorating jobs," *Applied Mathematical Modelling*, vol. 35, no. 8, pp. 4017–4022, 2011.
- [15] J.-B. Wang, J.-J. Wang, and P. Ji, "Scheduling jobs with chain precedence constraints and deteriorating jobs," *Journal of the Operational Research Society*, vol. 62, no. 9, pp. 1765–1770, 2011.
- [16] J.-B. Wang and C.-M. Wei, "Parallel machine scheduling with a deteriorating maintenance activity and total absolute differences penalties," *Applied Mathematics and Computation*, vol. 217, no. 20, pp. 8093–8099, 2011.
- [17] S.-H. Yang and J.-B. Wang, "Minimizing total weighted completion time in a two-machine flow shop scheduling under simple linear deterioration," *Applied Mathematics and Computation*, vol. 217, no. 9, pp. 4819–4826, 2011.
- [18] J.-B. Wang and M.-Z. Wang, "Single-machine scheduling with nonlinear deterioration," *Optimization Letters*, vol. 6, no. 1, pp. 87–98, 2012.
- [19] J.-B. Wang, M.-Z. Wang, and P. Ji, "Single machine total completion time minimization scheduling with a time-dependent learning effect and deteriorating jobs," *International Journal of Systems Science*, vol. 43, no. 5, pp. 861–868, 2012.
- [20] B. Alidaee and N. K. Womer, "Scheduling with time dependent processing times: review and extensions," *Journal of the Operational Research Society*, vol. 50, no. 7, pp. 711–720, 1999.
- [21] T. C. E. Cheng, Q. Ding, and B. M. T. Lin, "A concise survey of scheduling with time-dependent processing times," *European Journal of Operational Research*, vol. 152, no. 1, pp. 1–13, 2004.
- [22] P. S. Sundararaghavan and A. S. Kannanathur, "Single machine scheduling with start time dependent processing times: some solvable cases," *European Journal of Operational Research*, vol. 78, no. 3, pp. 394–403, 1994.
- [23] W. Kubiak and S. van de Velde, "Scheduling deteriorating jobs to minimize makespan," *Naval Research Logistics*, vol. 45, no. 5, pp. 511–523, 1998.
- [24] T. C. E. Cheng and Q. Ding, "Single machine scheduling with step-deteriorating processing times," *European Journal of Operational Research*, vol. 134, no. 3, pp. 623–630, 2001.
- [25] P. Hansen and N. Mladenović, "Variable neighborhood search: principles and applications," *European Journal of Operational Research*, vol. 130, no. 3, pp. 449–467, 2001.
- [26] N. Mladenović and P. Hansen, "Variable neighborhood search," *Computers & Operations Research*, vol. 24, no. 11, pp. 1097–1100, 1997.
- [27] A. S. Kannanathur and S. K. Gupta, "Minimizing the makespan with late start penalties added to processing times in a single facility scheduling problem," *European Journal of Operational Research*, vol. 47, no. 1, pp. 56–64, 1990.
- [28] G. Mosheiov, "Scheduling jobs with step-deterioration; Minimizing makespan on a single- and multi-machine," *Computers and Industrial Engineering*, vol. 28, no. 4, pp. 869–879, 1995.
- [29] A. A. K. Jeng and B. M. T. Lin, "Makespan minimization in single-machine scheduling with step-deterioration of processing times," *Journal of the Operational Research Society*, vol. 55, no. 3, pp. 247–256, 2004.
- [30] A. A. K. Jeng and B. M. T. Lin, "Minimizing the total completion time in single-machine scheduling with step-deteriorating jobs," *Computers and Operations Research*, vol. 32, no. 3, pp. 521–536, 2005.
- [31] C.-C. He, C.-C. Wu, and W.-C. Lee, "Branch-and-bound and weight-combination search algorithms

- for the total completion time problem with step-deteriorating jobs," *Journal of the Operational Research Society*, vol. 60, no. 12, pp. 1759–1766, 2009.
- [32] J. Layegh, F. Jolai, and M. S. Amalnik, "A memetic algorithm for minimizing the total weighted completion time on a single machine under step-deterioration," *Advances in Engineering Software*, vol. 40, no. 10, pp. 1074–1077, 2009.
  - [33] M. Y. Kovalyov and W. Kubiak, "A fully polynomial approximation scheme for minimizing makespan of deteriorating jobs," *Journal of Heuristics*, vol. 3, no. 4, pp. 287–297, 1998.
  - [34] T. C. E. Cheng, Q. Ding, M. Y. Kovalyov, A. Bachman, and A. Janiak, "Scheduling jobs with piecewise linear decreasing processing times," *Naval Research Logistics*, vol. 50, no. 6, pp. 531–554, 2003.
  - [35] G. Moslehi and A. Jafari, "Minimizing the number of tardy jobs under piecewise-linear deterioration," *Computers and Industrial Engineering*, vol. 59, no. 4, pp. 573–584, 2010.
  - [36] S. R. Gupta and J. S. Smith, "Algorithms for single machine total tardiness scheduling with sequence dependent setups," *European Journal of Operational Research*, vol. 175, no. 2, pp. 722–739, 2006.
  - [37] M. R. D. Paula, M. G. Ravetti, G. R. Mateus, and P. M. Pardalos, "Solving parallel machines scheduling problems with sequence-dependent setup times using variable neighbourhood search," *IMA Journal of Management Mathematics*, vol. 18, no. 2, pp. 101–115, 2007.
  - [38] D. Anghinolfi and M. Paolucci, "Parallel machine total tardiness scheduling with a new hybrid metaheuristic approach," *Computers & Operations Research*, vol. 34, no. 11, pp. 3471–3490, 2007.
  - [39] R. Driessel and L. Mönch, "Variable neighborhood search approaches for scheduling jobs on parallel machines with sequence-dependent setup times, precedence constraints, and ready times," *Computers and Industrial Engineering*, vol. 61, no. 2, pp. 336–345, 2011.
  - [40] J. Behnamian, M. Zandieh, and S. M. T. F. Ghomi, "Parallel-machine scheduling problems with sequence-dependent setup times using an ACO, SA and VNS hybrid algorithm," *Expert Systems with Applications*, vol. 36, no. 6, pp. 9637–9644, 2009.
  - [41] C.-L. Chen and C.-L. Chen, "Hybrid metaheuristics for unrelated parallel machine scheduling with sequence-dependent setup times," *International Journal of Advanced Manufacturing Technology*, vol. 43, no. 1-2, pp. 161–169, 2009.
  - [42] J. Gao, L. Sun, and M. Gen, "A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems," *Computers & Operations Research*, vol. 35, no. 9, pp. 2892–2907, 2008.
  - [43] V. Roshanaei, B. Naderi, F. Jolaib, and M. Khalili, "A variable neighborhood search for job shop scheduling with set-up times to minimize makespan," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 654–661, 2009.
  - [44] M. A. Adibi, M. Zandieh, and M. Amiri, "Multi-objective scheduling of dynamic job shop using variable neighborhood search," *Expert Systems with Applications*, vol. 37, no. 1, pp. 282–287, 2010.
  - [45] M. Amiri, M. Zandieh, M. Yazdani, and A. Bagheri, "A variable neighbourhood search algorithm for the flexible job-shop scheduling problem," *International Journal of Production Research*, vol. 48, no. 19, pp. 5671–5689, 2010.
  - [46] P. Hansen, N. Mladenović, and J. A. M. Pérez, "Variable neighbourhood search: methods and applications," *Annals of Operations Research*, vol. 175, pp. 367–407, 2010.
  - [47] M. L. Pinedo, *Scheduling: Theory, Algorithms and Systems*, Springer, New York, NY, USA, 2008.
  - [48] C. Ferreira, "Combinatorial optimization by gene expression programming: inversion revisited," in *Proceedings of the Argentine Symposium on Artificial Intelligence*, pp. 160–174, Santa Fe, NM, USA, 2002.

*Research Article*

## Global Sufficient Optimality Conditions for a Special Cubic Minimization Problem

Xiaomei Zhang,<sup>1</sup> Yanjun Wang,<sup>1</sup> and Weimin Ma<sup>2</sup>

<sup>1</sup> Department of Applied Mathematics, Shanghai University of Finance and Economics, Shanghai 200433, China

<sup>2</sup> School of Economics and Management, Tongji University, Shanghai 200092, China

Correspondence should be addressed to Weimin Ma, mawm@tongji.edu.cn

Received 9 February 2012; Accepted 11 June 2012

Academic Editor: Jianming Shi

Copyright © 2012 Xiaomei 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.

We present some sufficient global optimality conditions for a special cubic minimization problem with box constraints or binary constraints by extending the global subdifferential approach proposed by V. Jeyakumar et al. (2006). The present conditions generalize the results developed in the work of V. Jeyakumar et al. where a quadratic minimization problem with box constraints or binary constraints was considered. In addition, a special diagonal matrix is constructed, which is used to provide a convenient method for justifying the proposed sufficient conditions. Then, the reformulation of the sufficient conditions follows. It is worth noting that this reformulation is also applicable to the quadratic minimization problem with box or binary constraints considered in the works of V. Jeyakumar et al. (2006) and Y. Wang et al. (2010). Finally some examples demonstrate that our optimality conditions can effectively be used for identifying global minimizers of the certain nonconvex cubic minimization problem.

### 1. Introduction

Consider the following cubic minimization problem with box constraints:

$$\begin{aligned} \min \quad & f(x) = b^T x^3 + \frac{1}{2} x^T A x + a^T x, \\ \text{s.t.} \quad & x \in D = \prod_{i=1}^n [u_i, v_i], \end{aligned} \tag{CP1}$$

where  $u_i, v_i \in R$ ,  $u_i \leq v_i$ ,  $i = 1, 2, \dots, n$ , and  $a = (a_1, \dots, a_n)^T \in R^n$ ,  $b = (b_1, \dots, b_n)^T \in R^n$ ,  $A \in S^n$ , where  $S^n$  is the set of all symmetric  $n \times n$  matrices.  $x^3$  that means  $(x_1^3, \dots, x_n^3)^T$ .

The cubic optimization problem has spawned a variety of applications, especially in cubic polynomial approximation optimization [1], convex optimization [2], engineering design, and structural optimization [3]. Moreover, research results about cubic optimization problem can be applied to quadratic programming problems, which have been widely studied because of their broad applications, to enrich quadratic programming theory.

Several general approaches can be used to establish optimality conditions for solutions to optimization problems. These approaches can be broadly classified into three groups: convex duality theory [4], local subdifferentials by linear functions [5–7], and global  $L$ -subdifferential and  $L$ -normal cone by quadratic functions [8–11]. The third approach, which we extend in this paper, is often adopted to develop optimality conditions for special optimization forms: quadratic minimizations with box or binary constraints, quadratic minimization with quadratic constraints, bivalent quadratic minimization with inequality constraints, and so forth.

In this paper, we consider the cubic minimization problem, which generalizes the quadratic functions frequently considered in the mentioned papers. The proof method is based on extending the global  $L$ -subdifferentials by quadratic functions [8, 12] to cubic functions. We show how an  $L$ -subdifferential can be explicitly calculated for cubic functions and then develop the global sufficient optimality conditions for (CP1). We also derive the global optimality conditions for special cubic minimization problems with binary constraints. But when we use the sufficient conditions, we have to determine whether a diagonal matrix  $Q$  exists. It is hard to identify whether the matrix  $Q$  exists. So we rewrite the sufficient conditions in an other way through constructing a certain diagonal matrix. This method is applicable to the quadratic minimization problem with box or binary constraints considered in [8, 12].

This paper is organized as follows. Section 2 presents the notions of  $L$ -subdifferentials and develops the sufficient global optimality condition for (CP1). The global optimality condition for special cubic minimization with binary constraints is presented in Section 3. In Section 4, numerical examples are given to illustrate the effectiveness of the proposed global optimality conditions.

## 2. $L$ -Subdifferentials and Sufficient Conditions

In this section, basic definitions and notations that will be used throughout the paper are given. The real line is denoted by  $R$  and the  $n$ -dimensional Euclidean space is denoted by  $R^n$ . For vectors  $x, y \in R^n$ ,  $x \geq y$  means that  $x_i \geq y_i$ , for  $i = 1, \dots, n$ .  $A \succeq B$  means that the matrix  $A - B$  is a positive semidefinite. A diagonal matrix with diagonal elements  $\alpha_1, \dots, \alpha_n$  is denoted by  $\text{diag}(\alpha_1, \dots, \alpha_n)$ . Let  $L$  be a set of real-valued functions defined on  $R^n$ .

*Definition 2.1* ( $L$ -subdifferentials [13]). Let  $L$  be a set of real-valued functions. Let  $f : R^n \rightarrow R$ . An element  $l \in L$  is called an  $L$ -subgradient of  $f$  at a point  $x_0 \in R^n$  if

$$f(x) \geq f(x_0) + l(x) - l(x_0), \quad \forall x \in R^n. \quad (2.1)$$

The set  $\partial_L f(x)$  of all  $L$ -subgradients of  $f$  at  $x_0$  is referred to as  $L$ -subdifferential of  $f$  at  $x_0$ .

Throughout the rest of the paper, we use the specific choice of  $L$  defined by

$$L = \left\{ b^T x^3 + \frac{1}{2} x^T Q x + \beta^T x \mid Q = \text{diag}(\alpha_1, \dots, \alpha_n), \alpha_i \in R, \beta \in R^n \right\}. \quad (2.2)$$

**Proposition 2.2.** Let  $f(x) = b^T x^3 + (1/2)x^T A x + a^T x$  and  $\bar{x} = (\bar{x}_1, \dots, \bar{x}_n)^T \in R^n$ . Then

$$\partial_L f(\bar{x}) = \left\{ b^T x^3 + \frac{1}{2} x^T Q x + \beta^T x \middle| \begin{array}{l} A - Q \succeq 0, Q = \text{diag}(\alpha_1, \dots, \alpha_n) \\ \beta = (A - Q)\bar{x} + a, \alpha_i \in R \end{array} \right\}. \quad (2.3)$$

*Proof.* Suppose that there exists a diagonal matrix  $Q = \text{diag}(\alpha_1, \dots, \alpha_n)$ , such that  $A - Q \succeq 0$ . Let

$$l(x) = b^T x^3 + \frac{1}{2} x^T Q x + \beta^T x, \quad \beta = (A - Q)\bar{x} + a. \quad (2.4)$$

Then it suffices to prove that  $l(x) \in \partial_L f(\bar{x})$ . Let

$$\phi(x) = f(x) - l(x) = \frac{1}{2} x^T (A - Q)x + (a - \beta)^T x. \quad (2.5)$$

Since  $\nabla^2 \phi(x) = A - Q \succeq 0$ , for all  $x \in R^n$ , we know that  $\phi(x)$  is a convex function on  $R^n$ . Note that  $\nabla \phi(\bar{x}) = (A - Q)\bar{x} + (a - \beta) = 0$ , and so  $\bar{x}$  is a global minimizer of  $\phi(x)$ , that is,  $\phi(x) \geq \phi(\bar{x})$ , for all  $x \in R^n$ . This means that  $l(x) \in \partial_L f(\bar{x})$ .

Next we prove the converse.

Let  $l(x) \in \partial_L f(\bar{x})$ ,  $l(x) = b^T x^3 + (1/2)x^T Qx + \beta^T x$ . By definition,

$$f(x) \geq f(\bar{x}) + l(x) - l(\bar{x}), \quad \forall x \in R^n. \quad (2.6)$$

Hence

$$\phi(x) = f(x) - l(x) = \frac{1}{2} x^T (A - Q)x + (a - \beta)^T x \geq f(\bar{x}) - l(\bar{x}), \quad \forall x \in R^n. \quad (2.7)$$

Thus,  $\bar{x}$  is a global minimizer of  $\phi(x)$ . So,  $\nabla \phi(\bar{x}) = 0$  and  $\nabla^2 \phi(\bar{x}) \succeq 0$ , that is,

$$A - Q \succeq 0, \quad (A - Q)\bar{x} + (a - \beta) = 0, \quad (2.8)$$

hence  $\beta = (A - Q)\bar{x} + a$ . □

For  $\bar{x} = (\bar{x}_1, \dots, \bar{x}_n)^T \in D$ , define

$$\tilde{x}_i = \begin{cases} -1 & \text{if } \bar{x}_i = u_i, \\ 1 & \text{if } \bar{x}_i = v_i, \\ (A\bar{x})_i + a_i & \text{if } \bar{x}_i \in (u_i, v_i). \end{cases} \quad (2.9)$$

$$\tilde{X} = \text{diag}(\tilde{x}_1, \dots, \tilde{x}_n).$$

For  $Q = \text{diag}(\alpha_1, \dots, \alpha_n)$ ,  $\alpha_i \in R$ ,  $i = 1, \dots, n$ , define

$$\begin{aligned}\hat{\alpha}_i &= \min\{0, \alpha_i\}, \\ \hat{Q} &= \text{diag}(\hat{\alpha}_1, \dots, \hat{\alpha}_n).\end{aligned}\tag{2.10}$$

By Proposition 2.2, we obtain the following sufficient global optimality condition for (CP1).

**Theorem 2.3.** For (CP1), let  $\bar{x} = (\bar{x}_1, \dots, \bar{x}_n)^T \in D$  and  $u = (u_1, \dots, u_n)^T$ ,  $v = (v_1, \dots, v_n)^T$ . Suppose that there exists a diagonal matrix  $Q = \text{diag}(\alpha_1, \dots, \alpha_n)$ ,  $\alpha_i \in R$ ,  $i = 1, \dots, n$ , such that  $A - Q \succeq 0$ , and for all  $x \in D$ ,  $b_i(x_i - \bar{x}_i) \geq 0$ , ( $i = 1, \dots, n$ ). If

$$\tilde{X}(A\bar{x} + a) - \frac{1}{2}\hat{Q}(v - u) \leq 0,\tag{2.11}$$

then  $\bar{x}$  is a global minimizer of problem (CP1).

*Proof.* Suppose that condition (2.11) holds. Let

$$\begin{aligned}l(x) &= b^T x^3 + \frac{1}{2}x^T Q x + \beta^T x, \\ \beta &= (A - Q)\bar{x} + a.\end{aligned}\tag{2.12}$$

Then, by Proposition 2.2,  $l(x) \in \partial_L f(\bar{x})$ , that is,

$$f(x) - f(\bar{x}) \geq l(x) - l(\bar{x}), \quad \forall x \in R^n.\tag{2.13}$$

Obviously if  $l(x) - l(\bar{x}) \geq 0$  for each  $x \in D$ , then  $\bar{x}$  is a global minimizer of (CP1).

Note that

$$l(x) - l(\bar{x}) = \sum_{i=1}^n \frac{\alpha_i}{2} (x_i - \bar{x}_i)^2 + (A\bar{x} + a)^T (x - \bar{x}) + b^T (x^3 - \bar{x}^3).\tag{2.14}$$

If each term in the right side of the above equation satisfies

$$\frac{\alpha_i}{2} (x_i - \bar{x}_i)^2 + (A\bar{x} + a)_i (x_i - \bar{x}_i) + b_i (x_i^3 - \bar{x}_i^3) \geq 0, \quad i = 1, \dots, n, \quad x_i \in [u_i, v_i],\tag{2.15}$$

then, from (2.14), it holds that  $l(x) - l(\bar{x}) \geq 0$ . So  $\bar{x}$  is a global minimizer of  $l(x)$  over box constraints.

On the other hand, suppose that  $\bar{x}$  is a global minimizer of  $l(x)$ ,  $x \in D$ . Then it holds that

$$l(x) - l(\bar{x}) = \sum_{i=1}^n \frac{\alpha_i}{2} (x_i - \bar{x}_i)^2 + (A\bar{x} + a)^T (x - \bar{x}) + b^T (x^3 - \bar{x}^3) \geq 0, \quad \forall x \in D.\tag{2.16}$$

When  $x$  is chosen as a special point  $\tilde{x} \in D$  as follows:

$$\tilde{x} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_{i-1}, x_i, \bar{x}_{i+1}, \dots, \bar{x}_n), \quad x_i \in [u_i, v_i], \quad x_i \neq \bar{x}_i, \quad i = 1, \dots, n, \quad (2.17)$$

we still have

$$l(\tilde{x}) - l(\bar{x}) = \frac{\alpha_i}{2} (x_i - \bar{x}_i)^2 + (A\bar{x} + a)^T (x - \bar{x}) + b^T (x^3 - \bar{x}^3) \geq 0, \quad i = 1, \dots, n. \quad (2.18)$$

This means that if  $\bar{x}$  is a global minimizer of  $l(x)$  over box constraints. Then (2.15) holds.

Combining the above discussion, we can conclude that  $\bar{x}$  is a global minimizer of  $l(x)$  over box constraints if and only if (2.15) holds. So next, we just need to prove (2.15) in order to show that  $\bar{x}$  is a global minimizer of  $l(x)$ .

We first see from (2.11), for each  $i = 1, \dots, n$ , that

$$-\frac{\hat{\alpha}_i}{2} (v_i - u_i) + \tilde{x}_i (A\bar{x} + a)_i \leq 0. \quad (2.19)$$

Since  $\hat{\alpha}_i \leq 0$ , then for each  $x_i \in [u_i, v_i]$ ,  $i = 1, \dots, n$ ,

$$-\frac{\hat{\alpha}_i}{2} (x_i - u_i) + \tilde{x}_i (A\bar{x} + a)_i \leq 0, \quad (2.20)$$

and

$$\frac{\hat{\alpha}_i}{2} (x_i - v_i) + \tilde{x}_i (A\bar{x} + a)_i \leq 0. \quad (2.21)$$

For each  $i = 1, \dots, n$ , we consider the following three cases.

*Case 1.* (If  $\tilde{x}_i \in (u_i, v_i)$ , then  $\tilde{x}_i = (A\bar{x} + a)_i$ ). By (2.20),

$$-\frac{\hat{\alpha}_i}{2} (x_i - u_i) + (A\bar{x} + a)_i^2 \leq 0. \quad (2.22)$$

So,  $\hat{\alpha}_i = 0$  and  $(A\bar{x} + a)_i = 0$ , and then

$$\begin{aligned} & \frac{\alpha_i}{2} (x_i - \bar{x}_i)^2 + (A\bar{x} + a)_i (x_i - \bar{x}_i) + b_i (x_i^3 - \bar{x}_i^3) \\ & \geq \frac{\hat{\alpha}_i}{2} (x_i - \bar{x}_i)^2 + b_i (x_i^3 - \bar{x}_i^3) \\ & = b_i (x_i - \bar{x}_i) (x_i^2 + x_i \bar{x}_i + \bar{x}_i^2) \\ & \geq 0. \end{aligned} \quad (2.23)$$

*Case 2.* (If  $\bar{x}_i = u_i$ , then  $\tilde{x}_i = -1$ ). By (2.20),

$$\frac{\hat{\alpha}_i}{2}(x_i - u_i) + (A\bar{x} + a)_i \geq 0. \quad (2.24)$$

So we have

$$\begin{aligned} & \frac{\alpha_i}{2}(x_i - \bar{x}_i)^2 + (A\bar{x} + a)_i(x_i - \bar{x}_i) + b_i(x_i^3 - \bar{x}_i^3) \\ & \geq \frac{\hat{\alpha}_i}{2}(x_i - u_i)^2 + (A\bar{x} + a)_i(x_i - u_i) + b_i(x_i^3 - u_i^3) \\ & = \left\{ \frac{\hat{\alpha}_i}{2}(x_i - u_i) + (A\bar{x} + a)_i \right\}(x_i - u_i) + b_i(x_i^3 - u_i^3) \\ & \geq 0. \end{aligned} \quad (2.25)$$

*Case 3.* (If  $\bar{x}_i = v_i$ , then  $\tilde{x}_i = 1$ ). By (2.21),

$$\frac{\hat{\alpha}_i}{2}(x_i - v_i) + (A\bar{x} + a)_i \leq 0. \quad (2.26)$$

Then

$$\begin{aligned} & \frac{\alpha_i}{2}(x_i - \bar{x}_i)^2 + (A\bar{x} + a)_i(x_i - \bar{x}_i) + b_i(x_i^3 - \bar{x}_i^3) \\ & \geq \frac{\hat{\alpha}_i}{2}(x_i - v_i)^2 + (A\bar{x} + a)_i(x_i - v_i) + b_i(x_i^3 - v_i^3) \\ & \geq 0. \end{aligned} \quad (2.27)$$

So, if condition (2.11) holds, then (2.15) holds. And, from (2.14), we can conclude that  $\bar{x}$  is a global minimizer of (CP1).  $\square$

Theorem 2.3 shows that the existence of diagonal matrix  $Q$  plays a crucial role because if this diagonal matrix  $Q$  does not exist, then we have no way to use this theorem. If the diagonal matrix  $Q$  exists, then the key problem is how to find it. These questions also exist in [8, 12].

The following corollary will answer the questions above.

**Corollary 2.4.** For (CP1), let  $\bar{x} \in D$ . Assume that, for all  $x \in D$ , it holds that  $b_i(x_i - \bar{x}_i) \geq 0$  ( $i = 1, \dots, n$ ). Then one has the following conclusion.

(1) When  $A$  is a positive semidefinite matrix, if

$$\tilde{X}(A\bar{x} + a) \leq \mathbf{0}, \quad (2.28)$$

then  $\bar{x}$  is a global minimizer of (CP1).

(2) When  $A$  is not a positive semidefinite matrix, if there exists an index  $i_0$ ,  $1 \leq i_0 \leq n$ , such that

$$\tilde{x}_{i_0}(A\bar{x} + a)_{i_0} > \mathbf{0}, \quad (2.29)$$

then there is no such diagonal matrix  $Q$  that meets the requirements of the Theorem 2.3.

(3) Let

$$\begin{aligned} \alpha_i &= \frac{2\tilde{x}_i(A\bar{x} + a)_i}{v_i - u_i}, \quad i = 1, \dots, n, \\ Q &= \text{diag}(\alpha_1, \dots, \alpha_n). \end{aligned} \quad (2.30)$$

When  $A$  is not a positive semidefinite matrix and the condition  $\tilde{X}(A\bar{x} + a) \leq \mathbf{0}$  holds, if  $A - Q \succeq 0$  holds, then  $\bar{x}$  is a global minimizer of (CP1). Otherwise, one can conclude that there is no such diagonal matrix  $Q$  that meets the requirements of the Theorem 2.3.

*Proof.* (1) Suppose that  $A \succeq 0$  and the condition  $\tilde{X}(A\bar{x} + a) \leq \mathbf{0}$  holds. Choosing  $Q = \hat{Q} = \mathbf{0}$ , by Theorem 2.3, we can conclude that  $\bar{x}$  is a global minimizer of (CP1).

(2) When  $A$  is not a positive semidefinite matrix, if there exists an index  $i_0$ ,  $1 \leq i_0 \leq n$ , such that

$$\tilde{x}_{i_0}(A\bar{x} + a)_{i_0} > 0, \quad (2.31)$$

then

$$\frac{2\tilde{x}_{i_0}(A\bar{x} + a)_{i_0}}{v_{i_0} - u_{i_0}} > 0. \quad (2.32)$$

Suppose there exists a diagonal matrix  $Q$  that meets all conditions in Theorem 2.3. Condition (2.11) can be rewritten in the following form:

$$\tilde{x}_i(A\bar{x} + a)_i - \frac{1}{2}\hat{\alpha}_i(v_i - u_i) \leq 0, \quad i = 1, \dots, n. \quad (2.33)$$

Then it follows that

$$\hat{\alpha}_i \geq \frac{2\tilde{x}_i(A\bar{x} + a)_i}{v_i - u_i}, \quad i = 1, \dots, n. \quad (2.34)$$

For the index  $i_0$ , we still have the following inequality:

$$\hat{\alpha}_{i_0} \geq \frac{2\tilde{x}_{i_0}(A\bar{x} + a)_{i_0}}{v_{i_0} - u_{i_0}} > 0. \quad (2.35)$$

This conflicts with the fact that  $\hat{\alpha}_i = \min\{0, \alpha_i\} \leq 0$ ,  $i = 1, \dots, n$ .

(3) Next we will consider the case that  $A$  is not a positive semidefinite matrix, and condition  $\tilde{X}(A\bar{x} + a) \leq \mathbf{0}$  holds.

We construct a diagonal matrix  $Q = \text{diag}(\alpha_1, \dots, \alpha_n)$  where  $\alpha_i = 2\tilde{x}_i(A\bar{x} + a)_i / (v_i - u_i)$ ,  $i = 1, \dots, n$ , and  $A - Q \succeq 0$ . Then it suffices to show that condition (2.11) in Theorem 2.3 hold.

Note that  $\tilde{x}_i(A\bar{x} + a)_i \leq 0$ ,  $i = 1, \dots, n$ , and then

$$\alpha_i = \frac{2\tilde{x}_i(A\bar{x} + a)_i}{v_i - u_i} \leq 0, \quad i = 1, \dots, n. \quad (2.36)$$

Since  $\alpha_i \leq 0$ , we have  $\hat{\alpha}_i = \alpha_i$ . So

$$\hat{\alpha}_i = \frac{2\tilde{x}_i(A\bar{x} + a)_i}{v_i - u_i} \leq 0, \quad i = 1, \dots, n. \quad (2.37)$$

Rewriting the above inequality, we have

$$\tilde{x}_i(A\bar{x} + a)_i - \frac{1}{2}\hat{\alpha}_i(v_i - u_i) = 0, \quad i = 1, \dots, n. \quad (2.38)$$

Apparently this means that the constructed diagonal matrix  $Q$  also satisfies condition (2.11). According to Theorem 2.3, we can conclude that  $\bar{x}$  is a global minimizer of (CP1).

If the constructed diagonal matrix  $Q$  does not meet the condition  $A - Q \succeq 0$ , then we can conclude that there is no such diagonal matrix  $Q$  that can meet the requirements of Theorem 2.3.

To show this, suppose that there exists a diagonal matrix  $Q^* = \text{diag}(\alpha_1^*, \dots, \alpha_n^*)$ , which satisfies  $A - Q^* \succeq 0$  and (2.11).

From (2.11), we have

$$0 \geq \alpha_i^* \geq \frac{2\tilde{x}_i(A\bar{x} + a)_i}{v_i - u_i} = \alpha_i, \quad i = 1, \dots, n. \quad (2.39)$$

Obviously if  $A - Q^* \succeq 0$ , then there must exist a diagonal matrix  $Q = \text{diag}(\alpha_1, \dots, \alpha_n)$  such that  $A - Q \succeq 0$ . This conflicts the assumption.  $\square$

We now consider a special case of (CP1):

$$\begin{aligned} \min \quad & f(x) = \sum_{i=1}^n b_i x_i^3 + \sum_{i=1}^n \frac{r_i}{2} x_i^2 + \sum_{i=1}^n a_i x_i \\ \text{s.t.} \quad & x \in D = \prod_{i=1}^n [u_i, v_i], \end{aligned} \quad (CP2)$$

where  $u_i, v_i, a_i, b_i, r_i \in R$  and  $u_i \leq v_i$ ,  $i = 1, 2, \dots, n$ .

**Corollary 2.5.** For (CP2), let  $\bar{x} \in D$ . If, for each  $i = 1, \dots, n$ ,

$$\begin{aligned} \tilde{x}_i(r_i\bar{x}_i + a_i) - \frac{1}{2}\hat{r}_i(v_i - u_i) &\leq 0, \\ b_i(x_i - \bar{x}_i) &\geq 0, \end{aligned} \tag{2.40}$$

then  $\bar{x}$  is a global minimizer of (CP2), where  $\hat{r}_i = \min\{0, r_i\}$ .

*Proof.* For (CP2), choose  $Q = A = \text{diag}(r_1, \dots, r_n)$ . If (2.40) holds, then, by Theorem 2.3,  $\bar{x}$  is a global minimizer of (CP2).  $\square$

### 3. Sufficient Conditions of Bivalent Programming

In this section, we will consider the following bivalent programming:

$$\begin{aligned} \min \quad f(x) &= b^T x^3 + \frac{1}{2}x^T Ax + a^T x \\ \text{s.t.} \quad x \in D_B &= \prod_{i=1}^n \{u_i, v_i\}, \end{aligned} \tag{CP3}$$

where  $A$ ,  $a$ ,  $b$ , and  $u_i, v_i, i = 1, \dots, n$  are the same as in (CP1).

Similar to Theorem 2.3, we will obtain the global sufficient optimality conditions for (CP3).

**Theorem 3.1.** For (CP3), let  $\bar{x} = (\bar{x}_1, \dots, \bar{x}_n)^T \in D_B$ . Suppose that there exists a diagonal matrix  $Q = \text{diag}(\alpha_1, \dots, \alpha_n)$ ,  $\alpha_i \in R, i = 1, \dots, n$  such that  $A - Q \succeq 0$ , and for all  $x \in D_B$ ,  $b_i(x_i - \bar{x}_i) \geq 0$  ( $i = 1, \dots, n$ ). If

$$\tilde{X}(A\bar{x} + a) - \frac{1}{2}Q(v - u) \leq \mathbf{0}, \tag{3.1}$$

then  $\bar{x}$  is a global minimizer of problem (CP3).

*Proof.* Suppose that condition (3.1) holds. Let

$$\begin{aligned} l(x) &= b^T x^3 + \frac{1}{2}x^T Qx + \beta^T x, \\ \beta &= (A - Q)\bar{x} + a. \end{aligned} \tag{3.2}$$

Then

$$f(x) - f(\bar{x}) \geq l(x) - l(\bar{x}), \quad \forall x \in R^n. \tag{3.3}$$

Obviously if  $l(x) - l(\bar{x}) \geq 0$  for each  $x \in D_B$ , then  $\bar{x}$  is a global minimizer of (CP3).

Note that

$$l(x) - l(\bar{x}) = \sum_{i=1}^n \frac{\alpha_i}{2} (x_i - \bar{x}_i)^2 + (A\bar{x} + a)^T (x - \bar{x}) + b^T (x^3 - \bar{x}^3). \quad (3.4)$$

Thus,  $\bar{x}$  is a global minimizer of  $l(x)$  with binary constraints if and only if, for each  $i = 1, \dots, n$ ,  $x_i \in \{u_i, v_i\}$ ,

$$\frac{\alpha_i}{2} (x_i - \bar{x}_i)^2 + (A\bar{x} + a)_i (x_i - \bar{x}_i) + b_i (x_i^3 - \bar{x}_i^3) \geq 0. \quad (3.5)$$

Firstly, we note from (3.1), for each  $i = 1, \dots, n$ , that

$$-\frac{\alpha_i}{2} (v_i - u_i) + \tilde{x}_i (A\bar{x} + a)_i \leq 0. \quad (3.6)$$

Next we only show it from the following two cases.

*Case 1.* (If  $\bar{x}_i = u_i$ ), then (3.6) is equivalent to

$$\frac{\alpha_i}{2} (v_i - u_i) + (A\bar{x} + a)_i \geq 0. \quad (3.7)$$

It is obvious that, for each  $x_i \in \{u_i, v_i\}$ ,

$$\frac{\alpha_i}{2} (x_i - u_i)^2 + b_i (x_i^3 - u_i^3) + (A\bar{x} + a)_i (x_i - u_i) \geq 0. \quad (3.8)$$

So (3.5) holds.

*Case 2.* (If  $\bar{x}_i = v_i$ ), then (3.6) is equivalent to

$$-\frac{\alpha_i}{2} (v_i - u_i) + (A\bar{x} + a)_i \leq 0. \quad (3.9)$$

It is obvious that, for each  $x_i \in \{u_i, v_i\}$ ,

$$\frac{\alpha_i}{2} (v_i - x_i)^2 - b_i (v_i^3 - x_i^3) - (A\bar{x} + a)_i (v_i - x_i) \geq 0. \quad (3.10)$$

So (3.5) holds.

From (3.5), we can conclude that  $\bar{x}$  is a global minimizer of problem (CP3)  $\square$

Similar to Corollary 2.4, we have the following corollary.

**Corollary 3.2.** *For (CP3), let  $\bar{x} \in D_B$ . Suppose that, for all  $x \in D_B$ ,  $b_i(x_i - \bar{x}_i) \geq 0$  ( $i = 1, \dots, n$ ).  
(1) When  $A$  is a positive semidefinite matrix, if*

$$\tilde{X}(A\bar{x} + a) \leq \mathbf{0}, \quad (3.11)$$

*then  $\bar{x}$  is a global minimizer of (CP3).*

(2) Let

$$\alpha_i = \frac{2\tilde{x}_i(A\bar{x} + a)_i}{v_i - u_i}, \quad i = 1, \dots, n, \quad (3.12)$$

$$Q = \text{diag}(\alpha_1, \dots, \alpha_n).$$

When  $A$  is not a positive semidefinite matrix, if  $A - Q \succeq 0$  holds, then  $\bar{x}$  is a global minimizer of (CP3). Otherwise, one can conclude that there is no such diagonal matrix  $Q$  that meets the requirements of Theorem 3.1.

We just show the proof of (2).

*Proof.* We construct the diagonal matrix  $Q = \text{diag}(\alpha_1, \dots, \alpha_n)$ , where  $\alpha_i = 2\tilde{x}_i(A\bar{x} + a)_i / (v_i - u_i)$ ,  $i = 1, \dots, n$ . If  $A - Q \succeq 0$ , we just need to test condition (3.1).

Because

$$\alpha_i = \frac{2\tilde{x}_i(A\bar{x} + a)_i}{v_i - u_i}, \quad i = 1, \dots, n, \quad (3.13)$$

then rewriting the above equations, we have

$$\tilde{x}_i(A\bar{x} + a)_i - \frac{1}{2}\alpha_i(v_i - u_i) = 0, \quad i = 1, \dots, n. \quad (3.14)$$

It obviously means that the diagonal matrix  $Q$  also satisfies condition (3.1). According to Theorem 3.1,  $\bar{x}$  is a global minimizer of (CP3).  $\square$

Note that there is difference between formula (3.1) and formula (2.11). In formula (3.1), the diagonal elements  $\alpha_i$  of a diagonal matrix  $Q$  are allowed to be positive or nonpositive. But in formula (2.11), the diagonal elements  $\hat{\alpha}_i$  of a diagonal matrix  $\hat{Q}$  must meet the conditions  $\hat{\alpha}_i \leq 0$ . So we have to discuss the sign of the terms  $\tilde{x}_i(A\bar{x} + a)_i$  ( $i = 1, \dots, n$ ) in Corollary 3.2.

We now consider a special case of (CP3):

$$\begin{aligned} \min \quad & f(x) = \sum_{i=1}^n b_i x_i^3 + \sum_{i=1}^n \frac{r_i}{2} x_i^2 + \sum_{i=1}^n a_i x_i \\ \text{s.t.} \quad & x \in D_B = \prod_{i=1}^n \{u_i, v_i\}, \end{aligned} \quad (CP4)$$

where  $b_i, r_i, a_i, u_i, v_i \in R$  and  $u_i \leq v_i$ ,  $i = 1, \dots, n$ .

**Corollary 3.3.** For (CP4), let  $\bar{x} \in D_B$ . If, for each  $i = 1, \dots, n$ ,

$$\begin{aligned} \tilde{x}_i(r_i \bar{x}_i + a_i) - \frac{r_i}{2}(v_i - u_i) &\leq 0, \\ b_i(x_i - \bar{x}_i) &\geq 0, \end{aligned} \quad (3.15)$$

then  $\bar{x}$  is a global minimizer of (CP4).

*Proof.* For (CP4), choose  $Q = A = \text{diag}(r_1, \dots, r_n)$ . If conditions (3.15) hold, then, by Theorem 3.1,  $\bar{x}$  is a global minimizer of (CP4).  $\square$

## 4. Numerical Examples

In this section, six examples are given to test the proposed global sufficient optimality condition.

*Example 4.1.* Consider the following problem:

$$\begin{aligned} \min \quad & \frac{7}{3}x_1^3 + 5x_2^3 + 2x_3^3 + \frac{3}{2}x_1^2 + x_2^2 + \frac{1}{2}x_3^2 + 2x_1x_2 + x_1x_3 + x_2x_3 + \frac{3}{2}x_1 + 5x_2 + 3x_3 \\ \text{s.t.} \quad & x \in D = \prod_{i=1}^3 [1, 2]. \end{aligned} \quad (4.1)$$

Let

$$A = \begin{bmatrix} 3 & 2 & 1 \\ 2 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (4.2)$$

and  $a = (3/2, 5, 3)^T$ ,  $b = (7/3, 5, 2)^T$ . Obviously  $A$  is a positive semidefinite matrix.

Considering  $\bar{x} = (1, 1, 1)^T$ , obviously we have, for each  $x_i \in [1, 2]$ ,  $b_i(x_i - \bar{x}_i) \geq 0$  ( $i = 1, 2, 3$ ). Note that  $A\bar{x} + a = (15/2, 10, 6)^T$  and  $\tilde{X} = \text{diag}(-1, -1, -1)$ , and so

$$\tilde{X}(A\bar{x} + a) = \begin{bmatrix} -15 \\ 2 \\ -10 \\ -6 \end{bmatrix} < \mathbf{0}. \quad (4.3)$$

According to Corollary 2.4(1),  $\bar{x} = (1, 1, 1)$  is a global minimizer.

*Example 4.2.* Consider the following problem:

$$\begin{aligned} \min \quad & -x_1^3 - 3x_2^3 + \frac{1}{2}x_4^3 - x_1^2 - x_2^2 + \frac{3}{2}x_3^2 - \frac{1}{2}x_4^2 - 2x_1x_2 + x_1x_4 + 2x_2x_4 - x_1 - 4x_2 + 5x_4 \\ \text{s.t.} \quad & x \in D = \prod_{i=1}^4 [-1, 1]. \end{aligned} \quad (4.4)$$

Let

$$A = \begin{bmatrix} -2 & -2 & 0 & 1 \\ -2 & -2 & 0 & 2 \\ 0 & 0 & 3 & 0 \\ 1 & 2 & 0 & -1 \end{bmatrix} \quad (4.5)$$

and  $a = (-1, -4, 0, 5)^T$ ,  $b = (-1, -3, 0, 1/2)^T$ . Obviously  $A$  not is a positive semidefinite matrix.

Considering  $\bar{x} = (1, 1, 0, -1)^T$ , obviously we have, for each  $x_i \in [-1, 1]$ ,  $b_i(x_i - \bar{x}_i) \geq 0$  ( $i = 1, 2, 3, 4$ ).

Note that  $A\bar{x} + a = (-6, -10, 0, 9)^T$ . Then letting  $\tilde{X} = \text{diag}(1, 1, 0, -1)$ , we have  $\tilde{X}(A\bar{x} + a) = (-6, -10, 0, -9)^T \leq \mathbf{0}$ . Let  $\alpha_1 = 2\tilde{x}_1(A\bar{x} + a)_1/(v_1 - u_1) = -6$ ,  $\alpha_2 = -10$ ,  $\alpha_3 = 0$  and  $\alpha_4 = -9$ . Then  $Q = \text{diag}(-6, -10, 0, -9)$ , which satisfies  $A - Q > 0$ . According to Corollary 2.4(3),  $\bar{x} = (1, 1, 0, -1)$  is a global minimizer.

*Example 4.3.* Consider the following problem:

$$\begin{aligned} \min \quad & -x_1^3 + \frac{2}{3}x_2^3 - 3x_3^3 - 2x_1^2 - \frac{3}{5}x_2^2 - 4x_3^2 + \frac{1}{2}x_4^2 - 2x_1 + 5x_2 - 3x_3 \\ \text{s.t.} \quad & x \in D = \prod_{i=1}^4 [-1, 1]. \end{aligned} \quad (4.6)$$

Let  $A = \text{diag}(-4, -6/5, -8, 1)$  and  $a = (-2, 5, -3, 0)^T$ ,  $b = (-1, 2/3, -3, 0)^T$ ,  $r = (-4, -6/5, -8, 1)^T$ .

Consider  $\bar{x} = (1, -1, 1, 0)^T$ .

Let  $\tilde{X} = \text{diag}(1, -1, 1, 0)$ ,  $\hat{r} = (-4, -6/5, -8, 0)^T$ . Then

$$\begin{aligned} \tilde{x}_1(r_1\bar{x}_1 + a_1) - \frac{1}{2}\hat{r}_1(v_1 - u_1) &= -2 < 0, \\ \tilde{x}_2(r_2\bar{x}_2 + a_2) - \frac{1}{2}\hat{r}_2(v_2 - u_2) &= -5 < 0, \\ \tilde{x}_3(r_3\bar{x}_3 + a_3) - \frac{1}{2}\hat{r}_3(v_3 - u_3) &= -3 < 0, \\ \tilde{x}_4(r_4\bar{x}_4 + a_4) - \frac{1}{2}\hat{r}_4(v_4 - u_4) &= 0, \\ x_i \in [-1, 1], \quad b_i(x_i - \bar{x}_i) &\geq 0, \quad (i = 1, 2, 3, 4). \end{aligned} \quad (4.7)$$

According to Corollary 2.5,  $\bar{x} = (1, -1, 1, 0)$  is a global minimizer.

*Example 4.4.* Consider the following problem:

$$\begin{aligned} \min \quad & 4x_1^3 + \frac{5}{2}x_2^3 - 3x_3^3 - \frac{8}{3}x_4^3 - \frac{3}{2}x_1^2 - \frac{3}{2}x_2^2 + 3x_3^2 - x_4^2 + 2x_1x_2 + x_1x_4 + x_2x_3 - x_3x_4 \\ & + 5x_1 + \frac{9}{2}x_2 - 2x_3 - 2x_4 \\ \text{s.t.} \quad & x \in D_B = \prod_{i=1}^4 \{-1, 1\}. \end{aligned} \quad (4.8)$$

Let

$$A = \begin{bmatrix} -3 & 2 & 0 & 1 \\ 2 & -3 & 1 & 0 \\ 0 & 1 & 6 & -1 \\ 1 & 0 & -1 & -2 \end{bmatrix} \quad (4.9)$$

and  $a = (5, 9/2, -2, -2)^T$ ,  $b = (4, 5/2, -3, -8/3)^T$ . Obviously  $A$  not is a positive semidefinite matrix.

Considering  $\bar{x} = (-1, -1, 1, 1)^T$ , it follows that, for each  $x \in D_B$ ,  $b_i(x_i - \bar{x}_i) \geq 0$  ( $i = 1, 2, 3, 4$ ). Note that  $A\bar{x} + a = (7, 13/2, 2, -6)^T$  and  $\tilde{X} = \text{diag}(-1, -1, 1, 1)$ . Let  $\alpha_1 = (2\tilde{x}_1(A\bar{x} + a)_1)/(v_1 - u_1) = -7$ . Similarly we have,  $\alpha_2 = -13/2$ ,  $\alpha_3 = 2$  and  $\alpha_4 = -6$ . Then  $Q = \text{diag}(-7, -13/2, 2, -6)$ , which satisfies  $A - Q > 0$ . According to Theorem 3.1(2),  $\bar{x} = (-1, -1, 1, 1)$  is a global minimizer.

*Example 4.5.* Consider the following problem:

$$\begin{aligned} \min \quad & 3x_1^3 - 8x_2^3 + 5x_3^3 + \frac{1}{2}x_4^3 - x_1^2 + 4x_2^2 - \frac{5}{3}x_3^2 - 3x_4^2 + x_1 - 2x_2 + 3x_3 + 4x_4 \\ \text{s.t.} \quad & x \in D_B = \prod_{i=1}^4 \{-1, 0\}. \end{aligned} \quad (4.10)$$

Let  $a = (1, -2, 3, 4)^T$ ,  $b = (3, -8, 5, 1/2)^T$ , and  $r = (-2, 8, -10/3, -6)^T$ .

Consider  $\bar{x} = (-1, 0, -1, -1)^T$ .

Let  $\tilde{X} = \text{diag}(-1, 1, -1, -1)$ . Then

$$\begin{aligned} \tilde{x}_1(r_1\bar{x}_1 + a_1) - \frac{1}{2}r_1(v_1 - u_1) &= -2 < 0, \\ \tilde{x}_2(r_2\bar{x}_2 + a_2) - \frac{1}{2}r_2(v_2 - u_2) &= -6 < 0, \\ \tilde{x}_3(r_3\bar{x}_3 + a_3) - \frac{1}{2}r_3(v_3 - u_3) &= -\frac{14}{3} < 0, \end{aligned}$$

$$\begin{aligned}
& \tilde{x}_4(r_4\bar{x}_4 + a_4) - \frac{1}{2}r_4(v_4 - u_4) = -7 < 0, \\
& x \in D_B, \quad b_i(x_i - \bar{x}_i) \geq 0, \quad (i = 1, 2, 3, 4).
\end{aligned} \tag{4.11}$$

According to Corollary 3.3,  $\bar{x} = (-1, 0, -1, -1)$  is a global minimizer.

*Example 4.6.* Consider the following problem:

$$\begin{aligned}
\min \quad & 3x_1^3 - 8x_2^3 + 5x_3^3 + \frac{1}{2}x_4^3 - x_1^2 - 4x_2^2 + \frac{5}{3}x_3^2 - 3x_4^2 - x_1 + 2x_2 + 3x_3 + 4x_4 \\
\text{s.t.} \quad & x \in D_B = \prod_{i=1}^4 \{-1, 0\}.
\end{aligned} \tag{4.12}$$

Let  $a = (-1, 2, 3, 4)^T$ ,  $b = (3, -8, 5, 1/2)^T$  and  $r = (-2, -8, 10/3, -6)^T$ .

Consider  $\bar{x} = (-1, 0, -1, -1)^T$ .

Let  $\tilde{X} = \text{diag}(-1, 1, -1, -1)$ . Then

$$\begin{aligned}
& \tilde{x}_1(r_1\bar{x}_1 + a_1) - \frac{1}{2}r_1(v_1 - u_1) = 0, \\
& \tilde{x}_2(r_2\bar{x}_2 + a_2) - \frac{1}{2}r_2(v_2 - u_2) = 6 > 0, \\
& \tilde{x}_3(r_3\bar{x}_3 + a_3) - \frac{1}{2}r_3(v_3 - u_3) = -\frac{4}{3} < 0, \\
& \tilde{x}_4(r_4\bar{x}_4 + a_4) - \frac{1}{2}r_4(v_4 - u_4) = -7 < 0.
\end{aligned} \tag{4.13}$$

We can see that the conditions are not true in  $\bar{x} = (-1, 0, -1, -1)^T$  in Corollary 3.3. But  $\bar{x} = (-1, 0, -1, -1)$  is a global minimizer. This fact exactly shows that the conditions are just sufficient.

## Acknowledgment

This research was supported by Innovation Program of Shanghai Municipal Education Commission (12ZZ071), Shanghai Pujiang Program (11PJC059), and the National Natural Science Foundation of China (71071113).

## References

- [1] R. A. Canfield, "Multipoint cubic surrogate function for sequential approximate optimization," *Structural and Multidisciplinary Optimization*, vol. 27, pp. 326–336, 2004.
- [2] Y. Nesterov, "Accelerating the cubic regularization of Newton's method on convex problems," *Mathematical Programming*, vol. 112, no. 1, pp. 159–181, 2008.

- [3] C. S. Lin, P.-R. Chang, and J. Y. S. Luh, "Formulation and optimization of cubic polynomial joint trajectories for industrial robots," *IEEE Transactions on Automatic Control*, vol. 28, no. 12, pp. 1066–1074, 1983.
- [4] A. Beck and M. Teboulle, "Global optimality conditions for quadratic optimization problems with binary constraints," *SIAM Journal on Optimization*, vol. 11, no. 1, pp. 179–188, 2000.
- [5] J.-B. Hiriart-Urruty, "Global optimality conditions in maximizing a convex quadratic function under convex quadratic constraints," *Journal of Global Optimization*, vol. 21, no. 4, pp. 445–455, 2001.
- [6] J.-B. Hiriart-Urruty, "Conditions for global optimality 2," *Journal of Global Optimization*, vol. 13, no. 4, pp. 349–367, 1998.
- [7] A. Strekalovsky, "Global optimality conditions for nonconvex optimization," *Journal of Global Optimization*, vol. 12, no. 4, pp. 415–434, 1998.
- [8] V. Jeyakumar, A. M. Rubinov, and Z. Y. Wu, "Sufficient global optimality conditions for non-convex quadratic minimization problems with box constraints," *Journal of Global Optimization*, vol. 36, no. 3, pp. 471–481, 2006.
- [9] Z. Y. Wu, V. Jeyakumar, and A. M. Rubinov, "Sufficient conditions for global optimality of bivalent nonconvex quadratic programs with inequality constraints," *Journal of Optimization Theory and Applications*, vol. 133, no. 1, pp. 123–130, 2007.
- [10] V. Jeyakumar, A. M. Rubinov, and Z. Y. Wu, "Non-convex quadratic minimization problems with quadratic constraints: global optimality conditions," *Mathematical Programming A*, vol. 110, no. 3, pp. 521–541, 2007.
- [11] Z. Y. Wu, "Sufficient global optimality conditions for weakly convex minimization problems," *Journal of Global Optimization*, vol. 39, no. 3, pp. 427–440, 2007.
- [12] Y. Wang and Z. Liang, "Global optimality conditions for cubic minimization problem with box or binary constraints," *Journal of Global Optimization*, vol. 47, no. 4, pp. 583–595, 2010.
- [13] D. Pallaschke and S. Rolewicz, *Foundations of Mathematical Optimization, Convex Analysis without Linearity*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1997.

*Research Article*

## **A Two-Stage DEA to Analyze the Effect of Entrance Deregulation on Iranian Insurers: A Robust Approach**

**Seyed Gholamreza Jalali Naini and Hamid Reza Nouralizadeh**

*Department of Industrial Engineering, Iran University of Science and Technology, Narmak, Tehran 1684613114, Iran*

Correspondence should be addressed to Hamid Reza Nouralizadeh, nouralizadeh@iust.ac.ir

Received 17 February 2012; Revised 5 May 2012; Accepted 29 May 2012

Academic Editor: Yi-Chung Hu

Copyright © 2012 Seyed Gholamreza Jalali Naini et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We use two-stage data envelopment analysis (DEA) model to analyze the effects of *entrance deregulation* on the efficiency in the Iranian insurance market. In the first stage, we propose a *robust optimization* approach in order to overcome the sensitivity of DEA results to any uncertainty in the output parameters. Hence, the efficiency of each ongoing insurer is estimated using our proposed robust DEA model. The insurers are then ranked based on their relative efficiency scores for an eight-year period from 2003 to 2010. In the second stage, a comprehensive statistical analysis using *generalized estimating equations* (GEE) is conducted to analyze some other factors which could possibly affect the efficiency scores. The first results from DEA model indicate a decline in efficiency over the entrance deregulation period while further statistical analysis confirms that the solvency ignorance which is a widespread paradigm among state owned companies is one of the main drivers of efficiency in the Iranian insurance market.

### **1. Introduction**

In line with the Iran's Third Development Plan in 1999, the Iranian parliament approved the establishment of private insurance companies in August 2002 with the aim of improving efficiency, increasing consumer choices through increased rivalry, and finally enhancing transparency in the market [1]. There are some differences in the literature regarding to the deregulation impact analysis on the efficiency of insurance companies. While Rees et al. [2] reported small improvement in German and Britain life insurance market after deregulation, Hussels and Ward also could not find any strong evidence of deregulation effects on insurance business efficiency between 1992 until 2002 [3], Wang and her colleague showed the whole market getting more competitive due to leaders market-share losing after

entrance deregulation in life insurance [4]. Jeng and Lai's study showed that the deregulation and liberalization, reduction of government or other barriers to market, do not have major adverse impact on the technical, cost, and revenue efficiency of existing firms in the long run. The dominance of existing firms has declined but persisted throughout the sample period [5]. Cummins and Rubio-Misas found a positive effect of deregulation on the Spanish insurance companies' efficiency [6]. Boonyasai et al. also found similar evidence in some Southeast Asian countries stemmed from joint deregulation and liberalization policies [7]. Based on the previous studies, one could observe that in some cases the developed insurance markets have improved slightly after deregulation, while there is the possibility of having no effect or affecting negatively.

In this paper, we test the effects of deregulation on Iranian insurance market. After Iran's revolution in 1979, all private insurers were merged compulsorily into four state-owned companies. Following the entrance deregulation, the Iranian insurance market has witnessed considerable increase in the number of companies since 2003. A criticism of the entrance deregulation process was however the lack of sufficient supervision in the insurance market. To find more evidence whether the Iranian insurance companies have been able to improve their efficiency as a result of the deregulation process, this paper analyzes the efficiency of the Iranian insurance market using a two-stage robust DEA model. We use data from 2003 to 2010, which include the accomplishment of the entrance deregulation. To our best knowledge, this paper is the first research conducted to examine the relative efficiency of all Iranian insurance companies for the period of entrance deregulation.

The academic novelty of this research is presenting comprehensive literature review and applying a two-stage robust DEA model to the problem. We show that traditional DEA has a limitation to cover all aspects of insurers' behavior, and a robust DEA model can be used to overcome the limitations of previous methods. We also use GEE model to extract the most significant factors explaining robust CRS efficiency scores. According to our study, applying GEE model as second stage in analyzing efficiency scores is another novelty of this research. To the best of our knowledge, some efforts have been conducted to analyze the effects of insurance deregulation on the efficiency of the industry throughout the world using DEA, however, few researches, if any, considered this issue within Iranian insurance market.

The rest of the paper is arranged as follows: in Section 1, we review different characteristics of Iranian Insurance market. Section 2 reviews the existing literature on the topic, especially those related to mathematical programming approach in efficiency measurement (DEA), *robust optimization*, and *generalized estimating equations* (GEE). Section 3 discusses applications of DEA in institutional changes of insurance industry. In order to describe different approaches in output selection for DEA model, Section 4 investigates the *Value Added* versus the *Financial Intermediary* approaches. Section 5 presents the results of efficiency analysis of Iranian insurance companies from 2003 until 2010. This section presents the first stage and compares *traditional CRS* DEA scores with *robust CRS* DEA scores and tries to find factors explaining efficiencies. In Section 6, a GEE model is applied to capture the most important factors that explain efficiencies. Finally, Section 7 illustrates the conclusions.

## 2. Overview of the Iranian Insurance Market

The Iranian insurance market is one of the less developed in the world. Its position also is not different substantially from its neighbors. Over the past 10 years, as shown in the Table 1, Iranian insurance market is in the last rank based on Premium per Capita; while,

its rank based on Penetration Ratio is a little better among its neighbors, as shown in Table 2. However, If the index is compared with the global average, 6.89% in 2010 [20], then the level of the ratio would not be satisfactory. It seems that the insurance sector has not been able to play its primary role in the economy when compared with its counterparts in developed countries. During last 10 years, Iranian insurance industry has witnessed two major changes in the institutions. The first one was *entrance deregulation* which abolished the monopoly of state-owned companies in 2001. Based on the new law, the private insurance companies have been established and number of insurance companies increased to 23 from 4 since 2001. The effects of this institutions change can be seen in Tables 1 and 2, where the premium per capita and penetration ratio is increasing at very high clip from US\$ 11.1 to US\$ 34.4 and from 0.86% to 1.28%, respectively, within three years. The second institutions change was price deregulation in 2009. Based on the new law, the insurances companies are allowed to set their own premium rates in property and casualty lines of business (P&L). It is also obvious that following *price deregulation* the growth of the insurance industry has been revitalized after a period of calm. By the way, due to the lack of available data, the main purpose of this paper focuses on the first event.

The last issue is that the share of life insurance in Iran is very low in comparison with its global average. Based on the yearbook of Central Insurance of IR Iran, the average of life insurance share in total written premium of Iran's market is around 7.5% in the last 10 years. The analysis of this issue is beyond the scope of this paper, but this phenomenon has caused all companies established in Iran to be general (mixed) insurers which mostly do business in P&L lines. This in turn, helps us to deal with all companies in a similar way when assessing the efficiency.

### 3. Theoretical Framework

#### 3.1. The Economic Efficiency

Frontier methodologies have been used in majority of papers published in recent years. There are two common tools in frontier methodologies: the *econometric frontier analysis* and the *data envelopment analysis* (DEA). Both have their own pros and cons. Unlike the data envelopment analysis approach, it's not allowed in econometric stochastic frontier approach to use various inputs and outputs and it also requires researchers to define functional form on the data and set assumptions about distributional form of the inefficiency term. Both of them presume the production function is known. The economic efficiency is stemmed from production frontier in *theory of the firm*. Figure 1 shows a production frontier (PF) for a firm with single input/output.

If a firm is producing at point  $(I, J)$  in time  $t$ , it could produce more efficiently by moving to the frontier  $PF^t$  horizontally or vertically. If it moves to  $PF^t$  horizontally by reducing its excess input, this is called input-oriented and, it is called output-oriented if the point moves vertically by producing more output. The technical efficiency of the firm is calculated by the ratio  $OH/OI$  for input-oriented approach, which is the reciprocal of its distance from the frontier  $PF^t$ . Defining efficiency by the concept of the distance from the production frontier is formulated by Shephard [21] as below; assume a producer uses input

**Table 1:** Premium per Capita (PPC) trend of selected countries 2003–2010 (US Dollar).

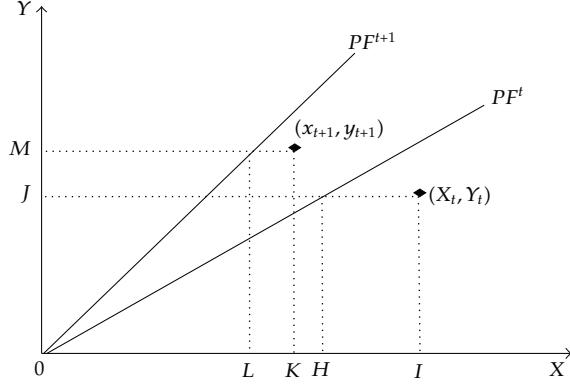
S.C.	Year										
	2001 P.P.C.	2002 P.P.C.	2003 A.G.	2004 P.P.C.	2005 A.G.	2006 P.P.C.	2007 A.G.	2008 P.P.C.	2009 A.G.	2010 P.P.C.	
UAE	302.4	317	5%	310.7	-2%	350.2	13%	414.2	18%	585.1	41%
Qatar	NA	NA	NA	NA	NA	442.3	0%	683.5	55%	640.2	-6%
Bahrain	219.7	295.2	34%	NA	NA	NA	NA	NA	NA	NA	548.6
Kuwait	131.5	154.1	17%	148	-4%	161.2	9%	185.5	15%	227.2	22%
Oman	77.4	84	9%	99	18%	103.1	4%	113.7	10%	143.7	26%
Saudi Arabia	47.2	41.6	-12%	41.2	-1%	51.4	25%	57.1	11%	63.1	11%
Iran	11.1	17.3	56%	22.9	32%	29.1	27%	34.4	18%	41.1	19%
								50.9	24%	59.1	16%
								63.9	8%	76.8	20%

**Table 2:** Penetration ratio (PR) trend of selected countries 2003–2010 (%).

S.C.	Year									
	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010
	P.R.	P.R.	A.G.	P.R.	A.G.	P.R.	A.G.	P.R.	A.G.	P.R.
UAE	1.25	1.28	2%	1.12	-13%	1.65	47%	1.53	-7%	1.7
Qatar	NA	1.09	-11%	1.1						
Bahrain	1.93	2.08	8%	NA						
Kuwait	0.79	0.95	20%	0.92	-3%	0.93	1%	0.79	-15%	0.7
Oman	0.96	1.01	5%	1.44	43%	1.28	-11%	1.14	-11%	1
Saudi Arabia	0.53	0.48	-9%	0.47	-2%	0.48	2%	0.46	-4%	0.5
Iran	0.86	1	16%	1.16	16%	1.25	8%	1.28	2%	1.3

S.C.: selected countries; P.P.C: premium per capita; A.G.: annual growth; P.R.: penetration ratio.

Source: [8].



**Figure 1:** Production Frontier (PF) for a firm with one input and one output.

$x = (x_1, x_2, \dots, x_k)$  to obtain output vector  $y = (y_1, y_2, \dots, y_n)$ , then Shephard's distance by definition is equal to

$$D(x, y) = \sup \left\{ \theta : \left( \frac{x}{\theta}, y \right) \in V(y) \right\} = (\inf \{ \theta : (\theta x, y) \in V(y) \})^{-1}, \quad (3.1)$$

where  $V$ ,  $D$ , and  $\theta$  are production frontier, distance function, and distance, respectively. The technical efficiency  $TE(x, y)$  is therefore defined as  $TE(x, y) = 1/D(x, y)$ . It should be mentioned here that  $TE(x, y)$  for each decision making unit can be obtained by linear programming [22]. If a firm uses two or more inputs, inefficiency can be also stemmed from the fact that it's not deploying the cost minimizing combination of inputs. This kind of inefficiency is called allocative inefficiency. A firm is considered to be fully cost efficient if it operates at a point where both technical and allocative efficiency are met. Cost efficiency is then defined as follows:

$$\text{Cost efficiency} = \text{Technical Efficiency} \times \text{Allocative Efficiency}. \quad (3.2)$$

We can also use production frontier to capture productivity improvement. Productivity means technology improvement between periods, and it differs from efficiency. Figure 1 shows production frontiers for periods  $t$  and  $t + 1$  ( $PF^t$  and  $PF^{t+1}$ , resp.) for the one input-one output firm. The frontier for period  $t$  is on the right of the frontier for period  $t + 1$ . This reveals that productivity has improved between period  $t$  and  $t + 1$ . Suppose a firm operating at point  $(x^t, y^t)$  in period  $t$  and at point  $(x^{t+1}, y^{t+1})$  at period  $t + 1$ . As it is obvious in Figure 1, both productivity and efficiency of the firm have improved between two periods. The firm's operation at period  $t + 1$  is impossible in period  $t$  indicating that productivity has been improved, and as the  $x^{t+1}$  is closer to its frontier than  $x^t$ , means that its efficiency has been also improved between  $t$  and  $t + 1$ . Based on these two kinds of distance, the distance can also be defined as the Malmquist Index to capture total factor productivity. If we want to determine whether productivity change has occurred between period  $t$  and  $t + 1$ , we can

choose  $\text{PF}^t$  or  $\text{PF}^{t+1}$  as the frontier reference. With respect to the period  $t$  frontier, an input-oriented Malmquist productivity index can be defined as:

$$M^t = \frac{D^t(x^t, y^t)}{D^t(x^{t+1}, y^{t+1})}. \quad (3.3)$$

Similarly, the Malmquist index based on period  $t + 1$  frontier can be defined. To avoid arbitrarily selecting one frontier to compute the index, the geometric mean could be applied as follows:

$$M(x^{t+1}, y^{t+1}, x^t, y^t) = \sqrt{\frac{D^t(x^t, y^t)}{D^t(x^{t+1}, y^{t+1})} \times \frac{D^{t+1}(x^t, y^t)}{D^{t+1}(x^{t+1}, y^{t+1})}}. \quad (3.4)$$

With some mathematical calculations, these expressions can be decomposed into technical change and efficiency. If production frontier changes by scale, one can separate efficiency into pure technical and scale efficiency. Pure technical efficiency is defined as the distance from the variable returns to scale (VRS) frontier, and the relationship  $\text{TE}(x, y) = \text{PT}(x, y) \times S(x, y)$  is applicable to separate pure technical and scale efficiency, in which  $S(x, y)$  represents scale efficiency and  $\text{PT}(x, y)$  pure technical efficiency. Two types of production frontiers can be considered for the single input-single output case; Constant returns to scale (CRS) frontier and variable returns to scale (VRS) frontier. It is socially and economically optimal for firms to operate at constant returns to scale, providing the motivation for separating pure technical and scale efficiency [23].

It is worth noting that assuming efficient frontier to be a convex set is criticized by some scholars. Deprins et al. [24] criticize the DEA methodology for imposing the convexity assumption. They proposed the elimination of the convexity assumption which leads to the free disposal hull (FDH) estimation technique. The FDH implies free disposability supposing that outputs do not change if some inputs increase. It has been shown that FDH increases goodness of fit [25]; however, the convex frontier may be required in some industries [23].

### 3.2. Mathematical Programming (DEA)

DEA is a mathematical programming introduced by Charnes et al. in 1978 [26]. There are two approaches in DEA modeling; the input oriented and the output oriented. Input oriented models have an objective in such a form that maximizes weighted outputs given the level of inputs. On the other hand, the objective function in output oriented models takes the form of minimizing weighted inputs given the level of outputs [27]. The basic fractional Constant Returns to Scale (CRS) DEA estimates the relative efficiencies of  $n$  DMUs which are described in model (3.5). Each DMU is shown with  $m$  for input and  $s$  for outputs denoted by  $b_{1j}, \dots, b_{mj}$

and  $a_{1j}, \dots, a_{sj}$ , respectively, where the ratio of the weighted sum of outputs to the weighted sum of inputs for some given  $DMU_o$  is maximized as follows:

$$\begin{aligned} \max \quad & \delta_0 = \frac{\sum_{r=1}^s x_r a_{r0}}{\sum_{i=1}^m v_i b_{i0}} \\ \text{subject to} \quad & \frac{\sum_{r=1}^s x_r a_{rj}}{\sum_{i=1}^m v_i b_{ij}} \leq 1, \quad \forall j = 1, \dots, n, \\ & x_r, v_i \geq 0, \end{aligned} \quad (3.5)$$

where  $x_r$  and  $v_i$  are the weight factors, and the  $\delta_0$ ,  $a_{r0}$ , and  $b_{i0}$  are the observed efficiency, output and input values, respectively, of  $DMU_o$ , the DMU to be evaluated. Model (3.5) is a nonlinear fractional programming model which could be converted into the following LP model [26]:

$$\begin{aligned} \max \quad & \delta_0 \\ \text{subject to} \quad & \sum_{r=1}^s x_r a_{rj} - \sum_{i=1}^m v_i b_{ij} \leq 0, \quad \forall j = 1, \dots, n, \\ & u_r, v_i \geq 0. \end{aligned} \quad (3.6)$$

In order to have an objective function without any uncertain parameter, we further reformulate model (3.7) using some auxiliary variable  $w$ , as follows:

$$\begin{aligned} \max \quad & W \\ \text{subject to} \quad & w - \sum_{r=1}^s x_r \tilde{\alpha}_{rj} \leq 0 \\ & \sum_{r=1}^s x_r \tilde{\alpha}_{rj} - \sum_{i=1}^m v_i b_{ij} \leq 0, \quad \forall j = 1, \dots, n \\ & \sum_{i=1}^m v_i b_{ij} = 1 \\ & x_r, v_i \geq 0, \end{aligned} \quad (3.7)$$

where,  $x$  and  $v$  are input and output variables and Indices  $i$ ,  $r$ , and  $j$  represent the number of inputs, outputs and DMUs respectively. One could observe that in this formulation, the parameter  $\tilde{\alpha}_{rj}$  is an uncertain value. We cannot solve this problem with popular Linear Programming techniques, since the primary assumption, that is, certainty of input parameters is violated.

### 3.3. Robust Optimization

The implicit assumption behind the traditional DEA is that input and output are deterministic. But output of an insurer is not necessarily deterministic. For example, if we consider paid loss as output of an insurer, it may be changed due to incorrect estimate of *outstanding loss reserve* (Outstanding loss reserve refers to the loss that incurred and reported but not paid yet.) or *IBNR reserve* (*IBNR reserve* refers to the loss that incurred but not reported yet.). These kinds of output can be modified over time; however, it is hard to find them in financial statements of insurers which are released to the public, so it's not possible to determine the real ultimate loss of an insurer for each year.

*Robust optimization* is one of the leading optimization methodologies to handle uncertainty [28]. In classical optimization modeling input parameters are considered as certain values. However, in real cases we are not certain about all parameter values as mentioned before. Robust optimization is a new approach to incorporate uncertainty within mathematical models. The approach based on robust optimization is the most preferred method among practitioners due to its applicability. Recently the robust optimization techniques become very popular among practitioners and have been applied in different context [29].

Soyster was the first one who addressed the uncertainty in optimization [30]. Soyster considered the worst possible cases for each data input realization where we may lose some part of the optimality but the final solution remains feasible for all possible cases. Ben-Tal and Nemirovski developed a new robust method based on cone programming. They proposed an ellipsoidal uncertainty which turned an ordinary linear programming problem into nonlinear programming [31]. Although, Ben-Tal and Nemirovski's robust approach has proven to be efficient but it requires solving nonlinear optimization problem which is not popular among many practitioners.

Bertsimas and Sim proposed another robust optimization method for linear programming problem under uncertainty [28]. They assumed the uncertainty set followed a polyhedral shape. Proposing new type of norm, they proved that the method held linearity of the problem. Gharakhani et al. developed such robust DEA model for educational context to measure the efficiency of public high-Scholl in Iran [32]. In this paper we apply a similar approach to insurance companies. In the next section, we formulate DEA model based on Bertsimas et al. approach [28]. In order to get familiar with the type of uncertainty which is used in this paper, consider the following standard linear programming problem:

$$\begin{aligned} \min \quad & c'x \\ \text{subject to} \quad & Ax \geq b, \\ & x \in X. \end{aligned} \tag{3.8}$$

The uncertainty is assumed to influence the technical coefficient matrix  $A$ . In order to introduce uncertainty in the coefficients, consider a particular row  $i$  of the matrix  $A$ , and let  $j$  represent the set of uncertain coefficient in row  $i$ . Each entry  $\tilde{a}_{ij}, j \in J_i$  is assumed as a symmetric and bounded random variable which only can take values in  $[a_{ij} - \hat{a}_{ij}, a_{ij} + \hat{a}_{ij}]$  centered at the point  $a_{ij}$  that is the expected value and  $\hat{a}_{ij}$  denotes the maximum possible deviation from the corresponding estimate.

Bertsimas and Sim proposed an approach for linear optimization that provides full control on the degree of conservatism and keeps the advantages of the linear framework of Soyster. They defined the scaled deviation from nominal value of  $a_{ij}$  as shown here;

$$\eta_{ij} = \frac{\tilde{a}_{ij} - a_{ij}}{\hat{a}_{ij}}, \quad (3.9)$$

$$\sum_{j=1}^n \eta_{ij} \leq \Gamma_i, \quad \forall i = 1, \dots, m,$$

where  $\eta_{ij}$  has an unknown but symmetric distribution which could only take value within  $[-1, 1]$ . Despite the fact that the cumulative scaled deviation of constraint  $i$  can take no value within  $[-n, n]$ , but it is assumed to be confined. In this approach,  $\Gamma_i$  is some control parameter known as the price of robustness. These parameters adjust the robustness of the method against the level of conservatism of the solution. For  $\Gamma_i = 0$ , we get nominal model and no uncertain coefficient involved. On the other hand,  $\Gamma_i = n$  means that the  $i$ th constraint of the problem is protected against all possible realizations of uncertain coefficients. For any value  $\Gamma$  within  $(0, n)$ , the decision maker takes into account a tradeoff between the level of the protection of constraint and the level of solution conservation.

Based on Bertsimas approach, we can reformulate DEA model. The reformulated DEA model, known as robust counterpart is as follows:

$$\begin{aligned} & \max \quad W \\ \text{subject to} \quad & W - \left( \sum_{r=1}^s x_r a_{ro} + z_0 \Gamma + \sum_{r=1}^s p_{r0} \right) \leq 0, \\ & \sum_{r=1}^s x_r a_{rj} - \sum_{i=1}^m v_i b_{ij} + z_j \Gamma + \sum_{r=1}^s p_{rj} \leq 0, \quad \forall j = 1, \dots, n, \\ & z_j + p_{rj} \geq \hat{a}_{rj} y_r, \quad \forall r = 1, \dots, s, \quad \forall j = 1, \dots, n, \\ & \sum_{i=1}^m v_i b_{ij} = 1, \quad \forall j = 1, \dots, n, \\ & -y_r \leq x_r \leq y_r, \quad \forall r = 1, \dots, s, \\ & x_r, z_j, y_r, v_i, p_{rj} \geq 0, \end{aligned} \quad (3.10)$$

where  $a_{rj}$  vector of nominal value of  $\tilde{a}_{rj}$ ,  $\Gamma$  degree of uncertainty within constraint parameters,  $\hat{a}_{rj}$  precision of estimation of  $a_{rj}$ ,  $z_j$  some auxiliary variable related to the robust counterpart denoting the cost of robustness in each constraints,  $p_{rj}$  some auxiliary variable related to the robust counterpart counting the number of uncertain parameters in each constraints, and  $y$  decision variable for making the absolute term  $|x_r|$  to linear one. Other notations are defined in previous equations.

This robust counterpart is obviously a linear programming model which can be solved with popular solver packages. Since original DEA model is linear programming problem,

incorporating uncertainty does not deteriorate solvability of the model. In other words, applying this reformulation preserves the type of original linear problem.

### 3.4. Statistical Analysis of Correlated Data Using Generalized Estimating Equations (GEE)

The *generalized estimating equations* (GEE) method, an extension of the quasi-likelihood approach, is used to analyze longitudinal and other correlated data. Textbooks all advise researchers not to treat observations from the same *cluster* as if they were independent and warn against being misled by *great masses of observations* [33]. Some articles do discuss how much statistical information is obtainable from observations on individuals in clusters such as insurance companies' efficiencies during 8 years. Investigators often have a conservative approach, but GEE approach uses weighted combinations of observations to extract the appropriate amount of information from correlated data. GEEs belong to a class of semi-parametric regression techniques as they rely on specification of only the first two moments. Under mild regularity conditions, parameter estimates from GEEs are consistent [34]. The parameter estimates typically obtained via the Newton-Raphson algorithm.

$\delta_{ij}$  is the response, CRS efficiency score, for DMU  $i$  at time  $j$ ;  $x_{ij}$ , explaining factors, is covariate;  $\beta$  is a  $p \times 1$  vector of unknown regression coefficients;  $g(\cdot)$  is the link function.

Let  $\omega$  be the vector of all (identifiable) parameters of the covariance structure of the observed dependent variables; let  $X_n$  be the fixed matrix collecting all  $x_{ij}$ , and  $\delta_n$  be a vector of response. It is assumed throughout that the interpretation of  $\beta$  does not depend on the value of  $\omega$ . The starting point is the assumption of the existence of a set of unbiased estimating functions for the parameters of the mean structure, denoted as  $g_n \equiv g_n(\delta_n, X_n, \beta)$ , such that  $E(g_n; \beta, \omega | X_n) = 0$  for all possible  $\beta, \omega$ , which are uncorrelated with each other. Optimal estimating functions in a variance minimizing sense with respect to  $g$  are given by

$$g = \sum_{n=1}^N E\left(\frac{\partial g_n}{\partial \beta}\right)^T \text{Cov}^{-1}(g_n) g_n, \quad (3.11)$$

where  $\text{Cov}(g_n)$  is the covariance of  $g_n$ , conditional on  $X_n$ . The use of  $g_n = (\delta_n - \mu_n)$ , where  $\mu_n$  is a correctly specified model of the conditional mean  $E(\delta_n | X_n)$  and is a function of  $\beta$  but not of  $\omega$ , leads to estimating functions which have been referred to as the generalized estimating equations (GEEs) by Liang and Zeger [34]. A GEE estimator of  $\beta$ ,  $\hat{\beta}$ , is obtained as the root of the unbiased estimating functions as follows:

$$0 = \sum_{n=1}^N E\left(\frac{\partial \mu_n}{\partial \beta}\right)^T \text{Cov}^{-1}(\delta_n)(\delta_n - \mu_n), \quad (3.12)$$

where  $\text{Cov}(\delta_n)$  is the covariance of  $\delta_n$ , conditional on  $X_n$ , and depends on  $\omega$ . Usually  $\omega$  is unknown and must be estimated. However, it can be shown that the nuisance parameter,  $\omega$ , has only little impact on  $g$  and on the solution of  $g = 0$  at least for large  $N$ . Thus, replacing  $\omega$  by any consistent estimator  $\hat{\omega}$  of  $\omega$ , for example, the classical minimum distance estimator, the asymptotic variance of  $\hat{\beta}$  is not affected. To complete the estimating equations for  $\beta$ , Liang and Zeger propose a *working correlation matrix*,  $R(\alpha)$ , which is common to all units and is a

*working model* of the correlation structure in the observed dependent variables, where  $\alpha$  is a possibly vector-valued parameter [34].

#### 4. Applying DEA in Institutions Changes Measurement; Cases from Insurance Industry

Due to convenient usage of DEA, the majority of researches have been done in recent years have used DEA approaches in efficiency analysis. Based on working paper of Eling and Luhnen, there are 87 studies which used DEA approach on efficiency analysis in insurance industry. They also found 11 studies of DEA and SFA whose main purpose is assessment of *regulation changes* in insurance industry [35]. We have completed the previous study in Table 3.

#### 5. Output Selection; Value Added versus Financial Intermediary

As it is shown in Table 3, scholars have consensus about input selection, but do not about outputs. Selecting and measuring outputs have been a challenging step in the insurance frontier efficiency studies. There are two major approaches to measure insurers' outputs: the *value-added* approach [23] and the *financial intermediary* approach [36].

The *value-added* approach uses outputs related to the amount of financial services insurance companies provide. In the value-added approach, P/L insurer's outputs consist primarily of intangible financial services; therefore, it is necessary to define suitable proxies that are highly correlated with the quantity of financial services provided. Based on value-added approach and the operating-cost allocations concept developed by Berger and Humphrey in 1992 [37], Cummins and Weiss discussed the three principal services provided by P/L insurers: risk-pooling and risk-bearing, real insurance services, and financial intermediation. They recommend that the most common proxy for the quantity of risk-pooling and real insurance services is losses incurred which is the sum of losses paid plus the net change in loss reserves for the period. Present value of real losses incurred (PV (L)) uses in practice as proxy for loss-based [23].

Alternatively, the *financial intermediary* approach developed by Brockett et al. in 2004 considers three outputs which have the most crucial rule in *financial safety* for three groups of stakeholders: firm's *policyholders*, *employees*, and *regulators*. They use a rule of thumb, *ceteris paribus*, discussed originally by Charnes and Cooper: an increase in an output or alternatively, a decrease in input should be desirable and should improve the efficiency score. This rule challenges the recommendation of Cummins and Weiss where they choose loss incurred as output. When a researcher wants to test whether a particular variable is an input or an output, while all other things being held constant, s/he should check that an increase in the quantity is favorable or unfavorable. They argue that no insurance firm would try to encourage their employees to perform in a manner that engendered large losses while charging premiums similar to their competitors. A firm that pays great losses due to a catastrophe without an appropriate change in premiums may become insolvent, not efficient [36].

*Financial intermediary* view an insurance firm provides a bundle of attributes to the stakeholders. In fact, the pledged payment of losses can be viewed as an intermediate stage by that the insurers collect money, investors get rewarded, consumers get a valued promise of quick claim payment, and consumers, regulators, and employees get a promise of future solvency of the firm. In the financial intermediary approach, the frontier efficiency method

**Table 3:** Summary of previous research.

Papers	Method	Units	Inputs	Outputs
Cummins et al. [9]	DEA-Mamquist	17 Italian life, 58 nonlife, and 19 mixed life insurance companies, 1985–1993.	Wages, administrative wages, fixed capital, equity capital, and other ratios.	Life insurance benefits and charges in reserves, nonlife incurred losses in auto property, in auto liability, in other property, and in other liability, and invested assets.
Fukuyama [10]	DEA-Mamquist	25 Japanese life insurance companies, 1988–1993.	Asset value, number of workers, and tied agents or sales representatives.	Insurance reserves and loans.
Cummins et al. [11]	DEA-Mamquist	US insurers 1981–1990.	Labor costs, materials, policy holders supplied debt capital and equity capital, and real invested assets	Short-tail personal lines, short-tail commercial lines, long-tail personal lines, long-tail commercial lines, and return on assets
Rees et al. [2]	DEA	German and British insurers.	Administration costs and acquisition costs.	Total premium income and annual changes of it for the UK companies, and aggregate sums insured (excluding participation in surpluses) and the annual changes of these for German companies.
Mahlberg and Url [12]	DEA	German insurers.	Administration and distribution costs.	Claims payments + net change in provisions + allocated investment returns, bonuses, and returned premia for each line of business.
Noulas et al. [13]	DEA	11 Greece life Insurance companies, 1991–1996.	Direct cost (claims) and indirect costs (salaries and other expenditures).	Premium income and revenue from investments.
Boonyasai et al. [7]	DEA	Cross-country studies; four life insurance markets: Korea, Philippines, Taiwan, and Thailand.	Salaries, wages, commissions, and business and services expense.	Group life insurance premiums, individual life insurance premiums, and investment income
Mahlberg and Url [14]	DEA	Austrian insurers.	Administration cost, distribution costs and Costs of capital investments.	Claims incurred + net change in provisions + allocated investment returns, bonuses, and returned premia.

**Table 3:** Continued.

Papers	Method	Units	Inputs	Outputs
Ennsfellner et al. [15]	SFA	Austrian health, life and nonlife insurance companies, 1994–1999.	Health, life, and nonlife: net operating expenses, equity capital, and technical provisions net of reinsurance.	Health and life: incurred benefits net of reinsurance, changes in reserves net of reinsurance, and total invested assets. Nonlife: claims incurred net of reinsurance, total invested assets.
Barros et al. [16]	DEA-Mamquist	27 Portugal insurance companies.	Wages, capital, total investment income, and premiums issued.	Claims paid and profits.
Cummins and Rubio-Misés [6]	DEA	Spanish stock and mutual life insurance companies, 1989–1997.	Price of nonlife output, price of life output, labor input, materials, equity capital, debt capital, price of labor, price of materials, price of equity capital, price of debt capital, total costs, total assets, nonlife premiums, life premiums, net income, reserves /total assets, net income /equity /income, debt capital/total capital, equity capital/total assets, and net income/total assets.	Total output, nonlife output, and life output.
Badunenko et al. [17]	DEA	160 Ukrainian insurance companies 2003–2005.	Equity, liabilities, and fixed and current assets	Various types of premiums, such as personal, property, liability, and so forth
Hussels and Ward [3]	DEA	German and british life insurance companies 1991–2002.	Annual average number of employees, total assets minus total liabilities.	Net written premiums, additions to reserves.
Barros et al. [18]	DEA	Nigerian insurers 1994–2005	Total operating costs, total number of employees, and total investments.	Profit and loss account, net premiums, settled claims, outstanding claims, and investment.
Jeng and Lai [5]	DEA	Taiwanese life insurance companies 1981–2004	Home office labor, agent labor, business service, and equity capital.	Benefit payments disaggregated into four categories: ordinary life insurance, personal accident insurance, individual health insurance, and group insurance, and increase in policy reserve (as the output of intermediation function).
Barros et al. [19]	Two-stage DEA, bootstrapped	Greek life insurance companies 1994–2003.	Labor cost, nonlabor cost, and equity capital.	Invested assets, losses incurred, reinsurance reserves, and own reserves.

is used as a goal-directing technique in which the firm's managers balance maintaining short-term claim paying ability and preserving the long-term ability to meet its fiduciary responsibilities against earning a reasonable financial return.

Leverty and Grace empirically examine two approaches for measuring outputs in property-liability insurer efficiency studies. Their study shows that the *value-added* approach is closely related to traditional measures of firm performance, but the *financial intermediary* approach is not. In addition, efficient value-added approach firms are less likely to go insolvent in comparison with firms characterized as efficient by the *financial intermediary* approach. They also find that the theoretical concern regarding the value-added approach's use of losses as a measure of output is not validated empirically then they go to conclude that the value-added approach is the appropriate measure for insurer efficiency [38]. Based on work of Leverty and Grace, we choose *value-added* approach in our research.

## 6. Computational Results

### 6.1. First Stage: Efficiency Scores, Traditional DEA versus Robust DEA Results

To calculate insurers' efficiencies, we used panel data for the years 2003–2010, obtained from the Central Insurance yearbook of IR Iran on 20 mixed (general) insurance companies for a period of 8 years. The sample consists of 139 observations. The insurance companies that are considered in this analysis represent almost all of the market. Keeping in mind the popular DEA rule of thumb, the number of companies in each year is almost greater than three times the number of inputs plus output.

To determine inputs, we followed previous works discussed in the literature review section, and for outputs we followed *value-added* approach as described in Cummins and Weiss [23] but in order to capture the intermediary function of insurance companies we added ROE as an output. In short, we measured output by: (1) losses incurred; (2) return on equity (ROE); measured inputs by (1) number of employees; (2) general and administrative expenses' (3) surplus that is total asset minus total liabilities. It should be mentioned here that all insurance companies in Iran are general (mixed), and in the last 10 years the average of life insurance share in total premium of Iran's market is around 7.5%. In fact Iranian insurers mostly do business in P&L line. This helps us to consider loss incurred as output for all observations. Table 4 presents the CRS efficiency scores for the Iranian insurance companies. Some conclusions can be stemmed from Table 4.

First, based on DEA with certain and uncertain outputs, the overall inefficiency gap for the Iranian insurance industry is 0.18 and 0.3, respectively. However, it should be mentioned that these gaps do not follow any particular trend. This finding implies that in overall, no major change could be found in the market. It is also true about standard deviations in both DEA models (see Table 4).

Second, as expected, CRS efficiency of models with certain outputs is always greater than that of with uncertain outputs, but the gaps between private and state owned insurers is deferent. As it is shown in Figure 2, the average gap in private insurers is much higher than that of in state owned insurers. It could be observed that the private insurers would be much more venerable if the conditions changed. It convince us to work with Robust CRS efficiency scores instead of traditional CRS scores because of the fact that in case of any

**Table 4:** CRS relative efficiency observed in Iranian insurance companies in percent (%): 2003–2010; DEA with certain and uncertain outputs (traditional DEA versus robust DEA).

DMUs	2003			2004			2005			2006			2007			2008			2009			2010			
	Certain outputs	Uncertain outputs	Certain outputs	Certain outputs	Uncertain outputs	Certain outputs	Certain outputs	Uncertain outputs	Certain outputs	Certain outputs	Uncertain outputs	Certain outputs	Certain outputs	Uncertain outputs	Certain outputs	Certain outputs	Uncertain outputs	Certain outputs	Certain outputs	Uncertain outputs	Certain outputs	Certain outputs	Uncertain outputs	Certain outputs	Certain outputs
DMU1 <sup>a</sup>	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
DMU2 <sup>a</sup>	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
DMU3 <sup>a</sup>	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
DMU4 <sup>a</sup>	46	51	45	72	61	56	42	64	64	64	64	71	71	74	73	73	73	73	73	73	73	73	73	73	73
DMU5	41	41	69	33	26	14	27	20	56	51	35	34	34	34	34	34	34	34	34	34	34	34	34	34	34
DMU6	100	88	71	58	100	99	79	79	75	74	74	78	78	100	100	100	100	100	100	100	100	100	100	100	100
DMU7	70	60	67	61	57	66	64	98	98	100	90	87	77	77	69	69	69	69	69	69	69	69	69	69	69
DMU8	86	71	100	100	86	81	100	100	100	100	53	100	100	100	100	100	100	100	100	100	100	100	100	100	
DMU9	32	31	49	42	55	55	51	26	51	28	54	54	54	54	54	54	54	54	54	54	54	54	54	54	54
DMU10	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
DMU11	100	94	100	78	100	44	100	62	65	31	52	38	44	44	44	44	44	44	44	44	44	44	44	44	44
DMU12	100	100	100	100	100	71	100	33	100	97	100	74	100	100	100	100	100	100	100	100	100	100	100	100	
DMU13	100	63	56	54	50	79	78	95	95	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
DMU14	92	46	21	6	68	57	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
DMU15																									
DMU16																									
DMU17																									
DMU18																									
DMU19																									
DMU20																									
Average	83	79	83	73	77	65	78	66	84	76	82	74	86	79	82	82	82	82	82	82	82	82	82	82	
Standard deviation	26	26	20	27	29	32	21	27	21	27	21	29	23	26	19	26	21	21	21	21	21	21	21	21	

<sup>a</sup>State-owned insurers.

change in outputs such as ROE or loss incurred, the efficiency of private insurers will change dramatically. Indeed, the right model for this uncertain data is robust model.

Third, it seems that some companies could be able to compete with their state-owned companies counterparts. For example, DMU8, DMU10, and DMU14 which established in 2003, 2003, and 2004, respectively, could reach to the state owned insurers (DMU1, DMU2, DMU3, and DMU4). It shows that state-owned insurers are not secure. They should upgrade the quality of their management practices, responding to the results of the present research (see Figure 3).

The reason why these insurers could be able to catch their state-owned counterparts is out of our research scope, and can be done in separate research. But it can be briefly noted that all these companies are captive.

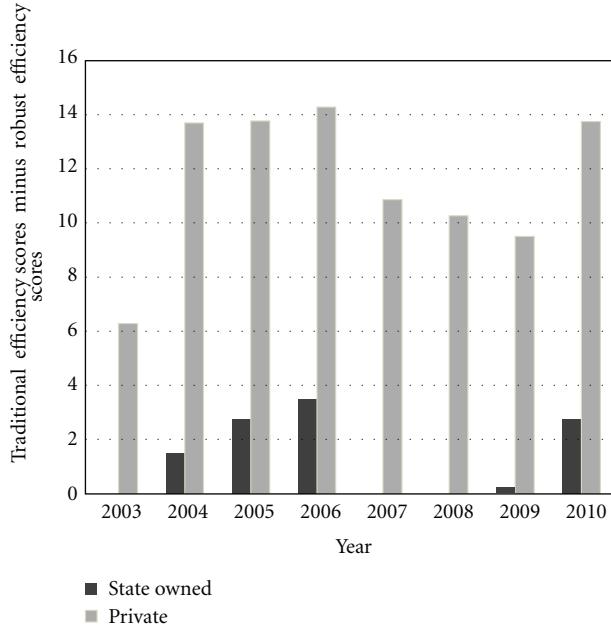
## **6.2. Second Stage: Finding the Factors Explaining Efficiency Scores**

To find the factors explaining efficiency scores, first we considered the research of Barros et al. [19], and then we modified it to fit with insurance industry in Iran. Barros et al. defined determinant factors as *Life*, *Nonlife*, *M&A*, *Foreign*, *Big*, *Quoted*, *MkShare*, and *CastNew*. *Life* is a dummy variable, which is one for life insurance companies. *Nonlife* is a dummy variable which is one for nonlife insurance companies. *M&A* is a dummy variable which is one for enterprises linked to mergers and acquisitions. *Foreign* is a dummy variable, which is equal to one for foreign insurance companies in the sample. *Big* is a dummy variable which is one for big companies measured by the total value of asset. *Quoted* is a dummy variable which is one for companies quoted in the stock market. *MkShare* is the logarithm of the market share of the insurance companies analyzed. *CastNew* is the logarithm of the ratio equity/invested assets.

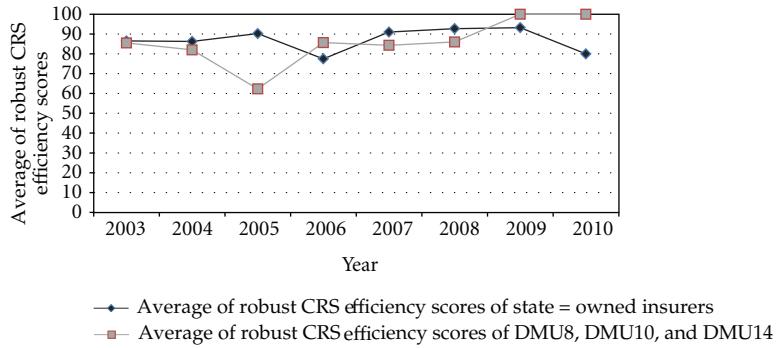
Due to the fact that all Iranian insurers are general, we omitted the *Life* and *NonLife* factors. And because of the fact that there is not any evidence of merger and acquisition, we did not consider it too. There is not any foreign insurer in Iran therefore this factor was not considered as well. Besides, we consider some other factors that are significant in Iran.

Figure 4 demonstrates two significant factors, average premium to surplus (P/S) ratio and ownership state of companies, in Iran from 2003 to 2010. The bars are representative of average P/S ratio and the lines show the average efficiency scores. The P/S ratio is an insurer's underwriting risk indication which is typically laid between 100% and 300%. The more an insurer is solvent, the more its P/S is closer to 100%. However, as illustrated in Figure 4 this ratio for Iranian companies varies considerably beyond its regular range, from below 100% up to 1200% due to absence of solvency regime. It is obvious that average P/S ratio for state-owned companies fluctuates in a noticeably higher range than private companies. Considering this challenge, the average efficiency of private companies is always lower than state-owned companies. Based on Figure 4 we conclude that ownership and P/S may have a significant effect on efficiency scores.

The *capital structure* for different types of companies is also compared in Figure 5. We use *financial leverage index* (FLI) as the main measure of the capital structure which is calculated through dividing return on equity (ROE) by return on asset (ROA). The FLI ratio is laid between 0 and 100%. FLI ratio of a company which creates higher liability with a small amount of capital is closer to 0. It could be observed from the Figure 5 that FLI varies increasingly between different types of companies. While their difference in capital structure was significant in the early years of observation period, it decreases over the course of



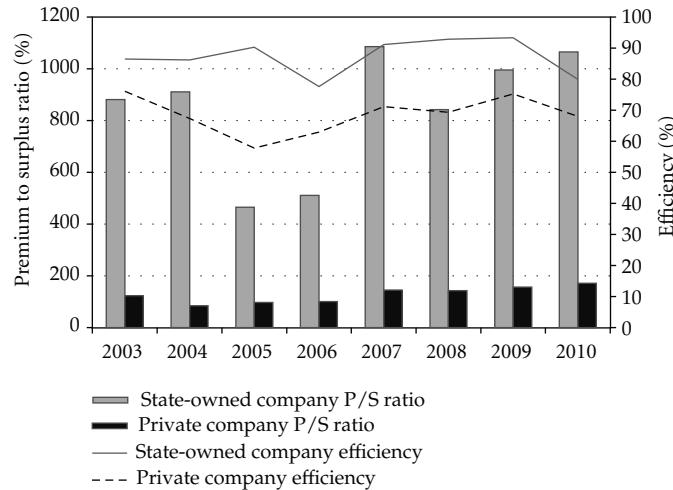
**Figure 2:** Average gap between traditional efficiency scores and robust efficiency scores for different types of insurance companies.



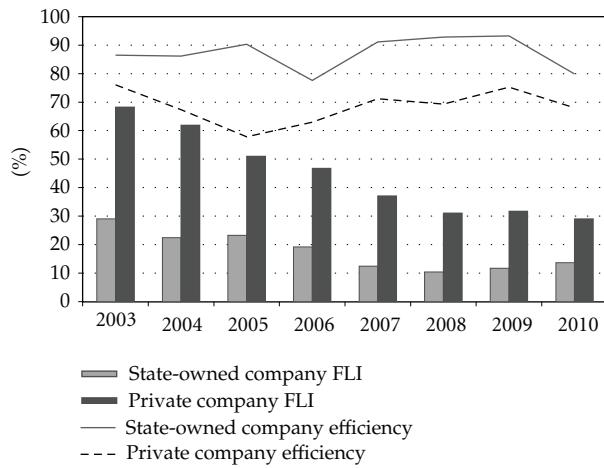
**Figure 3:** Comparison between average of robust CRS efficiency scores for state owned insurers and selective private insurers.

institutions change. Based on Figure 5 we conclude that FLI may have a significant effect on efficiency scores. This factor is similar to *CastNew* which is used in Barros et al. [19].

Based on the material before, to find the most significant factors explaining efficiency scores we consider six factors for next stage, GEE model. The factors are: *Ownership*, *Being in stock exchange market*, *Insurer's size*, *Insurer's market share*, *capital structure*, and *solvency index*.



**Figure 4:** Comparison between premium to surplus ratio and efficiency for different types of insurance companies.



**Figure 5:** Comparison between capital structure and efficiency for different types of insurance company Financial Leverage Index (FLI), a ratio indicating the capital structure, is defined as the ratio between assets to equity.

## 7. Second Stage: The Most Important Factors Explaining Efficiencies

In order to examine the hypothesis that Insurers' efficiency is explained by different contextual variables, we used the two-stage method, as proposed by Coelli et al. [39] estimating the regression as shown below:

$$\begin{aligned} \hat{\delta}_{it} = & \beta_1 + \beta_2 \times Private_{it} + \beta_3 \times Stock_{it} + \beta_4 \times Big_{it} + \beta_5 \times Marketshare_{it} \\ & + \beta_6 \times Capitalstructure_{it} + \beta_7 \times Prmtosrp_{it}, \end{aligned} \quad (7.1)$$

where  $\hat{\delta}_{it}$  represents the CRS efficiency score of insurer  $i$  at time  $t$ . *Private* is a dummy variable, which is one for state-owned insurance companies. The inclusion of this variable is based on the assumption that private companies may exhibit higher efficiency due to type of ownership. *Stock* is a dummy variable which is one for insurance companies whose shares could be traded on the stock exchange aiming to capture the effect of transparency due to the stock market governance requirements. *Big* is a dummy variable which is one for big companies, and is measured by the total value of assets. Following Barros et al. [19] we set a limit for companies' total assets. If a company's total assets values more than \$300 million U.S., it is considered as a big company, and variable *Big* for that is equal to 1. It helps us to summarize data without losing valuable information because there are two major categories of companies in Iran. Some of them are old and huge, and the others are recently established and small. *Marketshare* is the market share of each insurance company within the market. *Capitalstructure* is the ratio of equity to total assets (*Equity to total assets* is equal to dividing *return on asset (ROA)* by *return on equity (ROE)* or  $FLI^{-1}$ ) or  $FLI^{-1}$ , aiming to capture the effect of capital structure. *PrmtoSrp* is the ratio of premium divided by surplus aiming to measure the risk-aversion behavior of insurers and the company's attitude toward solvency issue.

It is obvious that an insurer's efficiency in a given year is correlated with its efficiency in the other years. In other words, there are correlations among the yearly observations belong to an insurer. GEE is an extension of the quasi-likelihood approach, is used to analyze longitudinal and other correlated data. Some articles do discuss how much statistical information is obtainable from observations on individuals in clusters such as cluster of an insurance company's efficiencies during 8 years.

Following Liang and Zeger [34], we employ GEE method to determine the most important variables, with 139 observations categorized into 20 clusters. The results are presented in Table 5. Lots of models are estimated in order to compare the results. The results are quite stable since the variables that were significant in the Model 1, remained significant after dropping the insignificant variables. Based on Table 5, we can conclude that *private* contributes negatively to efficiency, signifying that this type of companies face different constraints in the Iranian insurance market. Second, the variable *Capitalstructure* shows that the specific capital structure of Iranian insurance companies exercises a positive effect on efficiency. Finally the variable *PrmtoSrp* has a positive influence on efficiency. This result along with the result of *capitalstructure* shows that the companies which do not respect to solvency measures are more efficient than the others. *Marketshare*, *Big*, and *Stock* do not have a significant effect on efficiency.

Since the GEE model makes a working correlation matrix in order to obtain the optimal estimators, we test higher degrees of correlations between with-in-the-subjects observations. This can be seen in Table 6, where we estimate parameter with  $M = 5$ ,  $M = 6$ , and  $M = 7$ . This means that the insurer's efficiency score in a given year is related to the insurer's efficiency in adjacent 5 years if we set  $M = 5$ . It is obvious that  $m$  cannot be more than 7 due to the fact that the maximum number of observations within a cluster or with-in-the-subject is equal to 8 (a company's efficiency scores during 8 years). Table 6 shows that *Private* and *PrmtoSrp* are consistently significant while we increase the degree of with-in-the-subject dependency.

In GEE model, QIC is a measure used to choose best correlation structure which can be applied to determine the best subsets of covariates for a particular model. The best model is the one with the smallest QIC value. If we consider QIC as the model's goodness of fit, it gets better while we increase  $M$ , the with-in-the-subject dependency.

**Table 5:** Second stage—Generalized estimating equations models that capture the most important factors explaining the efficiency score for robust DEA.

Predictors	Models											
	Model 1		Model 2		Model 3		Model 4					
	B	Hypothesis Test-Sig.										
(Intercept)	.681	.006	.681	.005	.779	.000	1.141	.000				
Private	-.367	.000	-.370	.000	-.352	.000	-.402	.000				
Stock	.118	.143	.118	.138	—	—	—	—				
Big	.165	.193	.172	.129	.178	.093	—	—				
Marketshare	.001	.751	—	—	—	—	—	—				
Capitalstructure	.007	.003	.007	.003	.006	.004	.005	.005				
PrmtoSrp	.018	.000	.018	.000	.017	.000	.017	.000				
QIC <sup>a</sup>	46.367		44.537		41.933		35.142					
QICC <sup>a</sup>	25.739		23.785		21.424		19.649					
Residual normal test <sup>b</sup>								.224				
Asymp. Sig. (2-tailed)												

<sup>a</sup> Computed using the full log quasilielihood function.<sup>b</sup> Being normal is not rejected at 5% level if the statistic is more than 0.05.**Table 6:** Sensitivity analyzing of significant factors, *Private* and *PrmtoSrp*, based on the within-subject dependencies.

Predictors	Working Correlation Matrix Structure					
	5-dependant		6-dependant		7-dependant	
	B	Hypothesis Test-Sig.	B	Hypothesis Test-Sig.	B	Hypothesis Test-Sig.
(Intercept)	1.291	.000	.986	.000	1.040	.000
Private	-.333	.000	-.168	.004	-.203	.000
PrmtoSrp	.013	.117	.014	.009	.009	.012
QIC <sup>a</sup>	29.944		24.579		19.816	
QICC <sup>a</sup>	15.427		14.501		14.471	
Residual normal test <sup>b</sup>	.067		.244		.248	
Asymp. Sig. (2-tailed)						

<sup>a</sup> Computed using the full log quasi-likelihood function.<sup>b</sup> Being normal is not rejected at 5% level if the statistic is more than 0.05.

## 8. Concluding Remarks

In this paper, we have analyzed technical efficiency for Iranian insurance companies between 2003 and 2010, a period that insurers experienced intense volatility due to the entrance deregulation of the market. We propose a two-stage procedure to analysis the most important determinants affecting efficiency scores. In the first stage, we obtained the CRS efficiency scores by robust DEA model proposed by Bertsimas and Sim in 2003. In the second stage, we determined the most important factors that can explain the efficiency scores by using GEE developed by Liang and Zeger in 1986. The major results of our study are that *ownership type* and *failure to meet the risk management rules* are the main drivers of efficiency. In other words, any state owned insurer which issued more policies without respect to the sufficient capital

provision could obtain better score. It should be mentioned here that state owned companies issued more policies and paid more loss, in the hope that the government supports in case of difficult financial situations. Indeed, instead of issuing insurance policies in proportion to their capital, the insurance policies issued on behalf the government credit. This finding shows the violation of competition rules by state, and the inadequacies of the institutions necessary for private sector development.

What should the managers of inefficient insurance companies do to improve efficiency? First, in order to prepare the institutions, they must pursue the new regulations that require state-owned insurers to have sufficient capital and prevent them to issue insurance policies unlimitedly. Also, following the Williamson if we consider culture as the first level of institutions [40], it seems that private companies should try to change the commonly thought that state-owned companies are more reliable. This cannot be achieved unless their quality of services is as good as their state-owned counterparts, and it cannot be done unless they design and manage their processes efficiently. They should establish a benchmark management procedure in order to evaluate their relative position and to adopt appropriate managerial procedures for catching up with the frontier of "best practices." It seems that some private companies have been able to compete with their state-owned counterparts. Finally, the regulatory authority has an important role in making a fair business environment and improving the efficiency of insurers by (1) participation in developing new rules of business and in enforcing its regulatory duties. (2) Developing indicators to monitor solvency and requiring state-owned companies to comply with. One of the central insurance of IR Iran initial duties is to arrange the market so that insurers could compete fairly while they meet at least the minimum required solvency margin. (3) Publishing information in order to introduce greater transparency into the market especially those related to the sufficient capital and reserves for future commitments.

## References

- [1] Management and Planning Organization, "The law of the third economic, social, and cultural development plan," Tech. Rep., Management and Planning Organization, Tehran, Iran, 2003.
- [2] R. Rees, E. Kessner, P. Klemperer, and C. Matutes, "Regulation and efficiency in European insurance markets," *Economic Policy*, no. 29, pp. 365–397, 1999.
- [3] S. Hussels and D. R. Ward, "The impact of deregulation on the German and UK life insurance markets: an analysis of efficiency and productivity between 1991–2002," Working Paper, Cranfield Research Paper Series, 2007.
- [4] J. L. Wang, L. Y. Tzeng, and E. L. Wang, "The nightmare of the leader: the impact of deregulation on an oligopoly insurance market," *Journal of Insurance*, vol. 26, no. 1, pp. 15–28, 2003.
- [5] V. Jeng and G. C. Lai, "The impact of deregulation on efficiency: an analysis of life insurance industry in Taiwan from 1981 to 2004," *Risk Management and Insurance Review*, vol. 11, no. 2, pp. 349–375, 2008.
- [6] J. D. Cummins and M. Rubio-Misas, "Deregulation, consolidation, and efficiency: evidence from the Spanish insurance industry," *Journal of Money, Credit and Banking*, vol. 38, no. 2, pp. 323–355, 2006.
- [7] T. Boonyasai, M. F. Grace, and H. D. Skipper Jr., *The Effect of Liberalization and Deregulation on Life Insurer Efficiency [M.S. thesis]*, Georgia State University, Atlanta, Ga, USA, 1999.
- [8] Central Insurance of IR Iran, *Statistical Year Book 2009-2010*, Central Insurance of IR Iran, Tehran, Iran.
- [9] J. D. Cummins, G. Turchetti, and M. A. Weiss, "Productivity and technical. Productivity and technical efficiency in the Italian insurance industry," Working Paper 96-10, Wharton School, Philadelphia, Pa, USA, 1996.
- [10] H. Fukuyama, "Investigating productive efficiency and productivity changes of Japanese life insurance companies," *Pacific Basin Finance Journal*, vol. 5, no. 4, pp. 481–509, 1997.
- [11] J. D. Cummins, M. A. Weiss, and H. Zi, "Organizational form and efficiency: the coexistence of stock and mutual property-liability insurers," *Management Science*, vol. 45, no. 9, pp. 1254–1269, 1999.

- [12] B. Mahlberg and T. Url, "The transition to the single market in the German insurance industry," Working Paper, Austrian Institute of Economic Research, 2000.
- [13] A. G. Noulas, T. Hatzigayios, J. Lazaridis, and K. Lyroudi, "Non-parametric production frontier approach to the study of efficiency of non-life insurance companies in Greece," *Journal of Financial Management and Analysis*, vol. 14, no. 1, pp. 19–26, 2001.
- [14] B. Mahlberg and T. Url, "Effects of the single market on the Austrian insurance industry," *Empirical Economics*, vol. 28, no. 4, pp. 813–838, 2003.
- [15] K. C. Ennsfellner, D. Lewis, and R. I. Anderson, "Production efficiency in the Austrian insurance industry: a Bayesian examination," *Journal of Risk and Insurance*, vol. 71, no. 1, pp. 135–159, 2004.
- [16] C. P. Barros, N. Barroso, and M. R. Borges, "Evaluating the efficiency and productivity of insurance companies with a malmquist index: a case study for Portugal," *Geneva Papers on Risk and Insurance*, vol. 30, no. 2, pp. 244–267, 2005.
- [17] O. Badunenko, B. Grechanyuk, and O. Talavera, "Development under regulation: the way of the Ukrainian insurance market," Discussion Papers of DIW Berlin, German Institute for Economic Research, Berlin, Germany, 2006.
- [18] C. P. Barros, A. Ibiwoye, and S. Managi, "Productivity change of Nigerian insurance companies: 1994–2005," *African Development Review*, vol. 20, no. 3, pp. 505–528, 2008.
- [19] C. P. Barros, M. Nektarios, and A. Assaf, "Efficiency in the Greek insurance industry," *European Journal of Operational Research*, vol. 205, no. 2, pp. 431–436, 2010.
- [20] D. Stalb and L. Bevere, *World Insurance in 2010*, Swiss Reinsurance, 2011.
- [21] R. W. Shephard, *Theory of Cost and Production Functions*, Princeton University Press, Princeton, NJ, USA, 1970.
- [22] A. Charnes, W. W. Cooper, A. Y. Lewin, and L. M. Seiford, *Data Envelopment Analysis: Theory, Methodology and Applications*, Kluwer Academic, New York, NY, USA, 1995.
- [23] J. D. Cummins and M. A. Weiss, "Analyzing firm performance in the insurance industry using frontier efficiency methods," Center for Financial Institutions Working Papers, Wharton School Center for Financial Institutions, University of Pennsylvania, 1998.
- [24] D. Deprins, L. Simar, and H. Tulkens, "Measuring labor-efficiency in post offices," in *The Performance of Public Enterprises: Concepts and Measurement*, pp. 243–268, Elsevier Science, Amsterdam, The Netherlands, 1984.
- [25] J. D. Cummins and H. Zi, "Comparison of frontier efficiency methods: an application to the U.S. life insurance industry," *Journal of Productivity Analysis*, vol. 10, no. 2, pp. 131–152, 1998.
- [26] A. Charnes, W. W. Cooper, and E. Rhodes, "Measuring the efficiency of decision making units," *European Journal of Operational Research*, vol. 2, no. 6, pp. 429–444, 1978.
- [27] S. T. Cooper and E. Cohn, "Estimation of a frontier production function for the South Carolina educational process," *Economics of Education Review*, vol. 16, no. 3, pp. 313–327, 1997.
- [28] D. Bertsimas and M. Sim, "Robust discrete optimization and network flows," *Mathematical Programming B*, vol. 98, no. 1–3, pp. 49–71, 2003.
- [29] M. A. Shafiaa, M. Aghaeeb, and A. Jamilia, "A new mathematical model for the job shop scheduling problem with uncertain processing times," *International Journal of Industrial Engineering Computations*, vol. 2, pp. 295–306, 2010.
- [30] A. Soyster, "Convex programming with set-inclusive constraints and applications to inexact linear programming," *Operations Research*, vol. 21, no. 5, pp. 1154–1157, 1973.
- [31] A. Ben-Tal and A. Nemirovski, "Robust solutions of linear programming problems contaminated with uncertain data," *Mathematical Programming B*, vol. 88, no. 3, pp. 411–424, 2000.
- [32] M. Gharakhani, I. Kazemi, and H. Alizadeh Haji, "A robust DEA model for measuring the relative efficiency of Iranian high schools," *Management Science Letter*, vol. 1, no. 3, pp. 389–404, 2011.
- [33] J. A. Hanley, A. Negassa, M. D. D. B. Edwardes, and J. E. Forrester, "Statistical analysis of correlated data using generalized estimating equations: an orientation," *American Journal of Epidemiology*, vol. 157, no. 4, pp. 364–375, 2003.
- [34] K. Y. Liang and S. L. Zeger, "Longitudinal data analysis using generalized linear models," *Biometrika*, vol. 73, no. 1, pp. 13–22, 1986.
- [35] M. Eling and M. Luhnen, "Frontier efficiency methodologies to measure performance in the insurance industry: overview and new empirical evidence," University of St. Gallen Working Papers on Risk Management and Insurance Paper 56, 2008.
- [36] P. L. Brockett, W. W. Cooper, L. L. Golden, J. J. Rousseau, and Y. Wang, "Financial intermediary versus production approach to efficiency of marketing distribution systems and organizational structure of insurance companies," *Journal of Risk and Insurance*, vol. 72, no. 3, pp. 393–412, 2005.

- [37] A. N. Berger and D. B. Humphrey, "Measurement and efficiency issues in commercial banking," in *Output Measurement in the Service Sectors*, pp. 245–300, University of Chicago Press, Chicago, Ill, USA, 1992.
- [38] J. T. Leverty and M. F. Grace, "The robustness of output measures in property-liability insurance efficiency studies," *Journal of Banking and Finance*, vol. 34, no. 7, pp. 1510–1524, 2010.
- [39] T. J. Coelli, P. Rao, and G. E. Battese, *An Introduction to Efficiency and Productivity Analysis*, Kluwer Academic Press, New York, NY, USA, 1998.
- [40] O. E. Williamson, "The new institutional economics: taking stock, looking ahead," *Journal of Economic Literature*, vol. 38, no. 3, pp. 595–613, 2000.

*Research Article*

# Solving Constrained Global Optimization Problems by Using Hybrid Evolutionary Computing and Artificial Life Approaches

**Jui-Yu Wu**

*Department of Business Administration, Lunghwa University of Science and Technology, No. 300, Section 1, Wanshou Road, Guishan, Taoyuan County 333, Taiwan*

Correspondence should be addressed to Jui-Yu Wu, jywu@mail.lhu.edu.tw

Received 28 February 2012; Revised 15 April 2012; Accepted 19 April 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Jui-Yu 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.

This work presents a hybrid real-coded genetic algorithm with a particle swarm optimization (RGA-PSO) algorithm and a hybrid artificial immune algorithm with a PSO (AIA-PSO) algorithm for solving 13 constrained global optimization (CGO) problems, including six nonlinear programming and seven generalized polynomial programming optimization problems. External RGA and AIA approaches are used to optimize the constriction coefficient, cognitive parameter, social parameter, penalty parameter, and mutation probability of an internal PSO algorithm. CGO problems are then solved using the internal PSO algorithm. The performances of the proposed RGA-PSO and AIA-PSO algorithms are evaluated using 13 CGO problems. Moreover, numerical results obtained using the proposed RGA-PSO and AIA-PSO algorithms are compared with those obtained using published individual GA and AIA approaches. Experimental results indicate that the proposed RGA-PSO and AIA-PSO algorithms converge to a global optimum solution to a CGO problem. Furthermore, the optimum parameter settings of the internal PSO algorithm can be obtained using the external RGA and AIA approaches. Also, the proposed RGA-PSO and AIA-PSO algorithms outperform some published individual GA and AIA approaches. Therefore, the proposed RGA-PSO and AIA-PSO algorithms are highly promising stochastic global optimization methods for solving CGO problems.

## 1. Introduction

Many scientific, engineering, and management problems can be expressed as constrained global optimization (CGO) problems, as follows:

$$\begin{aligned} & \text{Minimize} && f(\mathbf{x}), \\ & \text{s.t.} && g_m(\mathbf{x}) \leq 0, \quad m = 1, 2, \dots, M, \end{aligned}$$

$$\begin{aligned}
h_k(\mathbf{x}) &= 0, \quad k = 1, 2, \dots, K, \\
x_n^l \leq x_n \leq x_n^u, \quad n &= 1, 2, \dots, N,
\end{aligned} \tag{1.1}$$

where  $f(\mathbf{x})$  denotes an objective function;  $g_m(\mathbf{x})$  represents a set of  $m$  nonlinear inequality constraints;  $h_k(\mathbf{x})$  refers to a set of  $k$  nonlinear equality constraints;  $\mathbf{x}$  represents a vector of decision variables which take real values, and each decision variable  $x_n$  is constrained by its lower and upper boundaries  $[x_n^l, x_n^u]$ ;  $N$  is the total number of decision variables  $x_n$ . For instance, generalized polynomial programming (GPP) belongs to the nonlinear programming (NLP) method. The formulation of GPP is a nonconvex objective function subject to nonconvex inequality constraints and possibly disjointed feasible region. The GPP approach has been successfully used to solve problems including alkylation process design, heat exchanger design, optimal reactor design [1], inventory decision problem (economic production quantity) [2], process synthesis and the design of separations, phase equilibrium, nonisothermal complex reactor networks, and molecular conformation [3].

Traditional local NLP optimization approaches based on a gradient algorithm are inefficient for solving CGO problems, while an objective function is nondifferentiable. Global optimization methods can be divided into deterministic or stochastic [4]. Often involving a sophisticated optimization process, deterministic global optimization methods typically make assumptions regarding the problem to be solved [5]. Stochastic global optimization methods that do not require gradient information and numerous assumptions have received considerable attention. For instance, Sun et al. [6] devised an improved vector particle swarm optimization (PSO) algorithm with a constraint-preserving method to solve CGO problems. Furthermore, Tsoulos [7] developed a real-coded genetic algorithm (RGA) with a penalty function approach for solving CGO problems. Additionally, Deep and Dipti [8] presented a self-organizing GA with a tournament selection method for solving CGO problems. Meanwhile, Wu and Chung [9] developed a RGA with a static penalty function approach for solving GPP optimization problems. Finally, Wu [10] introduced an artificial immune algorithm (AIA) with an adaptive penalty function method to solve CGO problems.

Zadeh [11] defined “soft computing” as the synergistic power of two or more fused computational intelligence (CI) schemes, which can be divided into several branches: granular computing (e.g., fuzzy sets, rough sets, and probabilistic reasoning), neurocomputing (e.g., supervised, unsupervised, and reinforcement neural learning algorithms), evolutionary computing (e.g., GAs, genetic programming, and PSO algorithms), and artificial life (e.g., artificial immune systems) [12]. Besides, outperforming individual algorithms in terms of solving certain problems, hybrid algorithms can solve general problems more efficiently [13]. Therefore, hybrid CI approaches have recently attracted considerable attention as a promising field of research. Various hybrid evolutionary computing (GA and PSO methods) and artificial life (such as AIA methods) approaches have been developed for solving optimization problems. These hybrid algorithms focus on developing diverse candidate solutions (such as chromosomes and particles) of population/swarm to solve optimization problems more efficiently. These hybrid algorithms use two different algorithms to create diverse candidate solutions using their specific operations and then merge these diverse

candidate solutions to increase the diversity of the candidate population. For instance, Abd-El-Wahed et al. [14] developed an integrated PSO algorithm and GA to solve nonlinear optimization problems. Additionally, Kuo and Han [15] presented a hybrid GA and PSO algorithm for bilevel linear programming to solve a supply chain distribution problem. Furthermore, Shelokar et al. [16] presented a hybrid PSO method and ant colony optimization method for solving continuous optimization problems. Finally, Hu et al. [17] developed an immune cooperative PSO algorithm for solving the fault-tolerant routing problem.

Compared to the above hybrid CI algorithms, this work optimizes the parameter settings of an individual CI method by using another individual CI algorithm. A standard PSO algorithm has certain limitations [17, 18]. For instance, a PSO algorithm includes many parameters that must be set, such as the cognitive parameter, social parameter, and constriction coefficient. In practice, the optimal parameter settings of a PSO algorithm are tuned based on trial and error and prior knowledge is required to successfully manipulate the cognitive parameter, social parameter, and constriction coefficient. The exploration and exploitative capabilities of a PSO algorithm are limited to optimum parameter settings. Moreover, conventional PSO methods involve premature convergence that rapidly losses diversity during optimization.

Fortunately, optimization of parameter settings for a conventional PSO algorithm can be considered an unconstrained global optimization (UGO) problem, and the diversity of candidate solutions of the PSO method can be increased using a multi-nonuniform mutation operation [19]. Moreover, the parameter manipulation of a GA and AIA method is easy to implement without prior knowledge. Therefore, to overcome the limitations of a standard PSO algorithm, this work develops two hybrid CI algorithms to solve CGO problems efficiently. The first algorithm is a hybrid RGA and PSO (RGA-PSO) algorithm, while the second algorithm is a hybrid AIA and PSO (AIA-PSO) algorithm. The proposed RGA-PSO and AIA-PSO algorithms are considered to optimize two optimization problems simultaneously. The UGO problem (optimization of cognitive parameter, social parameter, constriction coefficient, penalty parameter, and mutation probability of an internal PSO algorithm based on a penalty function approach) is optimized using external RGA and AIA approaches, respectively. A CGO problem is then solved using the internal PSO algorithm. The performances of the proposed RGA-PSO and AIA-PSO algorithms are evaluated using a set of CGO problems (e.g., six benchmark NLP and seven GPP optimization problems).

The rest of this paper is organized as follows. Section 2 describes the RGA, PSO algorithm, AIA, and penalty function approaches. Section 3 then introduces the proposed RGA-PSO and AIA-PSO algorithms. Next, Section 4 compares the experimental results of the proposed RGA-PSO and AIA-PSO algorithms with those of various published individual GAs and AIAs [9, 10, 20–22] and hybrid algorithms [23, 24]. Finally, conclusions are drawn in Section 5.

## 2. Related Works

### 2.1. Real-Coded Genetic Algorithm

GAs are stochastic global optimization methods based on the concepts of natural selection and use three genetic operators, that is, selection, crossover, and mutation, to explore and exploit the solution space. RGA outperforms binary-coded GA in solving continuous function optimization problems [19]. This work thus describes operators of a RGA [25].

### 2.1.1. Selection

A selection operation selects strong individuals from a current population based on their fitness function values and then reproduces these individuals into a crossover pool. The several selection operations developed include the roulette wheel, ranking, and tournament methods [19, 25]. This work uses the normalized geometric ranking method, as follows:

$$p_j = q'(1 - q)^{r-1}, \quad j = 1, 2, \dots, ps_{RGA}, \quad (2.1)$$

$p_j$  = probability of selecting individual  $j$ ,  $q$  = probability of choosing the best individual (here  $q = 0.35$ )

$$q' = \frac{q}{1 - (1 - q)^{ps_{RGA}}}, \quad (2.2)$$

$r$  = individual ranking based on fitness value, where 1 represents the best,  $r = 1, 2, \dots, ps_{RGA}$ ,  $ps_{RGA}$  = population size of the RGA.

### 2.1.2. Crossover

While exploring the solution space by creating new offspring, the crossover operation randomly selects two parents from the crossover pool and then uses these two parents to generate two new offspring. This operation is repeated until the  $ps_{RGA}/2$  is satisfied. The whole arithmetic crossover is easily implemented, as follows:

$$\begin{aligned} \mathbf{v}'_1 &= \beta \times \mathbf{v}_1 + (1 - \beta) \times \mathbf{v}_2, \\ \mathbf{v}'_2 &= (1 - \beta) \mathbf{v}_1 + \beta \times \mathbf{v}_2, \end{aligned} \quad (2.3)$$

where  $\mathbf{v}_1$  and  $\mathbf{v}_2$  = parents,  $\mathbf{v}'_1$  and  $\mathbf{v}'_2$  = offspring,  $\beta$  = uniform random number in the interval  $[0, 1.5]$ .

### 2.1.3. Mutation

Mutation operation can increase the diversity of individuals (candidate solutions). Multi-nonuniform mutation is described as follows:

$$x_{trial,n} = \begin{cases} x_{current,n} + (x_n^u - x_{current,n})pert(g_{RGA}) & \text{if } U_1(0,1) < 0.5, \\ x_{current,n} - (x_{current,n} - x_n^l)pert(g_{RGA}) & \text{if } U_1(0,1) \geq 0.5, \end{cases} \quad (2.4)$$

where  $pert(g_{RGA}) = [U_2(0,1)(1 - g_{RGA}/g_{max,RGA})]^2$ , perturbed factor,  $U_1(0,1)$  and  $U_2(0,1)$  = uniform random variable in the interval  $[0, 1]$ ,  $g_{max,RGA}$  = maximum generation of the RGA,  $g_{RGA}$  = current generation of the RGA,  $x_{current,n}$  = current decision variable  $x_n$ ,  $x_{trial,n}$  = trial candidate solution  $x_n$ .

## 2.2. Particle Swarm Optimization

Kennedy and Eberhart [26] first introduced a conventional PSO algorithm, which is inspired by the social behavior of bird flocks or fish schools. Like GAs, a PSO algorithm is a population-based algorithm. A population of candidate solutions is called a particle swarm. The particle velocities can be updated by (2.5), as follows:

$$\begin{aligned} v_{j,n}(g_{\text{PSO}} + 1) = & v_{j,n}(g_{\text{PSO}}) + c_1 r_{1j}(g_{\text{PSO}}) \left[ p_{j,n}^{lb}(g_{\text{PSO}}) - x_{j,n}(g_{\text{PSO}}) \right] \\ & + c_2 r_{2j}(g_{\text{PSO}}) \left[ p_{j,n}^{gb}(g_{\text{PSO}}) - x_{j,n}(g_{\text{PSO}}) \right] \quad j = 1, 2, \dots, p_{\text{PSO}}, \quad n = 1, 2, \dots, N, \end{aligned} \quad (2.5)$$

$v_{j,n}(g_{\text{PSO}} + 1)$  = particle velocity of decision variable  $x_n$  of particle  $j$  at generation  $g_{\text{PSO}} + 1$ ,  
 $v_{j,n}(g_{\text{PSO}})$  = particle velocity of decision variable  $x_n$  of particle  $j$  at generation  $g_{\text{PSO}}$ ,  $c_1$  = cognitive parameter,  $c_2$  = social parameter,  $x_{j,n}(g_{\text{PSO}})$  = particle position of decision variable  $x_n$  of particle  $j$  at generation  $g_{\text{PSO}}$ ,  $r_{1,j}(g_{\text{PSO}})$ ,  $r_{2,j}(g_{\text{PSO}})$  = independent uniform random numbers in the interval  $[0, 1]$  at generation  $g_{\text{PSO}}$ ,  $p_{j,n}^{lb}(g_{\text{PSO}})$  = best local solution at generation  $g_{\text{PSO}}$ ,  $p_{j,n}^{gb}(g_{\text{PSO}})$  = best global solution at generation  $g_{\text{PSO}}$ ,  $p_{\text{PSO}}$  = population size of the PSO algorithm.

The particle positions can be computed using (2.6), as follows:

$$x_{j,n}(g_{\text{PSO}} + 1) = x_{j,n}(g_{\text{PSO}}) + v_{j,n}(g_{\text{PSO}} + 1) \quad j = 1, 2, \dots, p_{\text{PSO}}, \quad n = 1, 2, \dots, N. \quad (2.6)$$

Shi and Eberhart [27] developed a modified PSO algorithm by incorporating an inertia weight ( $\omega_{\text{in}}$ ) into (2.7) to control the exploration and exploitation capabilities of a PSO algorithm, as follows:

$$\begin{aligned} v_{j,n}(g_{\text{PSO}} + 1) = & \omega_{\text{in}} v_{j,n}(g_{\text{PSO}}) + c_1 r_{1j}(g_{\text{PSO}}) \left[ p_{j,n}^{lb}(g_{\text{PSO}}) - x_{j,n}(g_{\text{PSO}}) \right] \\ & + c_2 r_{2j}(g_{\text{PSO}}) \left[ p_{j,n}^{gb}(g_{\text{PSO}}) - x_{j,n}(g_{\text{PSO}}) \right] \quad j = 1, 2, \dots, p_{\text{PSO}}, \quad n = 1, 2, \dots, N. \end{aligned} \quad (2.7)$$

A constriction coefficient ( $\chi$ ) was inserted into (2.8) to balance the exploration and exploitation tradeoff [28–30], as follows:

$$\begin{aligned} v_{j,n}(g_{\text{PSO}} + 1) = & \chi \left\{ v_{j,n}(g_{\text{PSO}}) + \rho_1(g_{\text{PSO}}) \left[ p_{j,n}^{lb}(g_{\text{PSO}}) - x_{j,n}(g_{\text{PSO}}) \right] \right. \\ & \left. + \rho_2(g_{\text{PSO}}) \left[ p_{j,n}^{gb}(g_{\text{PSO}}) - x_{j,n}(g_{\text{PSO}}) \right] \right\} \quad j = 1, 2, \dots, p_{\text{PSO}}, \quad n = 1, 2, \dots, N, \end{aligned} \quad (2.8)$$

where

$$\chi = \frac{2U_3(0, 1)}{|2 - \tau - \sqrt{\tau(\tau - 4)}|}, \quad (2.9)$$

$U_3(0, 1)$  = uniform random variable in the interval  $[0, 1]$ ,  $\tau = \tau_1 + \tau_2$ ,  $\tau_1 = c_1 r_{1j}$ ,  $\tau_1 = c_2 r_{2j}$ .

This work considers parameters  $\omega_{\text{in}}$  and  $\chi$  to update the particle velocities, as follows:

$$\begin{aligned} v_{j,n}(g_{\text{PSO}} + 1) &= \chi \left\{ \omega_{\text{in}} v_{j,n}(g_{\text{PSO}}) + c_1 r_{1j}(g_{\text{PSO}}) [p_{j,n}^{lb}(g_{\text{PSO}}) - x_{j,n}(g_{\text{PSO}})] \right. \\ &\quad \left. + c_2 r_{2j}(g_{\text{PSO}}) [p_{j,n}^{gb}(g_{\text{PSO}}) - x_{j,n}(g_{\text{PSO}})] \right\} \quad j = 1, 2, \dots, p_{\text{PSO}}, \quad n = 1, 2, \dots, N, \end{aligned} \quad (2.10)$$

where  $\omega_{\text{in}} = ((g_{\max, \text{PSO}} - g_{\text{PSO}})/g_{\max, \text{PSO}})$ , increased  $g_{\text{PSO}}$  value reduces the  $\omega_{\text{in}}$ ,  $g_{\max, \text{PSO}}$  = maximum generation of the PSO algorithm.

According to (2.10), the optimal values of parameters  $c_1$ ,  $c_2$ , and  $\chi$  are difficult to obtain through a trial and error. This work thus optimizes these parameter settings by using RGA and AIA approaches.

### 2.3. Artificial Immune Algorithm

Wu [10] presented an AIA based on clonal selection and immune network theories to solve CGO problems. The AIA approach comprises selection, hypermutation, receptor editing, and bone marrow operations. The selection operation is performed to reproduce strong antibodies (**Abs**). Also, diverse **Abs** are created using hypermutation, receptor editing, and bone marrow operations, as described in the following subsections.

#### 2.3.1. Ab and Ag Representation

In the human immune system, an antigen (**Ag**) has multiple epitopes (antigenic determinants), which can be recognized by various **Abs** with paratopes (recognizers), on its surface. In the AIA approach, an **Ag** represents known parameters of a solved problem. The **Abs** are the candidate solutions (i.e., decision variables  $x_n$ ,  $n = 1, 2, \dots, N$ ) of the solved problem. The quality of a candidate solution is evaluated using an **Ab-Ag** affinity that is derived from the value of an objective function of the solved problem.

### 2.3.2. Selection Operation

The selection operation, which is based on the immune network principle [31], controls the number of antigen-specific **Ab**s. This operation is defined according to **Ab-Ag** and **Ab-Ab** recognition information, as follows:

$$p_{r_j} = \frac{1}{N} \sum_{n=1}^N \frac{1}{e^{d_{nj}}}, \quad (2.11)$$

$$d_{nj} = \left| \frac{x_n^* - x_{nj}}{x_n^*} \right|, \quad j = 1, 2, \dots, rs, \quad n = 1, 2, \dots, N,$$

where  $p_{r_j}$  = probability that **Ab**  $j$  recognizes **Ab**\* (the best solution),  $x_n^*$  = the best **Ab**\* with the highest **Ab-Ag** affinity,  $x_{nj}$  = decision variables  $x_n$  of **Ab**  $j$ ,  $rs$  = repertoire (population) size of the AIA.

The **Ab**\* is recognized by other **Ab** $j$  in a current **Ab** repertoire. Large  $p_{r_j}$  implies that **Ab** $j$  can effectively recognize **Ab**\*. The **Ab** $j$  with  $p_{r_j}$  that is equivalent to or larger than the threshold degree  $p_{rt}$  is reproduced to generate an intermediate **Ab** repertoire.

### 2.3.3. Hypermutation Operation

Multi-nonuniform mutation [19] is used as the somatic hypermutation operation, which can be expressed as follows:

$$x_{\text{trial},n} = \begin{cases} x_{\text{current},n} + (x_n^u - x_{\text{current},n}) \text{pert}(g_{\text{AIA}}), & \text{if } U_4(0,1) < 0.5, \\ x_{\text{current},n} - (x_{\text{current},n} - x_n^l) \text{pert}(g_{\text{AIA}}), & \text{if } U_4(0,1) \geq 0.5, \end{cases} \quad (2.12)$$

where  $\text{pert}(g_{\text{AIA}}) = \{U_5(0,1)(1 - g_{\text{AIA}}/g_{\text{max,AIA}})\}^2$  = perturbation factor,  $g_{\text{AIA}}$  = current generation of the AIA,  $g_{\text{max,AIA}}$  = maximum generation number of the AIA,  $U_4(0,1)$  and  $U_5(0,1)$  = uniform random number in the interval  $[0, 1]$ .

This operation has two tasks, that is, a uniform search and local fine-tuning.

### 2.3.4. Receptor Editing Operation

A receptor editing operation is developed using the standard Cauchy distribution  $C(0,1)$ , in which the local parameter is zero and the scale parameter is one. Receptor editing is performed using Cauchy random variables that are generated from  $C(0,1)$ , owing to their ability to provide a large jump in the **Ab-Ag** affinity landscape to increase the probability of escaping from the local **Ab-Ag** affinity landscape. Cauchy receptor editing can be defined by

$$\mathbf{x}_{\text{trial}} = \mathbf{x}_{\text{current}} + U_5(0,1)^2 \times \boldsymbol{\sigma}, \quad (2.13)$$

where  $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \dots, \sigma_N]^T$ , vector of Cauchy random variables,  $U_5(0,1)$  = uniform random number in the interval  $[0, 1]$ .

This operation is employed in local fine-tuning and large perturbation.

### 2.3.5. Bone Marrow Operation

The paratope of an **Ab** can be generated by recombining gene segments  $V_H$   $D_H$   $J_H$  and  $V_L$   $J_L$  [32]. Therefore, based on this metaphor, diverse **Ab**s are synthesized using a bone marrow operation. This operation randomly chooses two **Ab**s from the intermediate **Ab** repertoire and a recombination point from the gene segments of the paratope of the selected **Ab**s. The selected gene segments (e.g., gene  $x_1$  of **Ab** 1 and gene  $x_1$  of the **Ab** 2) are reproduced to create a library of gene segments. The selected gene segments in the paratope are then deleted. The new **Ab** 1 is formed by inserting the gene segment, which is gene  $x_1$  of the **Ab** 2 in the library plus a random variable created from standard normal distribution  $N(0, 1)$ , at the recombination point. The literature details the implementation of the bone marrow [10].

## 2.4. Penalty Function Methods

Stochastic global optimization approaches, including GAs, AIAs, and PSO, are naturally unconstrained optimization methods. Penalty function methods, which are constraint handling approaches, are commonly used to create feasible solutions to a CGO problem and transform it into an unconstrained optimization problem. Two popular penalty functions exist, namely, the exterior and interior functions. Exterior penalty functions use an infeasible solution as a starting point, and convergence is from the infeasible region to the feasible one. Interior penalty functions start from a feasible solution, then move from the feasible region to the constrained boundaries. Exterior penalty functions are favored over interior penalty functions, because they do not require a feasible starting point and are easily implemented. The exterior penalty functions developed to date include static, dynamic, adaptive, and death penalty functions [33]. This work uses the form of a static penalty function, as follows:

$$\text{Minimize } f_{\text{pseudo}}(\mathbf{x}, \rho) = f(\mathbf{x}) + \rho \left\{ \sum_{m=1}^M \left\{ \max[0, g_m(\mathbf{x})] + \sum_{k=1}^K [h_k(\mathbf{x})] \right\}^2 \right\}, \quad (2.14)$$

where  $f_{\text{pseudo}}(\mathbf{x}, \rho)$  = pseudo-objective function obtained using an original objective function plus a penalty term,  $\rho$  = penalty parameter.

Unfortunately, the penalty function scheme is limited by the need to fine-tune the penalty parameter  $\rho$  [8]. To overcome this limitation, this work attempts to find the optimum  $\rho$  for each CGO problem using the RGA and AIA approaches. Additionally, to obtain high-quality RGA-PSO and AIA-PSO solutions accurate to at least five decimal places for the violation of each constraint to a specific CGO problem, the parameter  $\rho$  is within the search space  $[1 \times 10^9, 1 \times 10^{11}]$ .

## 3. Method

### 3.1. RGA-PSO Algorithm

Figure 1 shows the pseudocode of the proposed RGA-PSO algorithm. The external RGA approach is used to optimize the best parameter settings of the internal PSO algorithm, and the internal PSO algorithm is employed to solve CGO problems.

```

Procedure external RGA
begin
     $g_{RGA} \leftarrow 0$ 
    Step 1: Initialize the parameter settings
        (a) parameter setting
        (b) generate initial population
    While  $g_{RGA} \leq g_{max,RGA}$  do
        Step 2: Compute the fitness function value
        Candidate solution  $j, j = 1, 2, \dots, ps_{RGA}$   $\boxed{x \quad c_1 \quad c_2 \quad \rho \quad p_{m,PSO}} \rightarrow$ 
         $fitness_j = f(x_{PSO}^*) \quad \boxed{x_{PSO}} \quad \Leftarrow$ 
        Step 3: Implement a selection operation
        For each candidate solution  $j, j = 1, 2, \dots, ps_{RGA} / 2$  do
            if  $rand(\cdot) \leq p_c$  then
                Step 4: Perform a crossover operation
            endiff
        endFor
        For each candidate solution  $j, j = 1, 2, \dots, ps_{RGA}$  do
            if  $rand(\cdot) \leq p_{m, RGA}$  then
                Step 5: Conduct a mutation operation
            endiff
        endFor
        Step 6: Implement an elitist strategy
         $g_{RGA} \leftarrow g_{RGA} + 1$ 
    end
end

```

```

Procedure internal PSO Algorithm
 $g_{PSO} \leftarrow 0$ 
Step (1) Create an initial particle swarm
    (a) parameter settings from RGA
    (b) generate initial particle swarm
While executing time the predefined fixed total time do
    Step (2) Calculate the objective function value
    Step (3) Update the particle velocity and position
    For each candidate particle  $j, j = 1, 2, \dots, ps_{PSO}$  do
        if  $rand(\cdot) \leq p_{m, PSO}$  then
            Step (4) Implement a mutation operation
        endiff
    endFor
     $g_{PSO} \leftarrow g_{PSO} + 1$ 
end

```

**Figure 1:** The pseudocode of the proposed RGA-PSO algorithm.

Candidate solution1	$x$	$c_1$	$c_2$	$\rho$	$p_{m,PSO}$	$fitness_1$
Candidate solution 2	$x$	$c_1$	$c_2$	$\rho$	$p_{m,PSO}$	$fitness_2$
⋮						
Candidate solution $ps_{RGA}$	$x$	$c_1$	$c_2$	$\rho$	$p_{m,PSO}$	$fitness_{ps_{RGA}}$

**Figure 2:** Chromosome representation of the external RGA.

### External RGA

*Step 1* (initialize the parameter settings). Parameter settings are given such as  $ps_{RGA}$ , crossover probability  $p_c$ , mutation probability of the external RGA approach  $p_{m,RGA}$ , the lower and upper boundaries of these parameters  $c_1, c_2, \chi, \rho$ , and the mutation probability of the internal PSO algorithm  $p_{m,PSO}$ . The candidate solutions (individuals) of the external RGA represent the optimized parameters of the internal PSO algorithm. Finally, Figure 2 illustrates the candidate solution of the external RGA approach.

Candidate solution 1	$x_1$	$x_2$	$\dots$	$x_N$	$f(x_{\text{PSO},1})$
Candidate solution 2	$x_1$	$x_2$	$\dots$	$x_N$	$f(x_{\text{PSO},2})$
⋮					
Candidate solution $p_{\text{PSO}}$	$x_1$	$x_2$	$\dots$	$x_N$	$f(x_{\text{PSO},p_{\text{PSO}}})$

Figure 3: Candidate solution of the internal PSO algorithm.

Step 2 (compute the fitness function value). The fitness function value  $\text{fitness}_j$  of the external RGA approach is the best objective function value  $f(\mathbf{x}_{\text{PSO}}^*)$  obtained from the best solution  $\mathbf{x}_{\text{PSO}}^*$  of each internal PSO algorithm execution, as follows:

$$\text{fitness}_j = f(\mathbf{x}_{\text{PSO}}^*), \quad j = 1, 2, \dots, p_{\text{RGA}}. \quad (3.1)$$

Candidate solution  $j$  of the external RGA approach is incorporated into the internal PSO algorithm, and a CGO problem is then solved using the internal PSO algorithm, which is executed as follows.

#### Internal PSO Algorithm

Step (1) (create an initial particle swarm). An initial particle swarm is created based on the  $p_{\text{PSO}}$  from  $[x_n^l, x_n^u]$  of a CGO problem. A particle represents a candidate solution of a CGO problem, as shown in Figure 3.

Step (2) (calculate the objective function value). According to (2.14), the pseudo-objective function value of the internal PSO algorithm is defined by

$$f_{\text{pseudo},j} = f(\mathbf{x}_{\text{PSO},j}) + \left\{ \rho \times \sum_{m=1}^M \{\max[0, g_m(\mathbf{x}_{\text{PSO},j})]\}^2 \right\}, \quad j = 1, 2, \dots, p_{\text{PSO}}. \quad (3.2)$$

Step (3) (update the particle velocity and position). The particle position and velocity can be updated using (2.6) and (2.10), respectively.

Step (4) (implement a mutation operation). The standard PSO algorithm lacks evolution operations of GAs such as crossover and mutation. To maintain the diversity of particles, this work uses the multi-nonuniform mutation operator defined by (2.4).

Step (5) (perform an elitist strategy). A new particle swarm is created from internal step (3). Notably,  $f(\mathbf{x}_{\text{PSO},j})$  of a candidate solution  $j$  (particle  $j$ ) in the particle swarm

is evaluated. Here, a pairwise comparison is made between the  $f(\mathbf{x}_{\text{PSO},j})$  value of candidate solutions in the new and current particle swarms. A situation in which the candidate solution  $j$  ( $j = 1, 2, \dots, p_{\text{PSO}}$ ) in the new particle swarm is superior to candidate solution  $j$  in the current particle swarm implies that the strong candidate solution  $j$  in the new particle swarm replaces the candidate solution  $j$  in the current particle swarm. The elitist strategy guarantees that the best candidate solution is always preserved in the next generation. The current particle swarm is updated to the particle swarm of the next generation.

Internal steps (2) to (5) are repeated until the  $g_{\max, \text{PSO}}$  value of the internal PSO algorithm is satisfied.

*End*

*Step 3* (implement selection operation). The parents in a crossover pool are selected using (2.1).

*Step 4* (perform crossover operation). In GAs, the crossover operation performs a global search. Thus, the crossover probability  $p_c$  usually exceeds 0.5. Additionally, candidate solutions are created using (2.3).

*Step 5* (conduct mutation operation). In GAs, the mutation operation implements a local search. Additionally, a solution space is exploited using (2.4).

*Step 6* (implement an elitist strategy). This work updates the population using an elitist strategy. A situation in which the fitness<sub>j</sub> of candidate solution  $j$  in the new population is larger than that in the current population suggests that the weak candidate solution  $j$  is replaced. Additionally, a situation in which the fitness<sub>j</sub> of candidate solution  $j$  in the new population is equal to or worse than that in the current population implies that the candidate solution  $j$  in the current population survives. In addition to maintaining the strong candidate solutions, this strategy eliminates weak candidate solutions.

External Steps 2 to 6 are repeated until the  $g_{\max, \text{RGA}}$  value of the external RGA approach is met.

### 3.2. AIA-PSO Algorithm

Figure 4 shows the pseudocode of the proposed AIA-PSO algorithm, in which the external AIA approach is used to optimize the parameter settings of the internal PSO algorithm and the PSO algorithm is used to solve CGO problems.

*External AIA*

*Step 1* (initialize the parameter settings). Several parameters must be predetermined. These include  $rs$  and the threshold for **Ab-Ab** recognition  $p_{rt}$ , as well as the lower and upper boundaries of these parameters  $c_1$ ,  $c_2$ ,  $\chi$ ,  $\rho$ , and  $p_{m, \text{PSO}}$ . Figure 5 shows the **Ab** and **Ag** representation.

*Step 2 (evaluate the **Ab-Ag** affinity).*

*Internal PSO Algorithm*

The external AIA approach offers parameter settings  $c_1, c_2, \chi, \rho$ , and  $p_{m,PSO}$  for the internal PSO algorithm, subsequently leading to the implementation of internal steps (1)–(5) of the PSO algorithm. The PSO algorithm returns the best fitness value of PSO  $f(\mathbf{x}_{PSO}^*)$  to the external AIA approach.

Step (1) (create an initial particle swarm). An initial particle swarm is created based on  $\mathbf{ps}_{PSO}$  from  $[x_n^l, x_n^u]$  of a CGO problem. A particle represents a candidate solution of a CGO problem.

Step (2) (calculate the objective function value). Equation (3.2) is used as the pseudo-objective function value of the internal PSO algorithm.

Step (3) (update the particle velocity and position). Equations (2.6) and (2.10) can be used to update the particle position and velocity.

Step (4) (implement a mutation operation). The diversity of the particle swarm is increased using (2.4).

Step (5) (perform an elitist strategy). A new particle swarm (population) is generated from internal step (3). Notably,  $f(\mathbf{x}_{PSO,j})$  of a candidate solution  $j$  (particle  $j$ ) in the particle swarm is evaluated. Here, a pairwise comparison is made between the  $f(\mathbf{x}_{PSO,j})$  value of candidate solutions in the new and current particle swarms. The elitist strategy guarantees that the best candidate solution is always preserved in the next generation. The current particle swarm is updated to the particle swarm of the next generation.

Internal steps (2) to (5) are repeated until the  $g_{max,PSO}$  value of the internal PSO algorithm is satisfied.

*End*

Consistent with the **Ab-Ag** affinity metaphor, an **Ab-Ag** affinity is determined using (3.3), as follows:

$$\max(\text{affinity}_j) = -1 \times f(\mathbf{x}_{PSO}^*) \quad j = 1, 2, \dots, rs. \quad (3.3)$$

Following the evaluation of the **Ab-Ag** affinities of **Abs**s in the current **Ab** repertoire, the **Ab** with the highest **Ab-Ag** affinity (**Ab** $^*$ ) is chosen to undergo clonal selection operation in external Step 3.

*Step 3 (perform clonal selection operation).* To control the number of antigen-specific **Abs**s, (2.11) is used.

*Step 4 (implement **Ab-Ag** affinity maturation).* The intermediate **Ab** repertoire that is created in external Step 3 is divided into two subsets. These **Abs**s undergo somatic hypermutation operation by using (2.12) when the random number is 0.5 or less. Notably, these **Abs**s suffer receptor editing operation using (2.13) when the random number exceeds 0.5.

```

Procedure external AIA
begin
     $g_{AIA} \leftarrow 0$ 
    Step 1: Initialize the parameter settings
    while  $g_{AIA} < g_{max,AIA}$  do
        Step 2: Evaluate the Ab-Ag affinity  $\rightarrow [X | C_1 | C_2 | \rho | p_{m,PSO}]$ 
         $Ab^* \leftarrow max(affinity_j), j = 1, 2, \dots, rs \quad \boxed{-1 \times x_{PSO}} \leftarrow$ 
        Step 3: Perform clonal selection operation
        for each  $Ab_j, j = 1, 2, \dots, rs$  do
            if  $p_{rj} \geq p_{rt}$  then
                promote (clone)
            else
                suppress
            endif
        endfor
        Step 4: Implement affinity maturation
        for each promoted  $Ab_j$  do
            if rand ()  $\leq 0.5$  do
                somatic hypermutation
            else
                receptor editing
            endif
        endfor
        Step 5: Introduce diverse Abs
        Step 6: Update an Abrepertoire
         $g_{AIA} \leftarrow g_{AIA} + 1$ 
    endwhile
end
end

```

```

Procedure internal PSO Algorithm
gpso  $\leftarrow 0$ 
Step(1) Create an initial particle swarm
(a) parameter settings from RGA
(b) generate initial particle swarm
while executing time  $\leq$  the predefined fixed total time do
    Step(2) Calculate the objective function value
    Step(3) Update the particle velocity and position
    For each candidate particle  $j, j = 1, 2, \dots, ps_{PSO}$  do
        if rand ()  $\leq p_{m, PSO}$  then
            Step(4) Implement a mutation operation
        endif
    Step(5) Perform an elitist strategy
    endfor
    gpso  $\leftarrow gpso + 1$ 
end
end

```

Figure 4: The pseudocode of the AIA-PSO algorithm.

**Step 5 (introduce diverse **Abs**).** Based on the bone marrow operation, diverse **Abs** are created to recruit the **Abs** suppressed in external Step 3.

**Step 6 (update an **Ab** repertoire).** A new **Ab** repertoire is generated from external Steps 3–5. The **Ab-Ag** affinities of the **Abs** in the generated **Ab** repertoire are evaluated. This work presents a strategy for updating the **Ab** repertoire. A situation in which the **Ab-Ag** affinity of **Ab**  $j$  in the new **Ab** repertoire exceeds that in the current **Ab** repertoire implies that a strong **Ab** in the new **Ab** repertoire replaces the weak **Ab** in the current **Ab** repertoire. Additionally, a situation in which the **Ab-Ag** affinity of **Ab**  $j$  in the new **Ab** repertoire equals to or is worse than that in the current **Ab** repertoire implies that the **Ab**  $j$  in the current **Ab** repertoire survives. In addition to maintaining the strong **Abs**, this strategy eliminates nonfunctional **Abs**.

External Steps 2–6 are repeated until the termination criterion  $g_{max,AIA}$  is satisfied.

## 4. Results

The 13 CGO problems were taken from other studies [1, 20, 21, 23, 34]. The set of CGO problems comprises six benchmark NLP problems (TPs 1–4 and 12–13), and seven GPP problems, in which TP 5 (alkylation process design in chemical engineering), TP 6 (optimal reactor design), TP 12 (a tension/compression string design problem), and TP 13 (a pressure vessel design problem) are constrained engineering problems, were used to evaluate the performances of the proposed RGA-PSO and AIA-PSO algorithms. In the appendix, the objective function, constraints, boundary conditions of decision variables, and known global

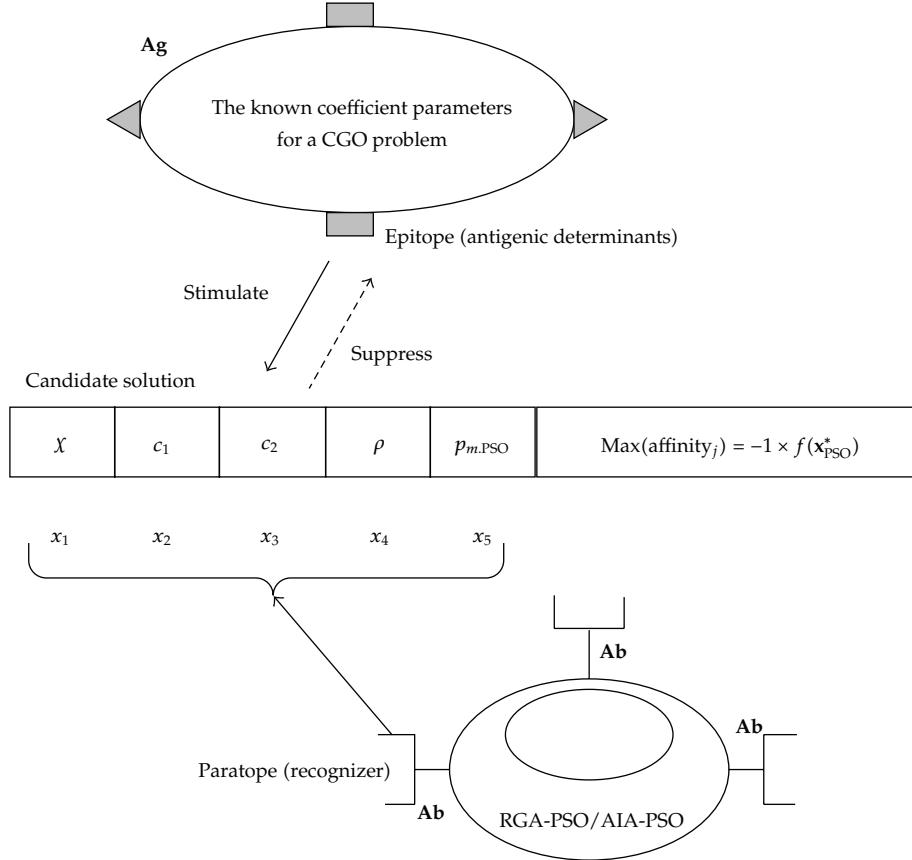


Figure 5: Ag and Ab representation.

optimum for TPs 1–11 are described and the problem characteristics of TPs 5, 12, and 13 are further detailed.

The proposed RGA-PSO and AIA-PSO algorithms were coded in MATLAB software and executed on a Pentium D 3.0 (GHz) personal computer. Fifty independent runs were conducted to solve each test problem (TP). Numerical results were summarized, including the best, median, mean, and worst results, as well as the standard deviation (S.D.) of objective function values obtained using RGA-PSO and AIA-PSO solutions, mean computational CPU times (MCCTs), and mean absolute percentage error MAPE, as defined by

$$\text{MAPE} = \frac{\sum_{s=1}^{50} |(f(\mathbf{x}^*) - f(\mathbf{x}_{\text{stochastic}}^s)) / f(\mathbf{x}^*)|}{50} \times 100\%, \quad s = 1, 2, \dots, 50, \quad (4.1)$$

where  $f(\mathbf{x}^*)$  = value of the known global solution,  $f(\mathbf{x}_{\text{stochastic}}^s)$  = values obtained from solutions of stochastic global optimization approaches (e.g., RGA-PSO and AIA-PSO algorithms).

Table 1 lists the parameter settings for the RGA-PSO and AIA-PSO algorithms, as shown in Table 1.

**Table 1:** The parameter settings for the RGA-PSO and AIA-PSO algorithms.

Methods	Parameter settings	Search space
The external RGA	$p_c = 1$ $p_{m,RGA} = 0.15$ $p_{sRGA} = 10$ $g_{max,RGA} = 3$ $p_{rt} = 0.9$ $rs = 10$ $g_{max,AIA} = 3$ $p_{sPSO} = 100$	$[x^l, x^u] = [0.1, 1]$ $[c_1^l, c_1^u] = [0.1, 2]$ $[c_2^l, c_2^u] = [0.1, 2]$ $[\rho^l, \rho^u] = [1 \times 10^9, 1 \times 10^{11}]$ $[p_{m,PSO}^l, p_{m,PSO}^u] = [0.1, 0.5]$
The external AIA		
The internal PSO algorithm	$g_{max,PSO} = 3500$ for TPs 1–4 $g_{max,PSO} = 3000$ for TPs 5–13	$[x_n^l, x_n^u]$ for a CGO problem

$[x^l, x^u]$ : the lower and upper boundaries of parameter  $x$ .

$[c_1^l, c_1^u]$ : the lower and upper boundaries of parameter  $c_1$ .

$[c_2^l, c_2^u]$ : the lower and upper boundaries of parameter  $c_2$ .

$[\rho^l, \rho^u]$ : the lower and upper boundaries of parameter  $\rho$ .

$[p_{m,PSO}^l, p_{m,PSO}^u]$ : the lower and upper boundaries of  $p_{m,PSO}$  for the internal PSO algorithm  $p_{m,PSO}$ .

#### 4.1. Comparison of the Results Obtained Using the RGA-PSO and AIA-PSO Algorithms

Table 2 summarizes the numerical results obtained using the proposed RGA-PSO and AIA-PSO algorithms for TPs 1–13. Numerical results indicate that the RGA-PSO and the AIA-PSO algorithms can obtain the global minimum solution to TPs 1–11, since each MAPE% is small. Moreover, the best, median, worst, and S.D. of objective function values obtained using the RGA-PSO and AIA-PSO solutions are identical for TPs 1, 2, 3, 4, 6, 7, 8, 9, and 11. Furthermore, the worst values obtained using the AIA-PSO algorithm for TPs 5 and 13 are smaller than those obtained using the RGA-PSO algorithm. Additionally, *t*-test is performed for each TP, indicating that the mean values obtained using the RGA-PSO and AIA-PSO algorithms are statistically significant for TPs 5, 10, 12, and 13, since *P* value is smaller than a significant level 0.05. Based on the results of *t*-test, the AIA-PSO algorithm yields better mean values than the RGA-PSO algorithm for TPs 5, 12, and 13, and the AIA-PSO algorithm yields worse mean value than the RGA-PSO algorithm for TP 10.

Tables 3 and 4 list the best solutions obtained using the RGA-PSO and AIA-PSO algorithms from TPs 1–13, respectively, indicating that each constraint is satisfied (i.e., the violation of each constraint is accurate to at least five decimal places) for every TP. Tables 5 and 6 list the best parameter settings of the internal PSO algorithm obtained using the external RGA and AIA approaches, respectively.

#### 4.2. Comparison of the Results for the Proposed RGA-PSO and AIA-PSO Algorithms with Those Obtained Using the Published Individual GA and AIA Approaches and Hybrid Algorithms

Table 7 compares the numerical results of the proposed RGA-PSO and AIA-PSO algorithms with those obtained using published individual GA and AIA approaches for TPs 1–4. In this table, GA-1 is a GA with a penalty function methods, as used by Michalewicz [20]. Notably,

Table 2: Numerical results of the proposed RGA-PSO and AIA-PSO algorithms for TPs 1-13.

TP number	Global optimum	Methods	Best	Mean	Median	Worst	MAPE %	S.D.	MCCT (sec)	P value
1	24.306	The proposed RGA-PSO	24.323	24.568	24.521	24.831	1.078	0.13	432.39	0.643
		The proposed AIA-PSO	24.358	24.579	24.564	24.809	1.125	0.11	417.63	
2	-30665.539	The proposed RGA-PSO	-30665.539	-30665.539	-30665.539	-30665.534	1.34E - 06	7.36E - 04	349.55	0.478
		The proposed AIA-PSO	-30665.539	-30665.539	-30665.539	-30665.539	9.95E - 07	3.81E - 05	349.76	
3	680.630	The proposed RGA-PSO	680.632	680.640	680.639	680.658	1.46E - 03	5.55E - 03	398.72	0.839
		The proposed AIA-PSO	680.633	680.640	680.640	680.657	1.50E - 03	0.01	398.84	
4	-15	The proposed RGA-PSO	-15	-15	-15	-15	1.27E - 06	1.27E - 06	466.00	0.294
		The proposed AIA-PSO	-15	-15	-15	-15	1.20E - 10	2.64E - 11	450.28	
5	1227.1978	The proposed RGA-PSO	1227.2321	1228.8471	1228.3935	1231.6003	1.34E - 01	1.27	481.86	0.002*
		The proposed AIA-PSO	1227.1598	1228.1800	1227.9762	1229.6590	8.02E - 02	0.71	461.30	
6	3.9511	The proposed RGA-PSO	3.9521	3.9577	3.9566	3.9722	1.67E - 01	5E - 03	439.16	0.682
		The proposed AIA-PSO	3.9516	3.9581	3.9569	3.9735	1.77E - 01	4.79E - 03	438.55	
7	-5.7398	The proposed RGA-PSO	-5.7398	-5.7398	-5.7398	-5.7398	2.54E - 04	1.13E - 05	325.73	0.358

Table 2: Continued.

TP number	Global optimum	Methods	Best	Mean	Median	Worst	MAPE %	S.D.	MCCT (sec)	P value
8	-83.254	The proposed AIA-PSO	-5.7398	-5.7398	-5.7398	-5.7398	2.42E - 04	7.74E - 06	329.12	
		The proposed RGA-PSO	-83.2497	-83.2497	-83.2497	-83.2497	5.13E - 03	1.26E - 06	211.91	0.291
9	-6.0482	The proposed AIA-PSO	-83.2497	-83.2497	-83.2497	-83.2497	5.13E - 03	6.85E - 07	213.01	
		The proposed RGA-PSO	-6.0441	-5.9883	-5.9968	-5.8051	0.991	0.05	440.72	0.577
10	6300	The proposed AIA-PSO	-6.0467	-5.9828	-5.9979	-5.8274	1.082	0.05	441.42	
		The proposed RGA-PSO	6299.8374	6299.8412	6299.8419	6299.8428	2.52E - 03	1.42E - 03	224.13	0.001*
11	10122.6964	The proposed AIA-PSO	6299.8395	6299.8420	6299.8423	6299.8425	2.51E - 03	6.83E - 04	223.91	
		The proposed RGA-PSO	10122.4732	10122.4925	10122.4925	10122.6444	2.01E - 03	2.26E - 02	340.89	0.84
12	—	The proposed AIA-PSO	10122.4852	10122.4920	10122.4927	10122.4931	2.02E - 03	1.85E - 03	338.37	
		The proposed RGA-PSO	0.012692	0.012724	0.012721	0.012784	—	1.46E - 05	293.67	0.014*
13	—	The proposed AIA-PSO	0.012667	0.012715	0.012719	0.012778	—	2.00E - 05	296.15	
		The proposed RGA-PSO	5885.3018	5895.0381	5885.3326	6005.4351	—	24.33	291.17	0.019*
	The proposed AIA-PSO	5885.3249	5886.5426	5885.3323	5906.7404	—	4.54	292.16		

(The “—” denotes unavailable information, and \*represents that the mean values obtained using the RGA-PSO and AIA-PSO algorithms are statistically different.)

**Table 3:** The best solutions obtained using the RGA-PSO algorithm from TPs 1–13.

TP number	$f(\mathbf{x}_{\text{RGA-PSO}}^*)$	$\mathbf{x}_{\text{RGA-PSO}}^*$
1	24.323	$\mathbf{x}_{\text{RGA-PSO}}^* = (2.16727760, 2.37789634, 8.78162804, 5.12372885, 0.97991270, 1.39940993, 1.31127142, 9.81945011, 8.30004549, 8.45891329)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (-0.000171 \leq 0, -0.003109 \leq 0, -0.000027 \leq 0, -0.000123 \leq 0, -0.001371 \leq 0, -0.002101 \leq 0, -6.245957 \leq 0, -47.846302 \leq 0)$
2	-30665.539	$\mathbf{x}_{\text{RGA-PSO}}^* = (78, 33, 29.99525450, 45, 36.77581373)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (-92.000000 \leq 0, 2.24E - 07 \leq 0, -8.840500 \leq 0, -11.159500 \leq 0, 4.28E - 07 \leq 0, -5.000000 \leq 0)$
3	680.632	$\mathbf{x}_{\text{RGA-PSO}}^* = (2.33860239, 1.95126191, -0.45483579, 4.36300325, -0.62317747, 1.02938443, 1.59588410)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (-7.76E - 07 \leq 0, -252.7211 \leq 0, -144.8140 \leq 0, -6.15E - 05 \leq 0)$
4	-15	$\mathbf{x}_{\text{RGA-PSO}}^* = (1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 1)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (0 \leq 0, 0 \leq 0, 0 \leq 0, -5 \leq 0, -5 \leq 0, 0 \leq 0, 0 \leq 0, 0 \leq 0)$
5	1227.1139	$\mathbf{x}_{\text{RGA-PSO}}^* = (1697.13793410, 54.17332240, 3030.44600072, 90.18199040, 94.99999913, 10.42097385, 153.53771370)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (0.999978 \leq 1, 0.980114 \leq 1, 1.000005 \leq 1, 0.980097 \leq 1, 0.990565 \leq 1, 1.000005 \leq 1, 1.000000 \leq 1, 0.976701 \leq 1, 1.000003 \leq 1, 0.459311 \leq 1, 0.387432 \leq 1, 0.981997 \leq 1, 0.980364 \leq 1, -8.241926 \leq 1)$
6	3.9521	$\mathbf{x}_{\text{RGA-PSO}}^* = (6.444100620, 2.243029250, 0.642672939, 0.582321363, 5.940008650, 5.531235784, 1.018087316, 0.403665649)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (1.000000 \leq 1, 1.000000 \leq 1, 0.999996 \leq 1, 0.99986 \leq 1)$
7	-5.7398	$\mathbf{x}_{\text{RGA-PSO}}^* = (8.12997229, 0.61463971, 0.56407162, 5.63623069)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (1.000000 \leq 1, 1.000000 \leq 1)$
8	-83.2497	$\mathbf{x}_{\text{RGA-PSO}}^* = (88.35595404, 7.67259607, 1.31787691)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (1.000000 \leq 1)$
9	-6.0441	$\mathbf{x}_{\text{RGA-PSO}}^* = (6.40497368, 0.64284563, 1.02766984, 5.94729224, 2.21044814, 0.59816471, 0.42450835, 5.54339987)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (0.999997 \leq 1, 0.999959 \leq 1, 0.999930 \leq 1, 0.999975 \leq 1)$
10	6299.8374	$\mathbf{x}_{\text{RGA-PSO}}^* = (108.66882633, 85.10837983, 204.35894362)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (1.000002 \leq 1)$
11	10122.4732	$\mathbf{x}_{\text{RGA-PSO}}^* = (78, 33, 29.99564864, 45, 36.77547104)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (-0.309991 \leq 1, 1.000004 \leq 1, -0.021379 \leq 1, 0.621403 \leq 1, 1.000002 \leq 1, 0.681516 \leq 1)$
12	0.012692	$\mathbf{x}_{\text{RGA-PSO}}^* = (0.050849843, 0.336663305, 12.579603478)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (-0.000143 \leq 0, -0.000469 \leq 0, -4.009021 \leq 0, -0.741658 \leq 0)$
13	5885.3018	$\mathbf{x}_{\text{RGA-PSO}}^* = (0.77816852, 0.38464913, 40.31961883, 199.99999988)$ $g_m(\mathbf{x}_{\text{RGA-PSO}}^*) = (1.23E - 07 \leq 0, 3.36E - 08 \leq 0, -0.006916 \leq 0, -40.00000012 \leq 0)$

**Table 4:** The best solutions obtained using the AIA-PSO algorithm from TPs 1–13.

TP number	$f(\mathbf{x}_{\text{AIA-PSO}}^*)$	$\mathbf{x}_{\text{AIA-PSO}}^*$
1	24.358	$\mathbf{x}_{\text{AIA-PSO}}^* = (2.18024296, 2.35746157, 8.75670935, 5.11326109, 1.03976363, 1.54784227, 1.32994030, 9.83127443, 8.27618717, 8.32717779)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (-0.000071 \leq 0, -0.003699 \leq 0, -0.000440 \leq 0, -0.684025 \leq 0, -0.000086 \leq 0, -0.001024 \leq 0, -5.973866 \leq 0, -47.371958 \leq 0)$
2	-30665.539	$\mathbf{x}_{\text{AIA-PSO}}^* = (78, 33, 29.99525527, 45, 36.77581286)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (-92.000000 \leq 0, 5.57E - 08 \leq 0, -8.840500 \leq 0, -11.159500 \leq 0, 2.76E - 07 \leq 0, -5.000000 \leq 0)$
3	680.633	$\mathbf{x}_{\text{AIA-PSO}}^* = (2.32925164, 1.95481519, -0.47307614, 4.35691576, -0.62313420, 1.05236194, 1.59750978)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (-4.26E - 06 \leq 0, -252.612733 \leq 0, -144.741194 \leq 0, -1.00E - 05 \leq 0)$
4	-15	$\mathbf{x}_{\text{AIA-PSO}}^* = (1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 1)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (0 \leq 0, 0 \leq 0, 0 \leq 0, -5 \leq 0, -5 \leq 0, -5 \leq 0, 0 \leq 0, 0 \leq 0)$
5	1227.1598	$\mathbf{x}_{\text{AIA-PSO}}^* = (1697.91645044, 53.78132825, 3031.07989059, 90.12651301, 94.99999995, 10.48103708, 153.53594861)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (1.000000 \leq 1, 0.980100 \leq 1, 1.000002 \leq 1, 0.980099 \leq 1, 0.990562 \leq 1, 1.000001 \leq 1, 1.000002 \leq 1, 0.976715 \leq 1, 1.000001 \leq 1, 0.459425 \leq 1, 0.387515 \leq 1, 0.981856 \leq 1, 0.987245 \leq 1, -8.302533 \leq 1)$
6	3.9516	$\mathbf{x}_{\text{AIA-PSO}}^* = (6.44373647, 2.23335111, 0.68233303, 0.60026617, 5.93745119, 5.53146186, 1.01862958, 0.40673661)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (0.999999 \leq 1, 0.999999 \leq 1, 0.999994 \leq 1, 0.999985 \leq 1)$
7	-5.7398	$\mathbf{x}_{\text{AIA-PSO}}^* = (8.13042985, 0.61758136, 0.56393603, 5.63618191)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (0.999999 \leq 1, 0.999998 \leq 1)$
8	-83.2497	$\mathbf{x}_{\text{AIA-PSO}}^* = (88.35635930, 7.67202113, 1.31765768)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (3.00E - 18 \leq 1)$
9	-6.0467	$\mathbf{x}_{\text{AIA-PSO}}^* = (6.46393173, 0.67575021, 1.01188804, 5.94072071, 2.24639462, 0.60683404, 0.39677469, 5.52596342)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (0.999980 \leq 1, 0.999999 \leq 1, 0.998739 \leq 1, 0.999928 \leq 1)$
10	6299.8395	$\mathbf{x}_{\text{AIA-PSO}}^* = (108.97708780, 85.02248757, 204.45439729)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (1.000003 \leq 1)$
11	10122.4852	$\mathbf{x}_{\text{AIA-PSO}}^* = (78, 33, 29.99571251, 45, 36.77534593)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (-0.309991 \leq 1, 1.000001 \leq 1, -0.021380 \leq 1, 0.621403 \leq 1, 1.000001 \leq 1, 0.681517 \leq 1)$
12	0.012667	$\mathbf{x}_{\text{AIA-PSO}}^* = (0.05164232, 0.35558085, 11.35742676)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (-8.83E - 05 \leq 0, -3.04E - 05 \leq 0, -4.050924 \leq 0, -0.728518 \leq 0)$
13	5885.3310	$\mathbf{x}_{\text{AIA-PSO}}^* = (0.77816843, 0.38464909, 40.31961929, 199.99999330)$ $g_m(\mathbf{x}_{\text{AIA-PSO}}^*) = (2.22E - 07 \leq 0, 7.80E - 08 \leq 0, -0.006015 \leq 0, -40.000007 \leq 0)$

**Table 5:** The best parameter settings of the best solution obtained using the RGA-PSO algorithm from TPs 1–13.

TP number	$\chi$	$c_1$	$c_2$	$\rho$	$p_{m,PSO}$
1	0.73759998	1.52939998	1.72753568	28709280994	0.3750
2	0.67057537	0.45010388	2	1000000000	0.2709
3	0.75493696	0.36925226	1.91198475	54900420223	0.3856
4	0.45438391	1.45998477	1.25508289	1000000000	0.1
5	1	0.77483233	2	1000000000	0.5
6	0.70600559	0.82360083	0.91946627	26622414511	0.1619
7	0.74584341	0.95474855	1.17537957	86557786169	0.1958
8	1	0.69725160	1.53028620	1000000000	0.1675
9	0.41638718	0.46594542	1.97807798	12976330236	0.3686
10	0.25378610	0.59619170	0.83891277	1000000000	0.1
11	0.48183123	2	2	1000000000	0.1
12	0.76190087	0.1	1.16855713	100000000000	0.5
13	0.71783704	1.39420750	1.33590124	29071187026	0.2098

**Table 6:** The best parameter settings of the best solution obtained using the AIA-PSO algorithm from TPs 1–13.

TP number	$\chi$	$c_1$	$c_2$	$\rho$	$p_{m,PSO}$
1	0.44896780	0.71709211	1.93302154	72140307110	0.4058
2	1	1.36225259	1.97905466	182831007	0.5
3	0.46599492	0.90346435	1.89697456	69125199709	0.2613
4	0.98124982	0.27882671	0.87437226	85199047430	0.1
5	0.99082484	0.1	1.11371788	4231387044	0.5
6	0.82869043	0.88773247	2	1387448505	0.5
7	0.87571243	1.89936723	0.74306310	94752095153	0.2194
8	0.93583844	1.53906226	1.30374874	22520728225	0.1798
9	1	0.22556712	1.52263349	67578847151	0.4507
10	1	1.93003999	0.1	1351461763	0.5
11	1	1.51209364	1.63826995	1811017789	0.5
12	0.52068067	0.1	2	81914376144	0.1
13	0.82395890	1.60107152	0.93611204	17767111886	0.1813

GA-2 represents a GA with a penalty function, but without any penalty parameter, as used by Deb [21]. Also, GA-3 is an RGA with a static penalty function, as developed by Wu and Chung [9]. Notably, AIA-1 is an AIA method called CLONALG, as proposed by Cruz-Cortés et al. [22]. Finally, AIA-2 is an AIA approach based on an adaptive penalty function, as developed by Wu [10]. The numerical results of GA-1, GA-2, and AIA-1 methods for solving TPs 1–4 were collected from the published literature [20–22]. Furthermore, the GA-1, GA-2, and AIA-1 approaches were executed under 350,000 objective function evaluations. To fairly compare the performances of the proposed hybrid CI algorithms and the individual GA and AIA approaches, the GA-3, AIA-2, the internal PSO algorithm of RGA-PSO method, and the internal PSO algorithm of AIA-PSO method were independently executed 50 times under 350,000 objective function evaluations for solving TPs 1–4.

For solving TP 1, the median values obtained using the RGA-PSO and AIA-PSO algorithms are smaller than those obtained using the GA-1, GA-3, and AIA-2 approaches, and

**Table 7:** Comparison of the results of the proposed RGA-PSO and AIA-PSO algorithms and those of the published individual GA and AIA approaches for TPs 1–4.

TP number	Global optimum	Methods	Best	Mean	Median	Worst	MAPE%
1	24.306	GA-1 [20]	24.690	—	29.258	36.060	—
		GA-2 [21]	24.372	—	24.409	25.075	—
		GA-3 [9]	24.935	27.314	27.194	33.160	12.377
		AIA-1 [22]	24.506	25.417	—	26.422	—
		AIA-2 [10]	24.377	24.669	24.663	24.988	1.495
		The proposed RGA-PSO	24.323	24.568	24.521	24.831	1.078
		The proposed AIA-PSO	24.358	24.579	24.564	24.809	1.125
2	−30665.539	GA-1 [20]	—	—	—	—	—
		GA-2 [21]	—	—	—	—	—
		GA-3 [9]	−30665.526	−30662.922	−30664.709	−30632.445	8.53E − 03
		AIA-1 [22]	−30665.539	−30665.539	—	−30665.539	—
		AIA-2 [10]	−30665.539	−30665.526	−30665.527	−30665.506	4.20E − 05
		The proposed RGA-PSO	−30665.539	−30665.539	−30665.539	−30665.534	1.34E − 06
		The proposed AIA-PSO	−30665.539	−30665.539	−30665.539	−30665.539	9.95E − 07
3	680.630	GA-1 [20]	680.642	—	680.718	680.955	—
		GA-2 [21]	680.634	—	680.642	680.651	—
		GA-3 [9]	680.641	680.815	680.768	681.395	2.72E − 02
		AIA-1 [22]	680.631	680.652	—	680.697	—
		AIA-2 [10]	680.634	680.653	680.650	680.681	3.45E − 03
		The proposed RGA-PSO	680.632	680.640	680.639	680.658	1.46E − 03
		The proposed AIA-PSO	680.633	680.640	680.640	680.657	1.50E − 03
4	−15	GA-1 [20]	−15	−15	−15	−15	—
		GA-2 [21]	—	—	—	—	—
		GA-3 [9]	−13.885	−12.331	−12.267	−10.467	17.795
		AIA-1 [22]	−14.987	−14.726	—	−12.917	—
		AIA-2 [10]	−14.998	−14.992	−14.992	−14.988	5.08E − 02
		The proposed RGA-PSO	−15	−15	−15	−15	1.27E − 06
		The proposed AIA-PSO	−15	−15	−15	−15	1.20E − 10

(The “—” denotes unavailable information.)

the worst values obtained using RGA-PSO and AIA-PSO algorithms are smaller than those obtained using the GA-1, GA-2, GA-3, AIA-1, and AIA-2 approaches. For solving TP 2, the median and worst values obtained using the RGA-PSO and AIA-PSO algorithms are smaller than those obtained using the GA-3 method. For solving TP 3, the median and worst values

**Table 8:** Results of the *t*-test for TPs 1–4.

TP number	GA-3 versus AIA-2	GA-3 versus RGA-PSO	GA-3 versus AIA-PSO	AIA-2 versus RGA-PSO	AIA-2 versus AIA-PSO	RGA-PSO versus AIA-PSO
1	0.000*	0.000*	0.000*	0.000*	0.001*	0.643
2	0.003*	0.003*	0.003*	0.000*	0.000*	0.478
3	0.000*	0.000*	0.000*	0.000*	0.000*	0.839
4	0.000*	0.000*	0.000*	0.000*	0.000*	0.294

\*Represents that the mean values obtained using two algorithms are statistically different.)

obtained using the RGA-PSO and AIA-PSO algorithms are smaller than those obtained using the GA-1 and GA-3 approaches. For solving TP 4, the median and worst values obtained using the RGA-PSO and AIA-PSO algorithms are smaller than those obtained using the GA-3 method, and the worst values obtained using the RGA-PSO and AIA-PSO algorithms are smaller than those obtained using the AIA-1 approach. Moreover, the GA-3 method obtained the worst MAPE% for TP 1 and TP 4. Table 8 lists the results of the *t*-test for the GA-3, AIA-2, RGA-PSO, and AIA-PSO methods. This table indicates that the mean values of the RGA-PSO, and AIA-PSO algorithms are not statistically significant, since *P* values are larger than a significant level 0.05, and the mean values between GA-3 versus AIA-2, GA-3 versus RGA-PSO, GA-3 versus AIA-PSO, AIA-2 versus RGA-PSO, and AIA-2 versus AIA-PSO are statistically significant. According to Tables 7 and 8, the mean values obtained using the RGA-PSO and AIA-PSO algorithms are better than those of obtained using the GA-3 and AIA-1 methods for TPs 1–4.

Table 9 compares the numerical results obtained using the proposed RGA-PSO and AIA-PSO algorithms and those obtained using AIA-2 and GA-3 for solving TPs 5–13. The AIA-2, GA-3, the internal PSO algorithm of the RGA-PSO approach, and the internal PSO algorithm of AIA-PSO approach were independently executed 50 times under 300,000 objective function evaluations. Table 9 shows that MAPE% obtained using the proposed RGA-PSO and AIA-PSO algorithms is close to 1%, or smaller than 1% for TPs 5–11, indicating that the proposed RGA-PSO and AIA-PSO algorithms can converge to global optimum for TPs 5–11. Moreover, the worst values obtained using the RGA-PSO and AIA-PSO algorithms are significantly smaller than those obtained using the GA-3 method for TPs 5, 6, 11, and 13. Additionally, the worst values obtained using the RGA-PSO and AIA-PSO algorithms are smaller than those obtained using the AIA-2 method for TPs 5, 6, and 13.

Table 10 summarizes the results of the *t*-test for TPs 5–13. According to Tables 9 and 10, the mean values of the RGA-PSO and AIA-PSO algorithms are smaller than those of the GA-3 approach for TPs 5, 6, 7, 8, 9, 10, 11, and 13. Moreover, the mean values obtained using the RGA-PSO and AIA-PSO algorithms are smaller than those of the AIA-2 approach for TPs 6, 7, 8, 10, and 12. Totally, according to Tables 7–10, the performances of the hybrid CI methods are superior to those of individual GA and AIA methods.

The TPs 12 and 13 have been solved by many hybrid algorithms. For instance, Huang et al. [23] presented a coevolutionary differential evolution (CDE) that integrates a coevolution mechanism and a DE approach. Zahara and Kao [24] developed a hybrid Nelder-Mead simplex search method and a PSO algorithm (NM-PSO). Table 11 compares

**Table 9:** Comparison of the numerical results of the proposed RGA-PSO and AIA-PSO algorithms and those of the published individual AIA and RGA for TPs 5–13.

TP number	Global optimum	Methods	Best	Mean	Median	Worst	MAPE%
5 1227.1978	AIA-2 [10]	1227.3191	1228.6097	1227.8216	1239.5205	1.15E – 01	
	GA-3 [9]	1228.5118	1279.3825	1263.2589	1481.3710	4.266	
	The proposed RGA-PSO	1227.2321	1228.8471	1228.3935	1231.6003	1.34E – 01	
	The proposed AIA-PSO	1227.1598	1228.1800	1227.9762	1229.6590	8.02E – 02	
	AIA-2 [10]	3.9518	4.1005	4.0460	4.2997	3.781	
	GA-3 [9]	3.9697	4.1440	4.1358	4.3743	4.881	
6 3.9511	The proposed RGA-PSO	3.9521	3.9577	3.9566	3.9722	1.67E – 01	
	The proposed AIA-PSO	3.9516	3.9581	3.9569	3.9735	1.77E – 01	
	AIA-2 [10]	-5.7398	-5.7398	-5.7398	-5.7396	6.03E – 04	
	GA-3 [9]	-5.7398	-5.7375	-5.7383	-5.7316	3.98E – 02	
	The proposed RGA-PSO	-5.7398	-5.7398	-5.7398	-5.7398	2.54E – 04	
	The proposed AIA-PSO	-5.7398	-5.7398	-5.7398	-5.7398	2.42E – 04	
7 -5.7398	AIA-2 [10]	-83.2504	-83.2499	-83.2499	-83.2496	4.93E – 03	
	GA-3 [9]	-83.2497	-83.2490	-83.2494	-83.2461	6.06E – 03	
	The proposed RGA-PSO	-83.2497	-83.2497	-83.2497	-83.2497	5.13E – 03	
	The proposed AIA-PSO	-83.2497	-83.2497	-83.2497	-83.2497	5.13E – 03	
	AIA-2 [10]	-6.0471	-5.9795	-6.0138	-5.7159	1.136	
	GA-3 [9]	-6.0417	-5.8602	-5.8772	-5.6815	3.108	
8 -83.254	The proposed RGA-PSO	-6.0441	-5.9883	-5.9968	-5.8051	0.991	
	The proposed AIA-PSO	-6.0466	-5.9808	-5.9969	-5.8274	1.082	
	AIA-2 [10]	6299.8232	6299.8388	6299.8403	6299.8494	2.56E – 03	
	GA-3 [9]	6299.8399	6299.8546	6299.8553	6299.9407	2.31E – 03	
	The proposed RGA-PSO	6299.8374	6299.8412	6299.8419	6299.8428	2.52E – 03	
	The proposed AIA-PSO	6299.8395	6299.8420	6299.8423	6299.8425	2.51E – 03	
9 -6.0482	AIA-2 [10]	10122.4343	10122.5014	10122.5023	10122.6227	1.93E – 03	
	GA-3 [9]	10122.4809	10125.2441	10123.9791	10155.1349	2.58E – 04	
	The proposed RGA-PSO	10122.4732	10122.4925	10122.4925	10122.6444	2.01E – 03	
	The proposed AIA-PSO	10122.4852	10122.4920	10122.4927	10122.4931	2.02E – 03	
	AIA-2 [10]	0.012665	0.012703	0.012719	0.012725	—	
	GA-3 [9]	0.012665	0.012739	0.012717	0.013471	—	
10 6300	The proposed RGA-PSO	0.012692	0.012724	0.012721	0.012784	—	
	The proposed AIA-PSO	0.012667	0.012715	0.012778	0.012778	—	
11 10122.6964	AIA-2 [10]	—	—	—	—	—	
	GA-3 [9]	—	—	—	—	—	
12 —	The proposed RGA-PSO	—	—	—	—	—	
	The proposed AIA-PSO	—	—	—	—	—	

**Table 9:** Continued.

TP number	Global optimum	Methods	Best	Mean	Median	Worst	MAPE%
		AIA-2 [10]	5885.5312	5900.4860	5894.7929	6014.2198	—
	—	GA-3 [9]	5885.3076	5943.4714	5897.2904	6289.7314	—
13	—	The proposed RGA-PSO	5885.3018	5895.0381	5885.3326	6005.4351	—
		The proposed AIA-PSO	5885.3310	5886.5426	5885.3323	5906.7404	—

(The “—” denotes unavailable information.)

**Table 10:** Results of  $t$ -test for TPs 5–13.

TP number	GA-3 versus AIA-2	GA-3 versus RGA-PSO	GA-3 versus AIA-PSO	AIA-2 versus RGA-PSO	AIA-2 versus AIA-PSO	RGA-PSO versus AIA-PSO
5	0.000*	0.000*	0.000*	0.485	0.163	0.002*
6	0.112	0.000*	0.000*	0.000*	0.000*	0.682
7	0.000*	0.000*	0.000*	0.000*	0.000*	0.358
8	0.000*	0.000*	0.000*	0.000*	0.000*	0.291
9	0.000*	0.000*	0.000*	0.516	0.814	0.577
10	0.000*	0.000*	0.000*	0.009*	0.001*	0.001*
11	0.000*	0.000*	0.000*	0.069	0.013*	0.884
12	0.049*	0.389	0.178	0.000*	0.007*	0.014*
13	0.003*	0.001*	0.000*	0.240	0.000*	0.019*

\*Represents that the mean values obtained using two algorithms are statistically different.)

**Table 11:** Comparison of the numerical results of the proposed RGA-PSO and AIA-PSO algorithms and those of the published hybrid algorithms for TPs 12–13.

TP number	Methods	Best	Mean	Median	Worst	S.D.
12	CDE [23]	0.0126702	0.012703	—	0.012790	2.7E – 05
	NM-PSO [24]	0.0126302	0.0126314	—	0.012633	8.73E – 07
	The proposed RGA-PSO	0.012692	0.012724	0.012721	0.012784	1.46E – 05
	The proposed AIA-PSO	0.012667	0.012715	0.012719	0.012778	2.00E – 05
13	CDE [23]	6059.7340	6085.2303	—	6371.0455	43.01
	NM-PSO [24]	5930.3137	5946.7901	—	5960.0557	9.16
	The proposed RGA-PSO	5885.3018	5895.0381	5885.3326	6005.4351	24.33
	The proposed AIA-PSO	5885.3310	5886.5426	5885.3323	5906.7404	4.54

the numerical results of the CDE, NM-PSO, RGA-PSO, and AIA-PSO methods for solving TPs 12–13. The table indicates that the best, mean, and worst values obtained using the NM-PSO method are superior to those obtained using the CDE, RGA-PSO, and AIA-PSO approaches for TP 12. Moreover, the best, mean, and worst values obtained using the AIA-PSO algorithm are better than those of the CDE, NM-PSO, and RGA-PSO algorithms.

According to the No Free Lunch theorem [35], if algorithm A outperforms algorithm B on average for one class of problems, then the average performance of the former must be worse than that of the latter over the remaining problems. Therefore, it is unlikely that any unique stochastic global optimization approach exists that performs best for all CGO problems.

### **4.3. Summary of Results**

The proposed RGA-PSO and AIA-PSO algorithms with a penalty function method have the following benefits.

- (1) Parameter manipulation of the internal PSO algorithm is based on the solved CGO problems. Owing to their ability to efficiently solve an UGO problem, the external RGA and AIA approaches are substituted for trial and error to manipulate the parameters ( $\chi$ ,  $c_1$ ,  $c_2$ ,  $\rho$ , and  $p_{m,PSO}$ ).
- (2) Besides obtaining the optimum parameter settings of the internal PSO algorithm, the RGA-PSO and AIA-PSO algorithms can yield a global optimum for a CGO problem.
- (3) In addition to performing better than approaches of some published individual GA and AIA approaches, the proposed RGA-PSO and AIA-PSO algorithms reduce the parametrization for the internal PSO algorithm, despite the RGA-PSO and AIA-PSO algorithms being more complex than individual GA and AIA approaches.

The proposed RGA-PSO and AIA-PSO algorithms have the following limitations.

- (1) The proposed RGA-PSO and AIA-PSO algorithms increase the computational CPU time, as shown in Table 2.
- (2) The proposed RGA-PSO and AIA-PSO algorithms are designed to solve CGO problems with continuous decision variables  $x_n$ . Therefore, the proposed algorithms cannot be applied to manufacturing problems such as job shop scheduling and quadratic assignment problems (combinatorial optimization problems).

## **5. Conclusions**

This work presents novel RGA-PSO and AIA-PSO algorithms. The synergistic power of the RGA with PSO algorithm and the AIA with PSO algorithm is also demonstrated by using 13 CGO problems. Numerical results indicate that, in addition to converging to a global minimum for each test CGO problem, the proposed RGA-PSO and AIA-PSO algorithms obtain the optimum parameter settings of the internal PSO algorithm. Moreover, the numerical results obtained using the RGA-PSO and AIA-PSO algorithms are superior to those obtained using alternative stochastic global optimization methods such as individual GA and AIA approaches. The RGA-PSO and AIA-PSO algorithms are highly promising stochastic global optimization approaches for solving CGO problems.

## Appendices

### A. TP 1 [20, 21]

TP 1 has ten decision variables, eight inequality constraints, and 20 boundary conditions, as follows:

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

$$\text{Subject to } g_1(\mathbf{x}) \equiv -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \leq 0,$$

$$g_2(\mathbf{x}) \equiv 10x_1 - 8x_2 - 17x_7 + 2x_8 \leq 0,$$

$$g_3(\mathbf{x}) \equiv -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leq 0,$$

$$g_4(\mathbf{x}) \equiv 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \leq 0, \quad (\text{A.1})$$

$$g_5(\mathbf{x}) \equiv 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \leq 0,$$

$$g_6(\mathbf{x}) \equiv x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \leq 0,$$

$$g_7(\mathbf{x}) \equiv 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30 \leq 0,$$

$$g_8(\mathbf{x}) \equiv -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \leq 0,$$

$$-10 \leq x_n \leq 10, \quad n = 1, 2, \dots, 10.$$

The global solution to TP 1 is as follows:

$$\begin{aligned} \mathbf{x}^* = & (2.171996, 2.363683, 8.773926, 5.095984, 0.9906548, \\ & 1.430574, 1.321644, 9.828726, 8.280092, 8.375927), \quad (\text{A.2}) \\ f(\mathbf{x}^*) = & 24.306. \end{aligned}$$

### B. TP 2 [21]

TP 2 involves five decision variables, six inequality constraints, and ten boundary conditions, as follows:

$$\text{Minimize } f(\mathbf{x}) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141$$

$$\text{Subject to } g_1(\mathbf{x}) \equiv -85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 + 0.0022053x_3x_5 \leq 0,$$

$$g_2(\mathbf{x}) \equiv -6.665593 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 \leq 0,$$

$$g_3(\mathbf{x}) \equiv 9.48751 - 0.0071371x_2x_5 - 0.0029955x_1x_2 - 0.0021813x_3^2 \leq 0,$$

$$g_4(\mathbf{x}) \equiv -29.48751 + 0.0071371x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2 \leq 0,$$

$$g_5(\mathbf{x}) \equiv 10.669039 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 \leq 0,$$

$$g_6(\mathbf{x}) \equiv -15.699039 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 \leq 0,$$

$$78 \leq x_1 \leq 102, \quad 33 \leq x_2 \leq 45, \quad 27 \leq x_n \leq 45, \quad n = 3, 4, 5. \quad (\text{B.1})$$

The global solution to TP 2 is

$$\begin{aligned} \mathbf{x}^* &= (78.0, 33.0, 29.995256, 45.0, 36.775812), \\ f(\mathbf{x}^*) &= -30665.539. \end{aligned} \quad (\text{B.2})$$

### C. TP 3 [20, 21]

TP 3 has seven decision variables, four inequality constraints, and 14 boundary conditions, as follows:

$$\begin{aligned} \text{Minimize } f(\mathbf{x}) &= (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + 10x_5^6 \\ &\quad + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7 \end{aligned}$$

$$\text{Subject to } g_1(\mathbf{x}) \equiv -127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leq 0,$$

$$g_2(\mathbf{x}) \equiv -282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leq 0, \quad (\text{C.1})$$

$$g_3(\mathbf{x}) \equiv -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leq 0,$$

$$g_4(\mathbf{x}) \equiv 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leq 0,$$

$$-10 \leq x_n \leq 10, \quad n = 1, 2, \dots, 7.$$

The global solution to TP 3 is

$$\begin{aligned}\mathbf{x}^* &= (2.330499, 1.951372, -0.4775414, 4.365726, -0.6244870, 1.038131, 1.594227), \\ f(\mathbf{x}^*) &= 680.630.\end{aligned}\quad (\text{C.2})$$

## D. TP 4 [20, 21]

TP 4 involves 13 decision variables, nine inequality constraints, and 26 boundary conditions, as follows:

$$\begin{aligned}\text{Minimize } \quad f(\mathbf{x}) &= 5 \sum_{n=1}^4 x_n - 5 \sum_{n=1}^4 x_n^2 - \sum_{n=5}^{13} x_n \\ \text{Subject to } \quad g_1(\mathbf{x}) &\equiv 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \leq 0, \\ g_2(\mathbf{x}) &\equiv 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \leq 0, \\ g_3(\mathbf{x}) &\equiv 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \leq 0, \\ g_4(\mathbf{x}) &\equiv -8x_1 + x_{10} \leq 0, \\ g_5(\mathbf{x}) &\equiv -8x_2 + x_{11} \leq 0, \\ g_6(\mathbf{x}) &\equiv -8x_3 + x_{12} \leq 0, \\ g_7(\mathbf{x}) &\equiv -2x_4 - x_5 + x_{10} \leq 0, \\ g_8(\mathbf{x}) &\equiv -2x_6 - x_7 + x_{11} \leq 0, \\ g_9(\mathbf{x}) &\equiv -2x_8 - x_9 + x_{12} \leq 0, \\ 0 \leq x_n &\leq 1, \quad n = 1, 2, \dots, 9, \\ 0 \leq x_n &\leq 100, \quad n = 10, 11, 12, \\ 0 \leq x_{13} &\leq 1.\end{aligned}\quad (\text{D.1})$$

The global solution to TP 4 is

$$\mathbf{x}^* = (1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 1), \quad f(\mathbf{x}^*) = -15. \quad (\text{D.2})$$

## E. TP 5 (Alkylation Process Design Problem in Chemical Engineering) [1]

TP 5 has seven decision variables subject to 12 nonconvex, two linear, and 14 boundary constraints. The objective function is to improve the octane number of some olefin feed

by reacting it with isobutane in the presence of acid. The decision variables  $x_n$  are olefin feed rate (barrels/day)  $x_1$ , acid addition rate (thousands of pounds/day)  $x_2$ , alkylate yield (barrels/day)  $x_3$ , acid strength  $x_4$ , motor octane number  $x_5$ , external isobutane-to-olefin ration  $x_6$ , and F-4 performance number  $x_7$ :

$$\begin{aligned}
 \text{Minimize } & f(\mathbf{x}) = (\omega_1 x_1 + \omega_2 x_1 x_6 + \omega_3 x_3 + \omega_4 x_2 + \omega_5 - \omega_6 x_3 x_5) \\
 \text{s.t. } & g_1(\mathbf{x}) = \omega_7 x_6^2 + \omega_8 x_1^{-1} x_3 - \omega_9 x_6 \leq 1, \\
 & g_2(\mathbf{x}) = \omega_{10} x_1 x_3^{-1} + \omega_{11} x_1 x_3^{-1} x_6 - \omega_{12} x_1 x_3^{-1} x_6^2 \leq 1, \\
 & g_3(\mathbf{x}) = \omega_{13} x_6^2 + \omega_{14} x_5 - \omega_{15} x_4 - \omega_{16} x_6 \leq 1, \\
 & g_4(\mathbf{x}) = \omega_{17} x_5^{-1} + \omega_{18} x_5^{-1} x_6 + \omega_{19} x_4 x_5^{-1} - \omega_{20} x_5^{-1} x_6^2 \leq 1, \\
 & g_5(\mathbf{x}) = \omega_{21} x_7 + \omega_{22} x_2 x_3^{-1} x_4^{-1} - \omega_{23} x_2 x_3^{-1} \leq 1, \\
 & g_6(\mathbf{x}) = \omega_{24} x_7^{-1} + \omega_{25} x_2 x_3^{-1} x_7^{-1} - \omega_{26} x_2 x_3^{-1} x_4^{-1} x_7^{-1} \leq 1, \\
 & g_7(\mathbf{x}) = \omega_{27} x_5^{-1} + \omega_{28} x_5^{-1} x_7 \leq 1, \\
 & g_8(\mathbf{x}) = \omega_{29} x_5 - \omega_{30} x_7 \leq 1, \\
 & g_9(\mathbf{x}) = \omega_{31} x_3 - \omega_{32} x_1 \leq 1, \\
 & g_{10}(\mathbf{x}) = \omega_{33} x_1 x_3^{-1} + \omega_{34} x_3^{-1} \leq 1, \\
 & g_{11}(\mathbf{x}) = \omega_{35} x_2 x_3^{-1} x_4^{-1} - \omega_{36} x_2 x_3^{-1} \leq 1, \\
 & g_{12}(\mathbf{x}) = \omega_{37} x_4 + \omega_{38} x_2^{-1} x_3 x_4 \leq 1, \\
 & g_{13}(\mathbf{x}) = \omega_{39} x_1 x_6 + \omega_{40} x_1 - \omega_{41} x_3 \leq 1, \\
 & g_{14}(\mathbf{x}) = \omega_{42} x_1^{-1} x_3 + \omega_{43} x_1^{-1} - \omega_{44} x_6 \leq 1, \\
 & 1500 \leq x_1 \leq 2000, \quad 1 \leq x_2 \leq 120, \quad 3000 \leq x_3 \leq 3500, \\
 & 85 \leq x_4 \leq 93, \quad 90 \leq x_5 \leq 95, \quad 3 \leq x_6 \leq 12, \quad 145 \leq x_7 \leq 162,
 \end{aligned} \tag{E.1}$$

where  $\omega_l$  ( $l = 1, 2, \dots, 44$ ) denotes positive parameters given in Table 12. The global solution to TP 5 is

$$\mathbf{x}^* = (1698.18, 53.66, 3031.30, 90.11, 10.50, 153.53), \quad f(\mathbf{x}^*) = 1227.1978. \tag{E.2}$$

**Table 12:** Coefficients for TP 5.

$l$	$\omega_l$	$l$	$\omega_l$	$l$	$\omega_l$
1	1.715	16	$0.19120592E - 1$	31	0.00061000
2	0.035	17	$0.56850750E + 2$	32	0.0005
3	4.0565	18	1.08702000	33	0.81967200
4	10.000	19	0.32175000	34	0.81967200
5	3000.0	20	0.03762000	35	24500.0
6	0.063	21	0.00619800	36	250.0
7	$0.59553571E - 2$	22	$0.24623121E + 4$	37	$0.10204082E - 1$
8	0.88392857	23	$0.25125634E + 2$	38	$0.12244898E - 4$
9	0.11756250	24	$0.16118996E + 3$	39	0.00006250
10	1.10880000	25	5000.0	40	0.00006250
11	0.13035330	26	$0.48951000E + 6$	41	0.00007625
12	0.00660330	27	$0.44333333E + 2$	42	1.22
13	$0.66173269E - 3$	28	0.33000000	43	1.0
14	0.17239878E - 1	29	0.02255600	44	1.0
15	$0.56595559E - 2$	30	0.00759500		

## F. TP 6 (Optimal Reactor Design Problem) [1]

TP 6 contains eight decision variables subject to four nonconvex inequality constraints and 16 boundary conditions, as follows:

$$\text{Minimize } f(\mathbf{x}) = \left( 0.4x_1^{0.67}x_7^{-0.67} + 0.4x_2^{0.67}x_8^{-0.67} + 10 - x_1 - x_2 \right)$$

$$\text{s.t. } g_1(\mathbf{x}) = 0.0588x_5x_7 + 0.1x_1 \leq 1,$$

$$g_2(\mathbf{x}) = 0.0588x_6x_8 + 0.1x_1 + 0.1x_2 \leq 1,$$

$$g_3(\mathbf{x}) = 4x_3x_5^{-1} + 2x_3^{-0.71}x_5^{-1} + 0.0588x_3^{-1.3}x_7 \leq 1.$$

$$g_4(\mathbf{x}) = 4x_4x_6^{-1} + 2x_4^{-0.71}x_6^{-1} + 0.0588x_4^{-1.3}x_8 \leq 1, \quad 0.1 \leq x_n \leq 10, \quad n = 1, 2, \dots, 8. \quad (\text{F.1})$$

The global solution to TP 6 is

$$\mathbf{x}^* = (6.4747, 2.2340, 0.6671, 0.5957, 5.9310, 5.5271, 1.0108, 0.4004), \quad f(\mathbf{x}^*) = 3.9511. \quad (\text{F.2})$$

### G. TP 7 [1]

TP 7 has four decision variables, two nonconvex inequality constraints, and eight boundary conditions, as follows:

$$\begin{aligned} \text{Minimize } f(\mathbf{x}) &= \left( -x_1 + 0.4x_1^{0.67}x_3^{-0.67} \right) \\ \text{s.t. } g_1(\mathbf{x}) &= 0.05882x_3x_4 + 0.1x_1 \leq 1 \\ g_2(\mathbf{x}) &= 4x_2x_4^{-1} + 2x_2^{-0.71}x_4^{-1} + 0.05882x_2^{-1.3}x_3 \leq 1, \quad 0.1 \leq x_1, x_2, x_3, x_4 \leq 10. \end{aligned} \quad (\text{G.1})$$

The global solution to TP 7 is

$$\mathbf{x}^* = (8.1267, 0.6154, 0.5650, 5.6368), \quad f(\mathbf{x}^*) = -5.7398. \quad (\text{G.2})$$

### H. TP 8 [1]

TP 8 contains three decision variables subject to one nonconvex inequality constraint and six boundary conditions, as follows:

$$\begin{aligned} \text{Minimize } f(\mathbf{x}) &= \left( 0.5x_1x_2^{-1} - x_1 - 5x_2^{-1} \right) \\ \text{s.t. } g_1(\mathbf{x}) &= 0.01x_2x_3^{-1} + 0.01x_1 + 0.0005x_1x_3 \leq 1, \quad 1 \leq x_n \leq 100, \quad n = 1, 2, 3. \end{aligned} \quad (\text{H.1})$$

The global solution to TP 8 is

$$\mathbf{x}^* = (88.2890, 7.7737, 1.3120), \quad f(\mathbf{x}^*) = -83.2540. \quad (\text{H.2})$$

### I. TP 9 [1]

TP 9 contains eight decision variables subject to four nonconvex inequality constraints and 16 boundary conditions, as follows:

$$\begin{aligned} \text{Minimize } f(\mathbf{x}) &= \left( -x_1 - x_5 + 0.4x_1^{0.67}x_3^{-0.67} + 0.4x_5^{0.67}x_7^{-0.67} \right) \\ \text{s.t. } g_1(\mathbf{x}) &= 0.05882x_3x_4 + 0.1x_1 \leq 1, \\ g_2(\mathbf{x}) &= 0.05882x_7x_8 + 0.1x_1 + 0.1x_5 \leq 1, \\ g_3(\mathbf{x}) &= 4x_2x_4^{-1} + 2x_2^{-0.71}x_4^{-1} + 0.05882x_2^{-1.3}x_3 \leq 1, \\ g_4(\mathbf{x}) &= 4x_6x_8^{-1} + 2x_6^{-0.71}x_8^{-1} + 0.05882x_6^{-1.3}x_7 \leq 1, \quad 0.01 \leq x_n \leq 10, \quad n = 1, 2, \dots, 8. \end{aligned} \quad (\text{I.1})$$

The global solution to TP 9 is

$$\mathbf{x}^* = (6.4225, 0.6686, 1.0239, 5.9399, 2.2673, 0.5960, 0.4029, 5.5288), \quad f(\mathbf{x}^*) = -6.0482. \quad (\text{I.2})$$

## J. TP 10 [34]

TP 10 contains three decision variables subject to one nonconvex inequality constraint and six boundary conditions, as follows:

$$\begin{aligned} \text{Minimize} \quad & f(\mathbf{x}) = \left( 5x_1 + 50000x_1^{-1} + 20x_2 + 72000x_2^{-1} + 10x_3 + 144000x_3^{-1} \right) \\ \text{s.t.} \quad & g_1(\mathbf{x}) = 4x_1^{-1} + 32x_2^{-1} + 120x_3^{-1} \leq 1, \quad 1 \leq x_n \leq 1000, \quad n = 1, 2, 3. \end{aligned} \quad (\text{J.1})$$

The global solution to TP 10 is

$$\mathbf{x}^* = (107.4, 84.9, 204.5), \quad f(\mathbf{x}^*) = 6300. \quad (\text{J.2})$$

## K. TP 11 [1, 34]

TP 11 involves five decision variables, six inequality constraints, and ten boundary conditions, as follows:

$$\begin{aligned} \text{Minimize} \quad & g_0(\mathbf{x}) = \left( 5.3578x_3^2 + 0.8357x_1x_5 + 37.2392x_1 \right) \\ \text{s.t.} \quad & g_1(\mathbf{x}) = 0.00002584x_3x_5 - 0.00006663x_2x_5 - 0.0000734x_1x_4 \leq 1, \\ & g_2(\mathbf{x}) = 0.000853007x_2x_5 + 0.00009395x_1x_4 - 0.00033085x_3x_5 \leq 1, \\ & g_4(\mathbf{x}) = 0.00024186x_2x_5 + 0.00010159x_1x_2 + 0.00007379x_3^2 \leq 1, \\ & g_3(\mathbf{x}) = 1330.3294x_2^{-1}x_5^{-1} - 0.42x_1x_5^{-1} - 0.30586x_2^{-1}x_3^2x_5^{-1} \leq 1, \\ & g_5(\mathbf{x}) = 2275.1327x_3^{-1}x_5^{-1} - 0.2668x_1x_5^{-1} - 0.40584x_4x_5^{-1} \leq 1, \\ & g_6(\mathbf{x}) = 0.00029955x_3x_5 + 0.00007992x_1x_3 + 0.00012157x_3x_4 \leq 1, \\ & 78 \leq x_1 \leq 102, \quad 33 \leq x_2 \leq 45, \quad 27 \leq x_3 \leq 45, \quad 27 \leq x_4 \leq 45, \quad 27 \leq x_5 \leq 45. \end{aligned} \quad (\text{K.1})$$

The global solution to TP 11 is

$$\mathbf{x}^* = (78.0, 33.0, 29.998, 45.0, 36.7673), \quad f(\mathbf{x}^*) = 10122.6964. \quad (\text{K.2})$$

## L. TP 12 (a Tension/Compression String Design Problem) [23]

TP 12 involves three decision variables, six inequality constraints, and six boundary conditions. This problem is taken from Huang et al. [23]. This problem attempts to minimize

the weight (i.e.,  $f(\mathbf{x})$ ) of a tension/compression spring subject to constraints on minimum deflection, shear stress, and surge frequency. The design variables are the mean coil diameter  $x_2$ , wire diameter  $x_1$ , and number of active coils  $x_3$ :

$$\begin{aligned} \text{Minimize } & f(\mathbf{x}) = (x_3 + 2)x_2x_1^2 \\ \text{s.t. } & g_1(\mathbf{x}) = 1 - \frac{x_2^3 x_3}{71785 x_1^4} \leq 0, \\ & g_2(\mathbf{x}) = \frac{4x_2^2 - x_1 x_2}{12566(x_2 x_1^3 - x_1^4)} + \frac{1}{5108 x_1^2} - 1 \leq 0, \\ & g_3(\mathbf{x}) = 1 - \frac{140.45 x_1}{x_2^2 x_3} \leq 0, \\ & g_4(\mathbf{x}) = \frac{x_1 + x_2}{1.5} - 1 \leq 0, \quad 0.05 \leq x_1 \leq 2, \quad 0.25 \leq x_2 \leq 1.3, \quad 2 \leq x_3 \leq 15. \end{aligned} \quad (\text{L.1})$$

### M. TP 13 (Pressure Vessel Design Problem) [23]

TP 13 involves four decision variables, four inequality constraints, and eight boundary conditions. This problem attempts to minimize the total cost ( $f(\mathbf{x})$ ), including cost of materials forming and welding. A cylindrical vessel is capped at both ends by hemispherical heads. Four design variables exist: thickness of the shell  $x_1$ , thickness of the head  $x_2$ , inner radius  $x_3$ , and length of the cylindrical section of the vessel, excluding the head  $x_4$ :

$$\begin{aligned} \text{Minimize } & f(\mathbf{x}) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3 \\ \text{s.t. } & g_1(\mathbf{x}) = -x_1 + 0.0193x_3 \leq 0, \\ & g_2(\mathbf{x}) = -x_2 + 0.00954x_3 \leq 0, \\ & g_3(\mathbf{x}) = -\pi x_2^3 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0, \\ & g_4(\mathbf{x}) = x_4 - 240 \leq 0, \quad 0 \leq x_1 \leq 100, \quad 0 \leq x_2 \leq 100, \quad 10 \leq x_3 \leq 200, \quad 10 \leq x_3 \leq 200. \end{aligned} \quad (\text{M.1})$$

### Acknowledgment

The author would like to thank the National Science Council of the Republic of China, Taiwan, for financially supporting this research under Contract no. NSC 100-2622-E-262-006-CC3.

### References

- [1] C. A. Floudas, P. M. Pardalos, C. S. Adjiman, and W. R. Esposito, *Handbook of Test Problems in Local and Global Optimization*, Kluwer, Boston, Mass, USA, 1999.

- [2] L. Kit-Nam Francis, "A generalized geometric-programming solution to economic production quantity model with flexibility and reliability considerations," *European Journal of Operational Research*, vol. 176, no. 1, pp. 240–251, 2007.
- [3] J. F. Tsai, "Treating free variables in generalized geometric programming problems," *Computers & Chemical Engineering*, vol. 33, no. 1, pp. 239–243, 2009.
- [4] P. Xu, "A hybrid global optimization method: the multi-dimensional case," *Journal of Computational and Applied Mathematics*, vol. 155, no. 2, pp. 423–446, 2003.
- [5] C. A. Floudas, *Deterministic Global Optimization*, Kluwer Academic, Boston, Mass, USA, 1999.
- [6] C. I. Sun, J. C. Zeng, J. S. Pan et al., "An improved vector particle swarm optimization for constrained optimization problems," *Information Sciences*, vol. 181, no. 6, pp. 1153–1163, 2011.
- [7] I. G. Tsoulos, "Solving constrained optimization problems using a novel genetic algorithm," *Applied Mathematics and Computation*, vol. 208, no. 1, pp. 273–283, 2009.
- [8] K. Deep and Dipti, "A self-organizing migrating genetic algorithm for constrained optimization," *Applied Mathematics and Computation*, vol. 198, no. 1, pp. 237–250, 2008.
- [9] J. Y. Wu and Y. K. Chung, "Real-coded genetic algorithm for solving generalized polynomial programming problems," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 11, no. 4, pp. 358–364, 2007.
- [10] J. Y. Wu, "Solving constrained global optimization via artificial immune system," *International Journal on Artificial Intelligence Tools*, vol. 20, no. 1, pp. 1–27, 2011.
- [11] L. A. Zadeh, "Fuzzy logic, neural networks, and soft computing," *Communications of the ACM*, vol. 37, no. 3, pp. 77–84, 1994.
- [12] A. Konar, *Computational Intelligence-Principles, Techniques and Applications*, Springer, New York, NY, USA, 2005.
- [13] H. Poorzahedy and O. M. Rouhani, "Hybrid meta-heuristic algorithms for solving network design problem," *European Journal of Operational Research*, vol. 182, no. 2, pp. 578–596, 2007.
- [14] W. F. Abd-El-Wahed, A. A. Mousa, and M. A. El-Shorbagy, "Integrating particle swarm optimization with genetic algorithms for solving nonlinear optimization problems," *Journal of Computational and Applied Mathematics*, vol. 235, no. 5, pp. 1446–1453, 2011.
- [15] R. J. Kuo and Y. S. Han, "A hybrid of genetic algorithm and particle swarm optimization for solving bi-level linear programming problem—a case study on supply chain model," *Applied Mathematical Modelling*, vol. 35, no. 8, pp. 3905–3917, 2011.
- [16] P. S. Shelokar, P. Siarry, V. K. Jayaraman, and B. D. Kulkarni, "Particle swarm and ant colony algorithms hybridized for improved continuous optimization," *Applied Mathematics and Computation*, vol. 188, no. 1, pp. 129–142, 2007.
- [17] Y. Hu, Y. Ding, and K. Hao, "An immune cooperative particle swarm optimization algorithm for fault-tolerant routing optimization in heterogeneous wireless sensor networks," *Mathematical Problems in Engineering*, vol. 2012, Article ID 743728, 19 pages, 2012.
- [18] X. Zhao, "A perturbed particle swarm algorithm for numerical optimization," *Applied Soft Computing*, vol. 10, no. 1, pp. 119–124, 2010.
- [19] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, New York, NY, USA, 1994.
- [20] Z. Michalewicz, "Genetic algorithm, numerical optimization, and constraints," in *Proceedings of the 6th International Conference on Genetic Algorithms*, pp. 151–158, San Mateo, Calif, USA, 1995.
- [21] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanics and Engineering*, vol. 186, no. 2–4, pp. 311–338, 2000.
- [22] N. Cruz-Cortés, D. Trejo-Pérez, and C. A. Coello Coello, "Handling constraints in global optimization using an artificial immune system," in *Proceedings of the 4th International Conference on Artificial Immune Systems*, pp. 234–247, Banff, Canada,, 2005.
- [23] F.-Z. Huang, L. Wang, and Q. He, "An effective co-evolutionary differential evolution for constrained optimization," *Applied Mathematics and Computation*, vol. 186, no. 1, pp. 340–356, 2007.
- [24] E. Zahara and Y. T. Kao, "Hybrid Nelder-Mead simplex search and particle swarm optimization for constrained engineering design problems," *Expert Systems with Applications*, vol. 36, no. 2, part 2, pp. 3880–3886, 2009.
- [25] C. R. Houck, J. A. Joines, and M. G. Kay, "A genetic algorithm for function optimization: a matlab implementation," in *NSCU-IE TR 95-09*, North Carolina State University, Raleigh, NC, USA, 1995.
- [26] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, Perth, Australia, 1995.

- [27] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *Proceedings of the IEEE International Conference on Evolutionary Computation*, pp. 69–73, Anchorage, Alaska, USA, 1998.
- [28] M. Clerc, "The swarm and the queen: towards a deterministic and adaptive particle swarm optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1951–1957, Washington, DC, USA, 1999.
- [29] 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.
- [30] A. P. Engelbrecht, *Fundamentals of Computational Swarm Intelligence*, John Wiley & Sons, 2005.
- [31] N. K. Jerne, "Idiotypic networks and other preconceived ideas," *Immunological Reviews*, vol. 79, pp. 5–24, 1984.
- [32] L. N. de Castro and F. J. Von Zuben, "Artificial Immune Systems: Part I—Basic Theory and Applications," FEEC/Univ. Campinas, Campinas, Brazil, 1999, <ftp://ftp.dca.unicamp.br/pub/docs/vonzuben/tr.dca/trdca0199.pdf>.
- [33] C. A. Coello Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art," *Computer Methods in Applied Mechanics and Engineering*, vol. 191, no. 11-12, pp. 1245–1287, 2002.
- [34] M. J. Rijckaert and X. M. Martens, "Comparison of generalized geometric programming algorithms," *Journal of Optimization Theory and Applications*, vol. 26, no. 2, pp. 205–242, 1978.
- [35] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997.

*Research Article*

## **A Hybrid Algorithm Based on ACO and PSO for Capacitated Vehicle Routing Problems**

**Yucheng Kao, Ming-Hsien Chen, and Yi-Ting Huang**

*Department of Information Management, Tatung University, 40 Chungshan N. Road, Sec. 3,  
Taipei 104, Taiwan*

Correspondence should be addressed to Yucheng Kao, [ykao@ttu.edu.tw](mailto:ykao@ttu.edu.tw)

Received 24 February 2012; Revised 5 May 2012; Accepted 8 May 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Yucheng Kao 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 vehicle routing problem (VRP) is a well-known combinatorial optimization problem. It has been studied for several decades because finding effective vehicle routes is an important issue of logistic management. This paper proposes a new hybrid algorithm based on two main swarm intelligence (SI) approaches, ant colony optimization (ACO) and particle swarm optimization (PSO), for solving capacitated vehicle routing problems (CVRPs). In the proposed algorithm, each artificial ant, like a particle in PSO, is allowed to memorize the best solution ever found. After solution construction, only elite ants can update pheromone according to their own best-so-far solutions. Moreover, a pheromone disturbance method is embedded into the ACO framework to overcome the problem of pheromone stagnation. Two sets of benchmark problems were selected to test the performance of the proposed algorithm. The computational results show that the proposed algorithm performs well in comparison with existing swarm intelligence approaches.

### **1. Introduction**

The vehicle routing problem (VRP) is a well-known combinatorial optimization problem in which the computational complexity is NP-hard. The capacitated vehicle routing problem is one of the variants of VRPs. The objective of CVRPs is to minimize the total traveling distance of vehicles which serve a set of customers. The following constraints are considered in a typical CVRP: each route is a tour which starts from a depot, visits a subset of the customers, and ends at the same depot; each customer must be assigned to exactly one of the vehicles; each customer has its own demand and the total demand of customers assigned to a vehicle must not exceed the vehicle capacity. In the past decades, researchers proposed different strategies to solve the CVRP. One of them is to cluster customers into different routes and then to arrange the visiting sequence for each route. The objective function value will

be apparently influenced by the results of customer clustering and sequencing. For more detailed descriptions of vehicle routing problems, the reader may refer to the articles by Laporte [1], Osman [2], and Cordeau et al. [3].

Because the CVRP is an NP-hard problem [4], the optimal solution of a large-size instance cannot be found within a reasonable time. To overcome this difficulty, many classical heuristic methods and metaheuristic methods are proposed in the past five decades. Some metaheuristic algorithms can provide competitive solutions to the CVRP, such as Simulated Annealing (SA), Tabu Search (TS), and Genetic Algorithm (GA). Metaheuristic algorithms have some advantages, for example, the abilities to escape from local optima through stochastic search, to speed convergence using solution replacement, to guide the search direction with the elitist strategy, and so on. The following paragraph gives a brief review of some articles which use these metaheuristic algorithms to solve CVRPs.

Barbarosoglu and Ozgur [5] designed a TS-based algorithm using a new neighborhood generation procedure for the single-depot vehicle routing problems. The neighbors are defined by using two procedures: the first one ignores the scattering patterns of customer locations, and the second one considers the underlying clustering of customer locations. Baker and Ayechew [6] put forward a hybrid of GA algorithms with neighborhood search methods. The pure GA has three specific processes: initialization, reproduction, and replacement. The neighborhood search methods are used to accelerate the convergence of GA. Computational results showed that this approach is competitive with published results obtained using TS and SA. Lin et al. [7] proposed a hybrid algorithm which takes the advantages of SA and TS. In their paper, SA is used to adjust the probability of accepting worse solutions according to the extent of solution improvement and the annealing temperature, while TS is embedded in the framework of SA to avoid cycling to some extent while searching for neighborhood.

In the recent ten years, swarm intelligence, a new category of metaheuristics, has emerged and attracted researchers' attention. Swarm intelligence mimics the social behavior of natural insects or animals to solve complex problems. Some commonly used swarm intelligence algorithms for the solution of the CVRP include ant colony optimization (ACO), particle swarm optimization (PSO) and artificial bee colony [8].

ACO is a population-based swarm intelligence algorithm and was proposed by Dorigo and Gambardella [9]. This algorithm has been inspired by the foraging behavior of real ant colonies and originally designed for the traveling salesman problem (TSP). The artificial ants use pheromone laid on trails as an indirect communication medium to guide them to construct complete solution routes step by step. More pheromone deposits on better routes attract more ants for later search. This effect is called dynamic positive feedback and helps speed convergence of ACO. Recently, some researchers have studied vehicle routing problems using ACO algorithms. Applying ACO to the CVRP is quite natural, since we can view ant nests as depots, artificial ants as vehicles, foods as customers, and trails as routes. Some of the relevant papers are briefly reviewed as follows.

Bell and McMullen [10] made modifications of the ACO algorithm in order to solve the vehicle routing problem. They used multiple ant colonies to search vehicle routes. Each vehicle route is marked with unique pheromone deposits by an ant colony, but the communication among ant colonies is limited. Later, Liu and Cai [11] proposed a new multiple ant colonies technique, which allows ant colonies to communicate with each other in order to escape from local optima. Chen and Ting [12] developed an improved ant colony system algorithm, in which pheromone trails will be reset to initial values for restarting the search if the solution is not improved after a given number of iterations. Zhang and

Tang [13] hybridized the solution construction mechanism of ACO with scatter search (SS). The algorithm stores better solutions in a reference set. Some new solutions are generated by combining solutions selected from the reference set, and some are produced by using the conventional ACO method. Yu et al. [14] also developed an improved ant colony optimization for vehicle routing problems. Their algorithm uses the ant-weight strategy to update pheromone in terms of solution quality and the contribution of each edge to the solution. Lee et al. [15] proposed an enhanced ACO algorithm for the CVRP. Their algorithm adopts the concept of information gain to measure the variation of pheromone concentrations and hence to dynamically adjust the value of heuristic parameter ( $\beta$ ) which determines the importance of heuristic value ( $\eta$ ) at different iterations.

PSO is also a population-based swarm intelligence algorithm and was originally proposed by Kennedy and Eberhart [16]. PSO is inspired by social behavior of bird flocking. It has been shown that PSO can solve continuous optimization problems very well. In the PSO, solution particles try to move to better locations in the solution space. The movements of particles are guided by the individuals' and the swarm's best positions. PSO can converge very fast due to its two unique mechanisms: memorizing personal best experiences ( $P_{best}$ ) and information sharing of global best experiences ( $G_{best}$ ). Note that the  $G_{best}$  solution of the particle swarm is equal to the  $P_{best}$  solution of the best particle. In 1998, Shi and Eberhart [17] enhanced PSO by adding the concept of inertia weight, which becomes the standard version of PSO.

PSO being originally developed for continuous optimization problems, a special solution representation or solution conversion should be designed first in order to solve CVRPs. Chen et al. [18] first proposed a PSO-based algorithm to solve the CVRP. In their approach, each iteration has two main steps: customers are first clustered by using a discrete PSO algorithm (DSPO) and then sequenced by applying a SA algorithm. Due to its long solution strings, their approach takes much computational time in solving large scale problems. To improve Chen et al.'s work, Kao and Chen [19] addressed a new solution representation and solved the CVRP with a combinatorial PSO algorithm. Ai and Kachitvichyanukul [20] presented two solution representations for solving CVRPs. For example, in their second solution representation (SR-2), each vehicle is represented in three dimensions, with two for the reference point and one for the vehicle coverage radius. SR-2 employs these points and radius to construct vehicle routes. The particle solutions are adjusted by using a continuous PSO. Marinakis et al. [21] proposed a hybrid PSO algorithm to tackle large-scale vehicle routing problems. Their proposed algorithm combines a PSO algorithm with three heuristic methods, with the first for particle initialization, the second for solution replacement, and the third for local search.

This study proposes a new hybrid algorithm for the capacitated vehicle routing problem, which is based on the framework of ACO and is hybridized with the merits of PSO. The reasons why ACO and PSO, rather than SA, TS, and GA, are adopted in the proposed algorithm are given as follows. First, SA and TS perform the so-called single-starting-point search and thus their performance relies highly on a good initial solution. However, GA, ACO, and PSO are all population-based algorithms and can start the search from multiple points. Their initial solutions have little influence on their performance. Thus, we consider adopting the population-based algorithms to solve CVRPs. Second, ACO and PSO have memory that enables the algorithms to retain knowledge of good solutions, while the genetic operators of GA may destroy previously learned knowledge when producing the offspring. In view of these two considerations, we select ACO and PSO as the solution approach for this paper.

In the past, most relevant papers adopted either ACO [10–15] or PSO [18–21] alone without trying to use both in combination for solving CVRPs. In this paper, we try to integrate ACO with PSO to develop a new hybrid approach which can take advantage of both algorithms. That is, the proposed algorithm uses the solution construction approach of ACO to cluster customers and build routes at the same time and use the short-term memory inspired by PSO to speed convergence through laying pheromone on the routes of  $G_{best}$  and  $P_{best}$  solutions.

Like most ACO-based algorithms, the proposed hybrid algorithm also faces the limitation of pheromone stagnation, which results in premature convergence. To solve this problem, Shuang et al. [22] employed the mechanism of PSO to modify the pheromone updating rules of ACO. Their proposed algorithm is called PS-ACO and is used to solve traveling salesman problems (TSPs). PS-ACO can improve the performance of ACO to some extent, but it may still be trapped in local optima due to the overaccumulation of pheromone on some edges when solving more complicated problems like CVRPs. To attain a high degree of search accuracy, this paper proposes a pheromone disturbance approach to overcome the problem of pheromone stagnation. The remainder of this paper is organized as follows. Section 2 defines the mathematical formulation of the CVRP. The proposed methodology is described in Section 3. Section 4 presents computational results. Finally, conclusions are drawn in the last section.

## 2. Mathematical Model of CVRP

This section gives a typical mathematical formulation of the CVRP, including notations, objective function, and constraint equations.

*Notations.* 0: index of depots;

$N$ : total number of customers;

$K$ : total number of vehicles;

$C_{ij}$ : cost incurred when traveling from customer  $i$  to customer  $j$ ;

$S_i$ : service time needed for customer  $i$ ,  $S_0 = 0$ ;

$Q$ : maximum loading capacity of a vehicle;

$T$ : maximum traveling distance of a vehicle;

$d_i$ : demand of customer  $i$ ,  $d_0 = 0$ ;

$X_{ij}^k$ : 0-1 variable, where  $X_{ij}^k = 1$  if the edge from customer  $i$  to customer  $j$  is traveled by vehicle  $k$ ; otherwise,  $X_{ij}^k = 0$ . Note that  $i \neq j$ ;

$p$ : penalty coefficient;

$R$ : set of customers served by a vehicle, and  $|R|$  is the cardinality of  $R$ .

*Objective function*

$$\text{Minimize} \quad \sum_{i=0}^N \sum_{j=0}^N \sum_{k=1}^K C_{ij} X_{ij}^k, \quad (2.1)$$

$$\text{subject to} \quad \sum_{k=1}^K \sum_{i=0}^N X_{ij}^k = 1, \quad j = 1, 2, \dots, N, \quad (2.2)$$

$$\sum_{k=1}^K \sum_{j=0}^N X_{ij}^k = 1, \quad i = 1, 2, \dots, N, \quad (2.3)$$

$$\sum_{i=0}^N X_{iu}^k - \sum_{j=0}^N X_{uj}^k = 0, \quad k = 1, 2, \dots, K; \quad u = 1, 2, \dots, N, \quad (2.4)$$

$$\sum_{i=0}^N \sum_{j=0}^N X_{ij}^k d_i \leq Q, \quad k = 1, 2, \dots, K, \quad (2.5)$$

$$\sum_{i=0}^N \sum_{j=0}^N X_{ij}^k (C_{ij} + S_i) \leq T, \quad k = 1, 2, \dots, K, \quad (2.6)$$

$$\sum_{j=1}^N X_{ij}^k = \sum_{j=1}^N X_{ji}^k \leq 1, \quad i = 0; \quad k = 1, 2, \dots, K, \quad (2.7)$$

$$\sum_{i,j \in R} X_{ij}^k \leq |R| - 1, \quad R \subseteq \{1, \dots, N\}, \quad 2 \leq |R| \leq N - 1; \quad k = 1, 2, \dots, K, \quad (2.8)$$

$$X_{ij}^k \in \{0, 1\}, \quad i, j = 0, 1, \dots, N; \quad k = 1, 2, \dots, K. \quad (2.9)$$

Equation (2.1) is the objective function of the CVRP. Equations (2.2) and (2.3) ensure that each customer can be served by only one vehicle. Equation (2.4) maintains the continuity at each node for every vehicle. Equation (2.5) ensures that the total customer demand of a vehicle cannot exceed its maximum capacity. Similarly, (2.6) ensures that the total route distance of a vehicle cannot exceed its route length limit. Equation (2.7) makes sure that every vehicle can be used at most once and must start and end at the depot. The subtour elimination constraints are given in (2.8). Equation (2.9) is the integrality constraint.

### 3. PACO Algorithm

This section describes the proposed solution algorithm to the capacitated vehicle routing problem. The algorithm, called PACO, hybridizes the solution construction mechanism of

ACO and the short-term memory mechanism of PSO to find optimal or near optimal vehicle routes.

### **3.1. Basic Idea**

The PACO algorithm incorporates the merits of PSO into the ACO algorithm. One of the advantages of applying ACO to the CVRP is that ACO can cluster customers and build routes at the same time. However, laying pheromone (long-term memory) on trails as ant communication medium is time consuming. The merit of PSO is that it can speed convergence through memorizing personal and global best solutions to guide the search direction. Inspired by the merit of PSO, the PACO algorithm allows artificial ants to memorize their own best solution so far and to share the information of swarm best solution. Hence PACO can speed convergence through intensifying pheromone on routes of  $G_{\text{best}}$  and  $P_{\text{best}}$  solutions.

To avoid falling into local optima, our approach employs elitist strategy, pheromone disturbance, and short-term memory resetting to resolve pheromone stagnation. After ants complete solution construction, the first  $r$  iteration-best ants are allowed to perform local search to improve their current solutions. After that, all ants update their  $P_{\text{best}}$  solutions and  $G_{\text{best}}$  solution. The ants with better  $P_{\text{best}}$  solutions are called elite ants. Pheromone updating is conducted by these elite ants only. Elite ants lay pheromone on their  $P_{\text{best}}$  solution routes in a distributed way. The elitist strategy updates the pheromone in terms of solution quality and attracts ants searching for solutions around distributed  $P_{\text{best}}$  solution paths.

When the  $G_{\text{best}}$  solution is not improved within a given number of iterations, PACO will carry out pheromone disturbance to change pheromone trails randomly in order to find new solutions. Pheromone disturbance can prevent the paths of the  $P_{\text{best}}$  solution of elite ants from becoming too dominant. After pheromone disturbance, the paths of the  $P_{\text{best}}$  solution of elite ants may become dominant again because they can still evoke memories of current  $P_{\text{best}}$  solutions which determine the way of laying pheromone. To avoid that, the algorithm should allow ants to reset their  $P_{\text{best}}$  solutions. It means that the ants will discard their current  $G_{\text{best}}$  and  $P_{\text{best}}$  solutions and find new ones in the following iteration according to the disturbed pheromone trails. The algorithm allows any ant, not limited to elite ants, to change their  $P_{\text{best}}$  solutions if and only if its  $P_{\text{best}}$  solution is very similar to the  $G_{\text{best}}$  solution.

### **3.2. Solution Representation**

For solving the CVRP, each artificial ant represents a candidate solution. A solution is represented with the route representation of all vehicle tours. Let 0 denote the depot and positive integers from 1 to  $N$  represent the customers. Suppose that the total number of vehicles is  $K$ . The solution code is a permutation of 1 to  $N$  and  $(K - 1)$  zeros. Zeros divide a solution code into  $K$  segments, with each of them representing a vehicle route. For example, a solution code (1, 2, 3, 0, 4, 5, 6, 0, 7, 8, 9) means that customers 1, 2, 3 are serviced by vehicle 1; customer 4, 5, 6 by vehicle 2; customers 7, 8, 9 by vehicle 3. Thus, the length of a solution string is  $(N + K - 1)$ . It is clear that this solution representation scheme can fully meet the constraints defined in (2.2)–(2.4).

The goodness of a solution is evaluated by using the objective function (2.1). Here, we modify the objective function in order to handle the infeasible solutions violating capacity

and length constraints (see (2.5) and (2.6)). Two penalty functions are added to the original objective function, as defined in (3.1), one for excess vehicle capacity and the other for excess route length. These penalty functions increase the objective function value of infeasible solutions so as to prefer feasible solution to be selected as elite ants which are allowed to perform local search and to update pheromone on their routes (see Sections 3.5 and 3.6).

$$\text{Minimize } f = \sum_{i=0}^N \sum_{j=0}^N \sum_{k=1}^K C_{ij} X_{ij}^k + p \sum_{k=1}^K \left[ \max \left\{ 0, \sum_{i=0}^N \sum_{j=0}^N X_{ij}^k d_i - Q \right\} + \max \left\{ 0, \sum_{i=0}^N \sum_{j=0}^N X_{ij}^k (C_{ij} + S_i) - T \right\} \right]. \quad (3.1)$$

### 3.3. Main Steps

The main steps of PACO are solution construction, local search,  $G_{\text{best}}$  and  $P_{\text{best}}$  updating, pheromone updating, and pheromone disturbance. Suppose there are  $m$  ants starting from the depot. Each ant selects customers to construct a solution route by applying the state transition rules of ACO. When all ants finish solution construction, the top  $r$  best ants perform local search to improve their solutions. Then, ants update their short-term memory:  $G_{\text{best}}$  solution and individual  $P_{\text{best}}$  solutions. After that, PACO updates the pheromone trails according to the  $G_{\text{best}}$  solution and the  $P_{\text{best}}$  solutions of elite ants. When the  $G_{\text{best}}$  is not improved over  $w$  consecutive iterations, PACO performs pheromone disturbance to modify current pheromone trails. PACO also resets the  $P_{\text{best}}$  solutions of some ants in order to prevent them from laying pheromone on the same routes again. The flowchart of the PACO algorithm is shown in Figure 1 and described briefly as follows.

*Step 1* (Initialization). Initialize all parameters.

*Step 2* (Solution construction). Let  $m$  ants construct solution routes.

*Step 3* (Local search). The top  $r$  best ants perform local search.

*Step 4*. Update the  $G_{\text{best}}$  and  $P_{\text{best}}$  solutions of ants and select  $r$  elite ants.

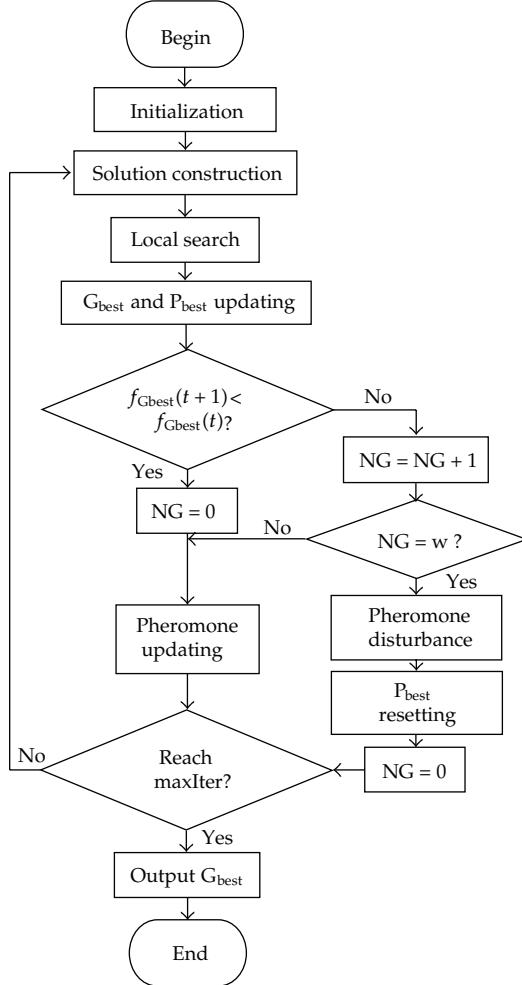
*Step 5*. If  $G_{\text{best}}$  is not improved within  $w$  successive iterations, go to Step 6; otherwise, go to Step 7.

*Step 6* (Pheromone disturbance). Randomly disturb the pheromone matrix, reset the  $P_{\text{best}}$  solutions for some ants, and go to Step 8.

*Step 7* (Pheromone updating). Update the pheromone matrix based on the  $P_{\text{best}}$  solutions of elite ants.

*Step 8*. If iteration number reaches the maximum number of iterations (MaxIte), go to Step 9; otherwise, go to Step 2 for the next iteration.

*Step 9*. Output  $G_{\text{best}}$ , the best solution ever found.



**Figure 1:** Flowchart of PACO.

### 3.4. Solution Construction

PACO can cluster customers into  $K$  vehicles and arrange vehicle visiting sequences at the same time. At each of the iterations,  $m$  ants construct individual solutions independently. Each solution contains  $K$  vehicle routes. Each ant starts from the depot, selects customers for the first vehicle, moves back to the depot before the capacity or distance limit is violated, then restarts from the depot and selects customers for the second vehicle. The procedure is repeated until all customers are selected and their sequences are arranged in used vehicles. It is possible that the last vehicle will serve all of the remaining customers even if it violates the capacity or distance limit. The procedure of ant solution construction ensures that the constraints defined in (2.7) and (2.8) can be satisfied.

The customer selection follows the state transition rule defined in (3.2). Suppose ant  $s$  is moving from customer  $i$  to the next customer,  $v$ :

$$v = \begin{cases} \arg \max_{j \in U_s} [(\tau_{ij})^\alpha (\eta_{ij})^\beta] & q \leq q_0, \\ V & q > q_0, \end{cases} \quad (3.2)$$

$$V: P_{ij} = \frac{(\tau_{ij})^\alpha (\eta_{ij})^\beta}{\sum_{j \in U_s} (\tau_{ij})^\alpha (\eta_{ij})^\beta}, \quad (3.3)$$

where  $U_s$  is the set of customers that remain to be selected by ant  $s$  positioned on customer  $i$ ,  $\tau_{ij}$  is the pheromone trail on edge  $(i, j)$ ,  $\eta_{ij}$  is the inverse of the distance of edge  $(i, j)$ ,  $\alpha$  and  $\beta$  are parameters which determine the relative importance of pheromone versus distance,  $q$  is a random number uniformly distributed in the interval of 0 and 1,  $q_0$  is a parameter ranged between 0 and 1, and  $P_{ij}$  is the probability that ant  $s$  moves from customer  $i$  to customer  $j$ . If  $q \leq q_0$ , then ant  $s$  uses the greedy method (3.2) to select the next customer; otherwise, it uses the probabilistic rule (3.3) to determine the next customer. If vehicle  $k$  is full or reaches its distance limit, ant  $s$  has to go back to the depot and restarts from the depot to load the next vehicle. However, if vehicle  $k$  is the last vehicle of ant  $s$ , then it has to serve all of the remaining customers, even if the capacity or distance limit is violated.

### 3.5. Local Search

After all ants complete route construction, only the first  $r$  iteration-best ants can perform local search to improve their current solutions. PACO performs three types of neighborhood search: sequence inversion, insertion, and swap. These methods are often used in CVRP papers to improve iteration solutions, as is the case with [7, 8, 15]. Sequence inversion selects a vehicle at random from the current solution string, chooses two customers randomly, and then inverts the substring between these two customers. The swap operation selects two customers at random and then swaps these two customers in their positions. The insertion operation selects a customer at random and then inserts the customer in a random position. The swap operation may select two customers in different vehicles, and the insertion operation may assign the selected customer to a different vehicle. For such a case, the capacity and distance limits of the vehicles have to be checked. If the constraints are violated, the new solution becomes invalid and another one should be generated.

The whole procedure of local search for an ant solution is controlled by using a simulated annealing approach. SA performs neighborhood search  $R$  times at each temperature  $T(t)$ . Each time one of the three local search methods is randomly selected with equal probability to implement neighborhood search. A worse solution may have a chance to be accepted according to the following equations:

$$\Delta = f(S') - f(S),$$

$$P(S') = \exp\left(-\frac{\Delta}{T(t)}\right), \quad (3.4)$$

where  $S$  is the current solution,  $S'$  is the new solution,  $f(S)$  is the objective function value of  $S$ ,  $f(S')$  is the new objective function value,  $t$  is the current temperature, and  $P(S')$  is the probability that SA accepts new solution  $S'$ . After carrying out neighborhood search  $R$  times, SA reduces the temperature. New temperature ( $T(t+1)$ ) is equal to  $\lambda \times T(t)$ , where  $0 < \lambda < 1$ . PACO adopts a best improvement strategy in its local search step. That is, the best-so-far solution will be recorded during the run of the SA algorithm. When the termination condition is met, the ant solution will be replaced with the best-so-far solution of SA if the latter is really better.

After solution construction and local search, all ants compare their  $P_{\text{best}}$  solutions with their iteration solutions and perform  $P_{\text{best}}$  replacement if the iteration solutions are better. Then, ants are ranked in terms of the goodness of their  $P_{\text{best}}$  solutions, and the first  $r$  ants are the elite ants. Of course, the  $G_{\text{best}}$  solution of all ants is equal to the  $P_{\text{best}}$  solution of the best elite ant.

### 3.6. Pheromone Updating

Pheromone trails play the role of long-term memory in the PACO algorithm. Global updating is used to enhance the search in the neighborhood of better solutions. The paths of better solutions have higher levels of pheromone so as to attract more ants for later search. The pheromone on other paths evaporates over time and becomes less attractive to ants. Therefore, convergence of PACO can be accelerated by intensifying pheromone on the paths of elite solutions. The pheromone updating rule is defined as follows:

$$\begin{aligned} \tau_{ij} &= (1 - \rho)\tau_{ij} + \sum_{s=2}^r \Delta\tau_{ij}^{P_{\text{best}s}} + \Delta\tau_{ij}^{G_{\text{best}}}, \\ \Delta\tau_{ij}^{P_{\text{best}s}} &= \frac{1}{f_{P_{\text{best}}}^s}, \\ \Delta\tau_{ij}^{G_{\text{best}}} &= \frac{1}{f_{G_{\text{best}}}}, \end{aligned} \quad (3.5)$$

where  $\rho$  is the pheromone evaporation rate and ranges between 0 and 1,  $r$  is the total number of elite ants,  $\Delta\tau_{ij}^{P_{\text{best}s}}$  and  $\Delta\tau_{ij}^{G_{\text{best}}}$ , are the pheromone added by elite ant  $s$  and the best ant, respectively, and  $f_{P_{\text{best}}}^s$  and  $f_{G_{\text{best}}}$  are the objective function values of elite ant  $s$  and the best ant, respectively.

### 3.7. Pheromone Disturbance

Since only elite ants lay pheromone on the routes of their  $P_{\text{best}}$  solutions, the pheromone on the paths of elite solutions will accumulate very fast. It leads to the state of pheromone stagnation, and the search may be trapped by local optima. To overcome this problem, PACO adopts pheromone disturbance to escape from local optima and to explore different areas of the search space. Pheromone disturbance is performed when the  $G_{\text{best}}$  solution of ants is not updated (improved) up to  $w$  successive iterations.  $w$  is called the disturbance period. Large  $w$  makes PACO easy to raise the chances of settling for a false optimum, while small  $w$  retards

**Table 1:** Original pheromone matrix.

<i>i</i>	1	2	3	4	5
1	0	0.1	0.8	0.2	0.5
2		0	0.2	0.1	0.2
3			0	0.3	0.5
4				0	0.8
5					0

the search convergence and takes more computational time. It is suggested that  $w$  be equal to the number of customers.

The basic idea of pheromone disturbance is similar to arithmetic crossover in GA, but it is used to produce new pheromone trails, rather than to generate new solutions. Pheromone disturbance has three steps. The first step is to select edges for disturbance. The disturbance rate ( $\mu$ ) determines the probability that an edge is selected for disturbance. If  $\mu$  is set too high, it will be difficult to retain previous search experiences; on the other hand, if it is set too low, the effect of disturbance will not be evident. The second step is to cluster selected edges into groups, each of which has two edges with the same customer node. The groups with single edges will be abandoned. The third step is to use a random number  $q \in [0, 1]$  to determine a disturbance type for each of the paired edges. There are three types of pheromone disturbance: unchanging, replacement, and weighted average. As defined in (3.6), paired edges  $(i, j)$  and  $(i, u)$  have three possible results. That is, the pheromone on edge  $(i, j)$  may remain the same, be replaced by the pheromone level of edge  $(i, u)$ , or be replaced with the weighted average of the pheromone levels of these two edges.

$$\tau_{ij}^{t+1} = \begin{cases} \tau_{ij}^t & q < 0.2, \\ \tau_{iu}^t & 0.2 \leq q < 0.4, \\ \delta\tau_{ij}^t + (1 - \delta)\tau_{iu}^t, & j \neq u, q \geq 0.4, \end{cases} \quad (3.6)$$

where  $\delta$  is a uniform random number in the range  $[0, 1]$  and determines the ratios of pheromone on the two edges.

After pheromone disturbance, ant  $s$  positioned on customer node  $i$  has a chance to explore different edges rather than to keep selecting the same next customer node to move to. Hence, pheromone disturbance increases the probability of finding optimal solutions. Note that we deal with symmetric CVRPs in this paper, where the distances between customer nodes are independent of the direction of traversing the edges. The same situation applies to pheromone trails. Accordingly,  $d_{ij} = d_{ji}$  and  $\tau_{ij} = \tau_{ji}$  for each pair of nodes.

We use an example to illustrate the idea of pheromone disturbance. Suppose we have five customers, then there are 20 edges with pheromone deposits. The original pheromone matrix is shown in Table 1. Since  $\tau_{ij} = \tau_{ji}$  for each pair of nodes, the algorithm considers only half the edges to be disturbed.

The first step is to select edges for disturbance. Each edge is selected with a probability of  $\mu = 0.3$ . For each edge, we generate a random number from uniform distribution in the range between 0 and 1. The generated random numbers are shown in Table 2. A edge is selected if its random number is less than  $\mu$ . Table 2 tells us that four edges are selected and marked in bold type. The second step is to pair two edges that have the same customer node.

**Table 2:** Edges selected for disturbance.

<i>i</i>		1	2	3	4	5
1		0	0.5	<b>0.2</b>	0.6	<b>0.1</b>
2			0	<b>0.1</b>	<b>0.2</b>	0.7
3				0	0.8	0.4
4					0	0.5
5						0

**Table 3:** Disturbance results.

Edge	$\tau_{ij}(t)$	<i>q</i>	Option	$\tau_{ij}(t+1)$
(1–3)	0.8	0.1	1	0.8
(1–5)	0.5	0.8	3	0.65
(2–3)	0.2	0.3	2	0.1
(2–4)	0.1	0.5	3	0.11

**Table 4:** New pheromone matrix.

<i>i</i>		1	2	3	4	5
1		0	0.1	<b>0.8</b>	0.2	<b>0.65</b>
2			0	<b>0.1</b>	<b>0.11</b>	0.2
3				0	0.3	0.5
4					0	0.8
5						0

Scanning the matrix in Table 2 row by row, we obtain the pairing results: {edge (1,3), edge (1,5)} and {edge (2,3), edge (2,4)}. The third step uses (3.6) to determine new pheromone trails for paired edges. Random number *q* is first generated to determine a disturbance type for each selected edge. The disturbance results are shown in Table 3. For example, for the first paired edges, option 1 (unchanging) applies to edge (1,3) and option 3 (weighted average) applies to edge (1,5). Note that here  $\delta$  is a random number in the range of [0, 1]. In our example of edge (1,5),  $\delta$  is equal to 0.5. The new pheromone matrix is presented in Table 4.

### 3.8. $P_{best}$ Solution Resetting

Since pheromone updating is based on the  $P_{best}$  solutions of elite ants, it may diminish the effect of pheromone disturbance very fast in the following iterations. To avoid that situation, PACO resets the  $P_{best}$  solutions of some better ants. That is, the  $P_{best}$  solutions of the ants are removed from their memory and find new ones in the next iteration. Note that not all of the ants need to reset their  $P_{best}$  solutions. The resetting is determined by the difference in objective function value between  $P_{best}$  and  $G_{best}$  solutions. For ant *s*, if the difference is less than or equal to a threshold,  $\Delta f$ , (i.e.,  $\Delta f \leq f_{P_{best}}^s - f_{G_{best}}$ ), then ant *s* has to reset its  $P_{best}$  solution.

## 4. Computational Results

The PACO algorithm described in Section 3 was coded in Java, and all experiments were performed on a personal computer with Intel Core 2 CPU T7500 running at 2.20 GHz. Two sets of benchmark problems were selected to evaluate the effectiveness of our proposed algorithm for the capacitated vehicle routing problem.

The first set has 16 test problems and can be downloaded from the website <http://www.branchandcut.org/VRP/data/>. The problems in this benchmark set are subject to capacity constraints only. The total number of customers varies from 29 to 134, and the total number of vehicles ranges from 3 to 10. The locations of customers appear in clusters in the problems with their names initiated with B and M, while in the remaining problems customers are randomly scattered or semiclustered. The first benchmark set was used by Chen et al. [18] and Ai and Kachitvichyanukul [20] to test their PSO-based algorithms.

The second benchmark set can be downloaded from the website <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/vrpinfo.html>. It has been widely used in previous studies and contains 14 classical test problems selected from Christofides et al. [23]. Problems 1, 2, 3, 4, 5, 11, and 12 consider the constraint of capacity only, while the remaining problems are subject to the capacity and distance limits. The total number of customers varies from 50 to 199, and the total number of vehicles ranges from 5 to 18. Besides, customers are randomly distributed in the first ten problems whereas customers are clustered in the last four problems.

The PACO parameters set as follows were found to be robust for most of the test problems according to our pilot tests. PACO parameters are maximum iteration number  $\text{MaxIte} = 1000$ , population size  $\text{pop} = N/2$ , number of elite ants  $r = 3$ , penalty coefficient  $p = 100$ ,  $q_0 = 0.8$ ,  $\rho = 0.5$ ,  $\alpha = 2$ ,  $\beta = 1$ , and  $\tau_0 = 1/(N \times L_{nn})$ , where  $L_{nn}$  is the tour length found by the nearest neighbor heuristic. Local search parameters are initial temperature  $t_0 = 2$ , final temperature  $t_f = 0.01$ ,  $R = \max\{N \times K/2, 250\}$ ,  $\lambda = 0.9$ . Pheromone disturbance parameters are:  $\mu = 0.3$ ,  $w = N$ ,  $\Delta f = 5$ .

Table 5 lists the computational results of PACO on two sets of test problems. The best solution, average solution, worst solution, and standard deviation (Std.) computed over 20 independent runs on each problem are summarized, along with their average computational time (in seconds) required to reach the final best solutions. The best solutions equal to the best-known solutions of benchmark problems are asterisked and typed in bold. Table 5 reveals that PACO is able to generate reasonable good solutions for most of CVRPs in terms of solution quality. Twelve out of sixteen test problems can be solved successfully by the proposed algorithm in the first benchmark set. For the second set, seven out of fourteen test problems can be solved successfully by PACO.

To evaluate the pheromone disturbance strategy and  $P_{\text{best}}$ -resetting operation, the proposed algorithm is compared with standard ACO [9] and PS-ACO [22] in terms of their convergence trends. Standard ACO and PS-ACO were originally proposed for solving traveling salesman problems (TSP), not CVRPs. We implemented these two algorithms in Java and tried to apply them to solve CVRPs with the same local search method and parameter settings used in PACO.

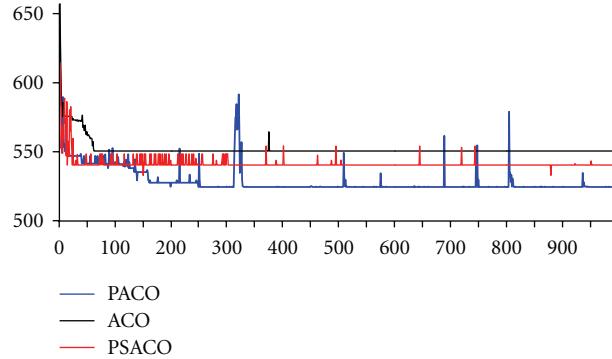
Benchmark problem C1 was selected to test three ACO-based algorithms. Three data sets of iteration-best solutions are plotted in Figure 2, The curves reveal that both PACO and PS-ACO can converge faster than standard ACO but only PACO can find the optimal solution. It demonstrates that, with pheromone disturbance and  $P_{\text{best}}$  solution resetting, PACO can effectively escape from local optima and find better solutions. In Figure 2, a

**Table 5:** Computational results of PACO over 20 runs on two benchmark sets.

	First benchmark set					Second benchmark set					
	Best	Avg.	Worst	Std.	AT (s)	Best	Avg.	Worst	Std.	AT (s)	
A-n33-k5	<b>661*</b>	661	661	0	0.87	C1	<b>524.61*</b>	527.54	543.16	4.98	32.3
A-n46-k7	<b>914*</b>	914	914	0	6.02	C2	<b>835.26*</b>	842.71	852.14	4.31	107.91
A-n60-k9	<b>1354*</b>	1356.4	1369	4.49	52.88	C3	829.92	838.29	844.65	3.76	141.89
B-n35-k5	<b>955*</b>	955	955	0	2.65	C4	1040.23	1053.22	1077.68	9.31	377.83
B-n45-k5	<b>751*</b>	751	751	0	5.85	C5	1348.73	1375.05	1392.93	12.73	1048.45
B-n68-k9	1275	1286.25	1288	2.86	62.97	C6	<b>555.43*</b>	556.72	560.24	1.09	27.09
B-n78-k10	<b>1221*</b>	1228.1	1252	8.6	98.78	C7	<b>909.68*</b>	917.93	932.06	7	98.70
E-n30-k3	<b>534*</b>	534	534	0	4.38	C8	868.61	880.47	895.7	8.37	117.74
E-n51-k5	<b>521*</b>	522.65	528	2.87	19.46	C9	1171.94	1194.96	1233.9	15.24	505.89
E-n76-k7	685	691.15	694	2.35	46.85	C10	1454.81	1498.23	1577.52	28.8	939.08
F-n72-k4	<b>237*</b>	237	237	0	30.64	C11	<b>1042.11*</b>	1045.01	1049.45	2.87	196.49
F-n135-k7	1170	1193.45	1229	15.99	248.77	C12	<b>819.56*</b>	821.55	825.95	1.91	148.67
M-n101-k10	<b>820*</b>	822.9	824	1.41	113.28	C13	1562.64	1575.55	1596.63	8.16	320.92
M-n121-k7	<b>1034*</b>	1039.6	1127	20.12	80.62	C14	<b>866.37*</b>	866.81	867.77	0.51	173.15
P-n76-K4	<b>593*</b>	597.95	616	5.72	53.48						
P-n101-k4	683	693.35	706	7.21	64.92						

\*The solution equals the best-known solution.

AT denotes the average CPU time required to reach the final best solution over 20 runs.

**Figure 2:** Convergence trends of three ACO-based algorithms tested on problem C1.

couple of peaks on the PACO curve indicate the effects of pheromone disturbance and  $P_{best}$ -resetting. The disadvantage of PS-ACO is that all of the local best solutions are considered in the pheromone updating procedure. When most ants have similar  $P_{best}$  solutions, the total amount of pheromone increment on edges will become very large. It results in the state of pheromone stagnation and the search is trapped by a local optimum.

We conducted a comparative study to compare PACO with a couple of swarm intelligence methods available for the CVRP. The comparative study contains two parts. In the first part, the PACO algorithm is compared with three different PSO-based algorithms, which were all tested on the 16 problems from the first benchmark set. The smaller the objective function value, the better the solution. Table 6 displays the computational results of PACO and the best results found in the papers of Chen et al. [18], Ai and Kachitvichyanukul [20], and Kao and Chen [19], denoted as DPSO-SA, SR-2, CPSO-SA, respectively. Experimental

**Table 6:** Comparison of PACO with three PSO-based algorithms.

No.	Problem	<i>N</i>	<i>K</i>	BKS	DPSO-SA [18]	SR-2 [20]	CPSO-SA [19]	PACO
1	A-n33-k5	32	5	661	661*/32.2	661*/13	661*/0.7	661*/0.12
2	A-n46-k7	45	7	914	914*/128.9	914*/23	917/2.4	914*/0.16
3	A-n60-k9	59	9	1354	1354*/308.8	1355/40	1354*/6.5	1354*/14.15
4	B-n35-k5	34	5	955	955*/37.6	955*/14	955*/1.2	955*/0.06
5	B-n45-k5	44	5	751	751*/134.2	751*/20	751*/4.8	751*/1.12
6	B-n68-k9	67	9	1272	1272*/344.2	1274/50	1274/27.2	1275/87.19
7	B-n78-k10	77	10	1221	1239/429.4	1223/64	1237/24	1221*/54.38
8	E-n30-k3	29	3	534	534*/28.4	534*/16	534*/0.3	534*/0.05
9	E-n51-k5	50	5	521	528/300.5	521*/22	521*/4.6	521*/0.51
10	E-n76-k7	75	7	682	688/526.5	682*/60	692/9.5	685/18.95
11	F-n72-k4	71	4	237	244/398.3	237*/53	237*/5.3	237*/6.28
12	F-n135-k7	134	7	1162	1215/1526.3	1162*/258	1200/202.8	1170/246.85
13	M-n101-k10	100	10	820	824/874.2	820*/114	825/6.1	820*/66.02
14	M-n121-k7	120	7	1034	1038/1733.5	1036/89	1039/51.5	1034*/8.18
15	P-n76-K4	75	4	593	602/496.3	594/48	596/27.6	593*/8.61
16	P-n101-k4	100	4	681	694/977.5	683/86	691/29.4	683/25.7

Notes:  $x/y = f(\text{best solution})/\text{shortest CPU Time (s)}$ , BKS is the best-known solution provided by published papers, DPSO-SA used Intel Pentium IV CPU 1.8 GHz with 256 M RAM, SR-2 used Intel Pentium IV CPU 3.4 GHz with 1 GB RAM, and CPSO-SA used Intel Core 2 CPU E8400 3 GHz with 3.5 G RAM.

**Table 7:** Comparison of PACO with three ACO-based algorithms.

Prob.	<i>N</i>	<i>K</i>	BKS	IACS [12]	SS_ACO [13]	EACO [15]	PACO
C1	50	5	524.61	524.61*/3	524.61*/55.23	524.61*/—	524.61*/0.79
C2	75	10	835.26	836.18/26	835.26*/70.43	835.26*/—	835.26*/163.44
C3	100	8	826.14	835.6/101	830.14/120.25	826.14*/—	829.92/183.08
C4	150	12	1028.42	1038.22/617	1038.20/250.76	1041.83/—	1040.23/547.88
C5	199	17	1291.45	1327.07/3080	1307.18/707.80	1338.48/—	1348.73/1455.64
C6	50	6	555.43	555.43*/5	559.12/65.17	555.43*/—	555.43*/7.72
C7	75	11	909.68	909.68*/41	912.68/90.42	909.68*/—	909.68*/66.68
C8	100	9	865.94	865.94*/115	869.34/210.46	865.94*/—	868.61/146.5
C9	150	14	1162.55	1173.76/853	1179.4/520.52	1168.81/—	1171.94/614.53
C10	199	18	1395.85	1413.83/4223	1410.26/1012.23	1413.69/—	1454.81/1114.21
C11	120	7	1042.11	1042.11*/204	1044.12/232.46	1045.5/—	1042.11*/95.33
C12	100	10	819.56	832.67/88	824.31/156.47	819.56*/—	819.56*/122.87
C13	120	11	1541.14	1547.07/428	1556.52/467.24	1554.93/—	1562.64/156.32
C14	100	11	866.37	866.37*/125	870.26/368.72	866.37*/—	866.37*/32.75

Notes:  $x/y = f(\text{best solution})/\text{shortest CPU Time (s)}$ , BKS is the best-known solution provided by published papers, — denotes that the data is not available, IACS used Intel Pentium III CPU 1000 MHz with 128 MB RAM, SS\_ACO used IBM computer CPU 1600 MHz with 512 MB RAM, and EACO used Pentium IV CPU 3.0 GHz.

results show that PACO is able to obtain the same or better results compared with three PSO-based algorithms.

In the second part of the comparative study, the computational results of PACO are compared with three ACO-based algorithms, which were all tested on the 14 problems from the second benchmark set. Table 7 reports the best results obtained by these various ACO algorithms. The best results of IACS, SS\_ACO, and EACO are found in [12, 13, 15],

respectively. Table 7 indicates that PACO generates very good solutions for seven problems, which are equal to the best solutions published so far. For three compared algorithms, only EACO is better than PACO because it can find best-known solutions for eight problems. It can be noticed that both PACO and EACO can obtain optimum solutions for six problems in common. For problem C11, PACO can reach the best-known solution but EACO cannot, while, for problems C3 and C8, EACO can outperform PACO. However, the best solutions produced by PACO and EACO are nearly equal for these three problems.

Tables 6 and 7 also list the shortest computational times required to reach the final best solution over 20 independent runs. From the data, it can be observed that the computation time taken by PACO is reasonable in practice in comparison with existing PSO- and ACO-based algorithms. It shows that pheromone disturbance can improve solutions but does not increase much computational time. Note that the computation time has not been reported in [15]. Instead, the authors of [15] used the maximum execution time as the termination condition. They stopped the EACO algorithm after one hour of running for 14 benchmark problems. The overall result of comparative study shows that the proposed algorithm is competitive with recent swarm intelligence approaches in terms of solution quality and CPU time.

## 5. Conclusion

This paper proposes a hybrid algorithm, PACO, which takes advantage of ant colony optimization and particle swarm optimization for capacitated vehicle problems. During the searching process, artificial ants construct solution routes, memorize the best solution ever found, and lay pheromone on the routes of swarm and personal best solutions. To prevent being trapped in local optima and to increase the probability of obtaining better solutions, PACO performs pheromone disturbance and short-term memory resetting operations to adjust stagnated pheromone trails. Disturbed pheromone trails guide ants to find new  $P_{best}$  and  $G_{best}$  solutions. The merits of PSO adopted in PACO can speed convergence during a run, even after pheromone disturbance operations. Computational results show that the performance of PACO is competitive in terms of solution quality when compared with existing ACO- and PSO-based approaches. For future research, PACO can be modified to extend its application to vehicle routing problems with time windows or multiple depots, among others.

## Acknowledgments

The authors would like to thank the referees for their valuable comments and suggestions that have greatly improved the quality of this paper. The authors are also grateful to National Science Council, Taiwan (Grant no. NSC 99-2410-H-036-003-MY2) for the financial support.

## References

- [1] G. Laporte, "The vehicle routing problem: an overview of exact and approximate algorithms," *European Journal of Operational Research*, vol. 59, no. 3, pp. 345–358, 1992.
- [2] I. H. Osman, "Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem," *Annals of Operations Research*, vol. 41, no. 4, pp. 421–451, 1993.

- [3] J. F. Cordeau, G. Laporte, M. W. P. Savelsbergh, and D. Vigo, "Vehicle routing," in *Handbook in Operations Research and Management Science*, C. Barnhart and G. Laporte, Eds., vol. 14, pp. 367–428, 2007.
- [4] J. K. Lenstra and A. H. G. Rinnooy Kan, "Complexity of vehicle routing and scheduling problems," *Networks*, vol. 11, no. 2, pp. 221–228, 1981.
- [5] G. Barbarosoglu and D. Ozgur, "A tabu search algorithm for the vehicle routing problem," *Computers and Operations Research*, vol. 26, no. 3, pp. 255–270, 1999.
- [6] B. M. Baker and M. A. Aye chew, "A genetic algorithm for the vehicle routing problem," *Computers and Operations Research*, vol. 30, no. 5, pp. 787–800, 2003.
- [7] S. W. Lin, Z. J. Lee, K. C. Ying, and C. Y. Lee, "Applying hybrid meta-heuristics for capacitated vehicle routing problem," *Expert Systems with Applications*, vol. 36, no. 2, pp. 1505–1512, 2009.
- [8] W. Y. Szeto, Y. Wu, and S. C. Ho, "An artificial bee colony algorithm for the capacitated vehicle routing problem," *European Journal of Operational Research*, vol. 215, no. 1, pp. 126–135, 2011.
- [9] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53–66, 1997.
- [10] J. E. Bell and P. R. McMullen, "Ant colony optimization techniques for the vehicle routing problem," *Advanced Engineering Informatics*, vol. 18, no. 1, pp. 41–48, 2004.
- [11] Z. Liu and Y. Cai, "Sweep based multiple ant colonies algorithm for capacitated vehicle routing problem," in *Proceedings of the IEEE International Conference on e-Business Engineering (ICEBE '05)*, pp. 387–394, October 2005.
- [12] C. H. Chen and C. J. Ting, "An improved ant colony system algorithm for the vehicle routing problem," *Journal of the Chinese Institute of Industrial Engineers*, vol. 23, no. 2, pp. 115–126, 2006.
- [13] X. Zhang and L. Tang, "A new hybrid ant colony optimization algorithm for the vehicle routing problem," *Pattern Recognition Letters*, vol. 30, no. 9, pp. 848–855, 2009.
- [14] B. Yu, Z. Z. Yang, and B. Yao, "An improved ant colony optimization for vehicle routing problem," *European Journal of Operational Research*, vol. 196, no. 1, pp. 171–176, 2009.
- [15] C. Y. Lee, Z. J. Lee, S. W. Lin, and K. C. Ying, "An enhanced ant colony optimization (EACO) applied to capacitated vehicle routing problem," *Applied Intelligence*, vol. 32, no. 1, pp. 88–95, 2010.
- [16] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [17] Y. Shi and R. Eberhart, "Modified particle swarm optimizer," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC'98)*, pp. 69–73, May 1998.
- [18] A. L. Chen, G. K. Yang, and Z. M. Wu, "Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem," *Journal of Zhejiang University: Science*, vol. 7, no. 4, pp. 607–614, 2006.
- [19] Y. Kao and M. Chen, "A hybrid PSO algorithm for the CVRP problem," in *Proceedings of the International Conference on Evolutionary Computation Theory and Applications (ECTA '11)*, pp. 539–543, Paris, France, October 2011.
- [20] T. J. Ai and V. Kachitvichyanukul, "Particle swarm optimization and two solution representations for solving the capacitated vehicle routing problem," *Computers and Industrial Engineering*, vol. 56, no. 1, pp. 380–387, 2009.
- [21] Y. Marinakis, M. Marinaki, and G. Dounias, "A hybrid particle swarm optimization algorithm for the vehicle routing problem," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 4, pp. 463–472, 2010.
- [22] B. Shuang, J. Chen, and Z. Li, "Study on hybrid PS-ACO algorithm," *Applied Intelligence*, vol. 34, no. 1, pp. 64–73, 2011.
- [23] N. Christofides, A. Mingozzi, and P. Toth, "The vehicle routing problem," in *Combinatorial Optimization*, N. Christofides, A. Mingozzi, P. Toth, and C. Sandi, Eds., Wiley, Chichester, UK, 1979.

*Research Article*

## New Bounds for Ternary Covering Arrays Using a Parallel Simulated Annealing

**Himer Avila-George,<sup>1</sup> Jose Torres-Jimenez,<sup>2</sup> and Vicente Hernández<sup>3</sup>**

<sup>1</sup> Instituto Tecnológico Superior de Salvatierra, Madero 303, 38900 Salvatierra, Guanajuato, Mexico

<sup>2</sup> Information Technology Laboratory, Cinvestav Tamaulipas, Km. 5.5 Carretera Victoria-Soto La Marina, 87130 Victoria, TAMPS, Mexico

<sup>3</sup> Instituto de Instrumentación para Imagen Molecular (I3M), Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia, Spain

Correspondence should be addressed to Himer Avila-George, hiavgeo@posgrado.upv.es

Received 24 February 2012; Revised 15 June 2012; Accepted 7 July 2012

Academic Editor: John Gunnar Carlsson

Copyright © 2012 Himer Avila-George 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 covering array (CA) is a combinatorial structure specified as a matrix of  $N$  rows and  $k$  columns over an alphabet on  $v$  symbols such that for each set of  $t$  columns every  $t$ -tuple of symbols is covered at least once. Given the values of  $t$ ,  $k$ , and  $v$ , the optimal covering array construction problem (CAC) consists in constructing a CA  $(N; t, k, v)$  with the minimum possible value of  $N$ . There are several reported methods to attend the CAC problem, among them are direct methods, recursive methods, greedy methods, and metaheuristics methods. In this paper, There are three parallel approaches for simulated annealing: the independent, semi-independent, and cooperative searches are applied to the CAC problem. The empirical evidence supported by statistical analysis indicates that cooperative approach offers the best execution times and the same bounds as the independent and semi-independent approaches. Extensive experimentation was carried out, using 182 well-known benchmark instances of ternary covering arrays, for assessing its performance with respect to the best-known bounds reported previously. The results show that cooperative approach attains 134 new bounds and equals the solutions for other 29 instances.

### 1. Introduction

A covering array, denoted by  $\text{CA}(N; t, k, v)$ , is a matrix  $M$  of size  $N \times k$  which takes values from the set of symbols  $\{0, 1, 2, \dots, v - 1\}$  (called the alphabet), and every submatrix of size  $N \times t$  contains each tuple of symbols of size  $t$  (or  $t$ -tuple), at least once. The value  $N$  is the number of rows of  $M$ ,  $k$  is the number of parameters, where each parameter can take  $v$  values, and the interaction degree between parameters is described by the strength  $t$ . Each

combination of  $t$  columns must cover all the  $v^t$   $t$ -tuples. Given that there are  $\binom{k}{t}$  sets of  $t$  columns in  $M$ , the total number of  $t$ -tuples in  $M$  must be  $v^t \binom{k}{t}$ . When a  $t$ -tuple is missing in a specific set of  $t$  columns, we will refer to it as a missing  $t$ -wise combination. Then,  $M$  is a covering array if the number of missing  $t$ -wise combinations is zero.

When a matrix has the minimum possible value of  $N$  to be a CA( $N; t, k, v$ ), the value  $N$  is known as the covering array number. The covering array number is formally defined as  $\text{CAN}(t, k, v) = \min\{N : \exists \text{ CA}(N; t, k, v)\}$ . Given the values of  $t$ ,  $k$ , and  $v$ , the optimal covering array construction problem (CAC) consists in constructing a CA( $N; t, k, v$ ) such that the value of  $N$  is minimized.

A major application of covering arrays (CAs) arises in software interaction testing [1], where a covering array can be used to represent an interaction test suite as follows. In a software test, we have  $k$  components or factors. Each of these parameters has  $v$  values or levels. A test suite is an  $N \times k$  array where each row is a test case. Each column represents a component, and a value in the column is the particular configuration. By mapping a software test problem to a covering array of strength  $t$ , we can guarantee that we have tested, at least once, all  $t$ -way combinations of component values [2]. Thus, software testing costs can be substantially reduced by minimizing the number of test cases  $N$  in the covering array. Please observe that software interaction testing is a black-box testing technique, and thus it exhibits weaknesses that should be addressed by employing white-box testing techniques. For a detailed example of the use of covering arrays in software interaction testing, the reader is referred to [3].

In this paper, we aim to develop an enhanced sequential simulated annealing (ESSA) algorithm for finding near-optimal covering arrays. Simulated annealing algorithm is a general-purpose stochastic optimization technique that has proved to be an effective tool for approximating globally optimal solutions to many optimization problems. However, one of the major drawbacks of the technique is the time it requires to obtain good solutions (moreover, when the evaluation function requires too much time). To address this drawback, we propose three parallel simulated annealing approaches to solve the CAC problem. The objective is to find the best bounds to some ternary covering arrays by using parallelism. To our knowledge, the application of parallel simulated annealing to the CAC problem has not been reported in the literature. Some methods of parallelization of simulated annealing are discussed in [4–8].

The remainder of this paper is organized as follows. Some techniques that have been used for constructing covering arrays are presented in Section 2. Section 3 describes the components of our sequential annealing algorithm. In Section 4, three parallel simulated annealing approaches are discussed. Section 5 describes the experimental results. Finally, Section 6 presents the conclusions derived from the research presented in this paper.

## 2. Review of Covering Arrays Construction Methods

Because of the importance of the construction of (near) optimal covering arrays, much research has been carried out in developing effective methods for constructing them. There are several reported methods for constructing these combinatorial models. Among them are (1) direct methods; (2) recursive methods; (3) greedy methods; (4) metaheuristics methods.

Chateauneuf and Kreher [9] introduced a new method to construct covering arrays of strength three. This construction uses the structure of covering arrays and the repetition in covering arrays. The idea is to construct a covering array from a small array, a *starter vector*,

and a group. This construction builds the covering array column by column by considering the group acting on the columns of the starter vector. Meagher and Stevens [10] extended the idea exhibited in [9], presenting a strategy for obtaining the starter vector by local search and the selection of a group action. Recently, Lobb et al. [11] presented a generalization of this method to permit any number of fixed points, permit an arbitrary group acting on the symbols, and permit an arbitrary group acting on the columns. With all these generalizations were obtained new bounds for covering arrays of strength two.

Tang and Woo [12] used *constant weight vectors* to construct test suites to be applied to logic circuit testing. Martinez-Pena et al. [13] introduced a new method for constructing covering arrays using trinomial coefficients, and they improved the results presented in [12]. Martinez-Pena et al. used the trinomial coefficients for the representation of the search space in the construction of ternary covering arrays. It is clear that any covering array is formed by a row set. In this sense, a trinomial coefficient represents a particular subset of rows which may belong to a ternary covering array.

Hartman [3] presented a recursive construction which gives a method of squaring the number  $k$  of columns in a covering array of strength  $t$  while multiplying the rows  $N$  by a factor dependent only on  $t$  and  $v$ , but independent of  $k$ . This factor is related to the Turan numbers  $T(t; v)$  that are defined to be the number of edges in the Turan graph.

Colbourn et al. [14] presented a product construction for  $t = 2$ . In general, the product of two covering arrays where  $t = 2$  consists in obtaining a new covering array where the number of columns is equal to the product of the columns of the ingredients, and the number of rows is equal to the sum of the rows of each ingredient.

The majority of commercial and open-source test data generating tools use greedy algorithms for covering arrays construction (AETG, TCG, ACTS, DDA, among others).

Cohen et al. [1] presented a strategy called AETG. In AETG, covering arrays are constructed *one row at a time*. To generate a row, the first  $t$ -tuple is selected based on the one involved in most uncovered pairs. Remaining factors are assigned levels in a random order. Levels are selected based on the one that covers the most new  $t$ -tuples. For each row that is actually added to the covering array, there are a number of,  $M$ , candidate rows that are generated, and only a candidate that covers the most new  $t$ -tuples is added to the covering array. Once a covering is constructed, a number,  $R$ , of test suites are generated and the smallest test suite generated is reported. This process continues until all pairs are covered.

Tung and Aldiwan [15] proposed a tool called TCG. In TCG, one row is added at a time to a covering array until all pairs are covered. Before each row is added, a number of up to  $M$  candidate rows are generated, and the best candidate (covering the most new pairs) is added.  $M$  is defined to be the maximum cardinality of factors (the maximum number of levels associated with any factor). To construct each row, factors are assigned levels in an order based on a nonascending order of the cardinality of each factor. Each level for the factor is evaluated, and a count of the number of pairs that are covered is used to determine whether or not to select a level for a factor.

Bryce and Colbourn [16] presented an algorithm called DDA. The DDA constructs one row of a covering array at a time using a steepest ascent approach. Factors are dynamically fixed one at a time in an order based on density. New rows are continually added until all interactions have been covered.

Lei and Tai [17] introduced a new algorithm called IPO, for pairwise testing. For a system with two or more input parameters, the IPO strategy generates a pairwise test set for the first two parameters, extends the test set to generate a pairwise test set for the first three parameters, and continues to do so for each additional parameter. Contrary to many other

algorithms that build covering arrays *one row at a time*, the IPO strategy constructs them *one column at a time*. Lei et al. [18] introduced an algorithm for the efficient production of covering arrays, called IPOG, which generalizes the IPO strategy from pairwise testing to multiway testing. The main idea is that covering arrays of  $k - 1$  columns can be used to efficiently build a covering array with degree  $k$ .

Ronneseth and Colbourn [19] introduced a new algorithm for constructing covering arrays, the BBA. The BBA's fundamental idea is to combine smaller covering arrays by reordering the rows and then to append additional rows for the remaining uncovered pairs.

Some stochastic algorithms in artificial intelligence, such as *tabu search* [20, 21], *simulated annealing* [22], *generic algorithms*, and *ant colony optimization algorithm* [23], provide an effective way to find approximated solutions. In these algorithms, the optimization focuses on one value of  $N$  at a time, attempting to find a covering array for that size.

A simulated annealing metaheuristic (henceforth called SAC) has been applied by Cohen et al. in [22] for constructing covering arrays. SAC starts with a randomly generated initial solution  $M$  whose cost  $c(M)$  is measured as the number of uncovered  $t$ -tuples. In their implementation, Cohen et al. use a simple geometric function  $T_n = 0.9998T_{n-1}$  with an initial temperature fixed at  $T_i = 0.20$ . At each temperature, 2000 neighboring solutions are generated. The algorithm stops either if a valid covering array is found, or if no change in the cost of the current solution is observed after 500 trials. The authors justify their choice of these parameter values based on some experimental tuning. They conclude that their simulated annealing implementation is able to produce smaller covering arrays than other computational methods, sometimes improving upon algebraic constructions.

Torres-Jimenez and Rodriguez-Tello [24] introduced a new simulated annealing implementation for constructing binary covering arrays.

### 3. Sequential Simulated Annealing

In this section, we briefly review simulated annealing (SA) algorithm and propose an enhanced sequential simulated annealing (ESSA) implementation to solve the CAC problem. ESSA is an extension of the simulated annealing presented in [24] for constructing covering arrays for  $v > 2$  and mixed-level covering arrays.

Simulated annealing is a randomized local search method based on the simulation of annealing of metal. The acceptance probability of a trial solution is given by (3.1), where  $T$  is the *temperature* of the system,  $\Delta C$  is the difference of the costs between the trial and the current solutions (the cost change due to the perturbation), and (3.1) means that the trial solution is accepted by nonzero probability  $e^{-\Delta C/T}$  even though the solution deteriorates (*uphill move*)

$$(P) = \begin{cases} 1 & \text{if } \Delta C < 0, \\ e^{-\Delta C/T} & \text{otherwise.} \end{cases} \quad (3.1)$$

Uphill moves enable the system to escape from the local minima; without them, the system would be trapped into a local minimum. Too high of a probability for the occurrence of uphill moves, however, prevents the system from converging. In simulated annealing, the probability is controlled by temperature in such a manner that at the beginning of the procedure the temperature is sufficiently high, in which a high probability is available, and

as the calculation proceeds, the temperature is gradually decreased, lowering the probability [25].

The following paragraphs will describe each of the components of the implementation of our simulated annealing implementation. The description is done given the matrix representation of a covering array.

### 3.1. Internal Representation

Let  $M$  be a potential solution in the search space  $\mathcal{M}$ , that is, a covering array  $\text{CA}(N; t, k, v)$  of size  $N$ , strength  $t$ , degree  $k$ , and order  $v$ . Then  $M$  is represented as an  $N \times k$  array on  $v$  symbols, in which the element  $m_{i,j}$  denotes the symbol assigned in the test configuration  $i$  to the parameter  $j$ . The size of the search space  $\mathcal{M}$  is then given by

$$|\mathcal{M}| = v^{Nk}. \quad (3.2)$$

### 3.2. Initial Solution

The *initial solution*  $M$  is constructed by generating  $M$  as a matrix with maximum Hamming distance. The Hamming distance  $d(x, y)$  between two rows  $x, y \in M$  is the number of elements in which they differ. Let  $r_i$  be a row of the matrix  $M$ . To generate a random matrix  $M$  of maximum Hamming distance, the following steps are performed:

- (1) generate the first row  $r_1$  at random,
- (2) generate  $s$  rows  $c_1, c_2, \dots, c_s$  at random, which will be candidate rows,
- (3) select the candidate row  $c_i$  that maximizes the Hamming distance according to (3.3) and added to the  $i$ th row of the matrix  $M$ ,
- (4) repeat from step 2 until  $M$  is completed;

$$g(r_i) = \sum_{s=1}^{i-1} \sum_{v=1}^k d(m_{s,v}, m_{i,v}), \quad \text{where } d(m_{s,v}, m_{i,v}) = \begin{cases} 1 & \text{if } m_{s,v} \neq m_{i,v}, \\ 0 & \text{otherwise.} \end{cases} \quad (3.3)$$

An example is shown in (3.4); the number of symbols different between rows  $r_1$  and  $c_1$  is 4 and between  $r_2$  and  $c_1$  is 3 summing up 7. Then, the Hamming distance for the candidate row  $c_1$  is 7:

$$\text{Rows} \left\{ \begin{array}{l} r_1 = \{2 \ 1 \ 0 \ 1\} \\ r_2 = \{1 \ 2 \ 0 \ 1\} \\ c_1 = \{0 \ 2 \ 1 \ 0\} \end{array} \right. , \quad \text{Distances} \left\{ \begin{array}{l} d(r_1, c_1) = 4 \\ d(r_2, c_1) = 3 \\ g(c_1) = 7 \end{array} \right. , \quad (3.4)$$

Equation (3.4) is an example of the Hamming distance between two rows  $r_1, r_2$  that are already in the matrix  $M$  and a candidate row  $c_1$ .

### 3.3. Evaluations Function

The *evaluation function*  $C(M)$  is used to estimate the goodness of a candidate solution. Previously reported metaheuristic algorithms for constructing covering arrays have commonly evaluated the quality of a potential solution (covering array) as the number of combination of symbols missing in the matrix  $M$  [20, 22, 23]. Then, the expected solution will be zero missing. In the proposed simulated annealing implementation, this evaluation function definition was used. Its computational complexity is equivalent to  $O(N(\binom{k}{t}))$ .

### 3.4. Neighborhood Function

Given that ESSA is based on local search (LS), then a neighborhood function must be defined. The main objective of the neighborhood function is to identify the set of potential solutions which can be reached from the current solution in a local search (LS) algorithm. In case two or more neighborhoods exhibit complementary characteristics, it is then possible and interesting to create more powerful compound neighborhoods. The advantage of such an approach is well documented in [26]. Following this idea, and based on the results of our preliminary experimentations, a neighborhood structure composed by two different functions is proposed for ESSA.

The neighborhood function  $\mathcal{N}_1(s)$  makes a random search of a missing  $t$ -tuple and then tries by setting the  $j$ th combination of symbols in every row of  $M$ . The neighborhood function  $\mathcal{N}_2(s)$  randomly chooses a position  $(i, j)$  of the matrix  $M$  and makes all possible changes of symbol. During the search process, a combination of both  $\mathcal{N}_1(s)$  and  $\mathcal{N}_2(s)$  neighborhood functions is employed by ESSA algorithm. The former is applied with probability  $\mathbb{P}$ , while the latter is employed at a  $(1 - \mathbb{P})$  rate. This combined neighborhood function  $\mathcal{N}_3(s, x)$  is defined in (3.5), where  $x$  is a random number in the interval  $[0, 1]$ :

$$\mathcal{N}_3(s, x) = \begin{cases} \mathcal{N}_1(s) & \text{if } x \leq \mathbb{P}, \\ \mathcal{N}_2(s) & \text{if } x > \mathbb{P}. \end{cases} \quad (3.5)$$

### 3.5. Cooling Schedule

The *cooling schedule* determines the degree of uphill movement permitted during the search and is thus critical to the ESSA algorithm's performance. The parameters that define a cooling schedule are an initial temperature, a final temperature or a stopping criterion, the maximum number of neighboring solutions that can be generated at each temperature, and a rule for decrementing the temperature. The literature offers a number of different cooling schedules, see, for instance, [4, 27]. ESSA uses a geometrical cooling scheme mainly for its simplicity. It starts at an initial temperature  $T_i$  which is decremented at each round by a factor  $\alpha$  using the relation  $T_k = \alpha T_{k-1}$ . For each temperature, the maximum number of visited neighboring solutions is  $L$ . It depends directly on the parameters ( $N$ ,  $k$ , and  $v$ ) of the studied covering array. This is because more moves are required for covering arrays with alphabets of greater cardinality.

```

(1) INITIALIZE( $M, T, L$ ) /* Create the initial solution. */
(2)  $M^* \leftarrow M$ ; /* Memorize the best solution. */
(3)  $T = T_0$ ; /* Initial temperature of SA. */
(4) repeat
(5)   for  $i \leftarrow 1$  to  $L$  do
(6)      $M_i \leftarrow \text{GENERATE } (M)$ ; /* Perturb current state. */
(7)      $\Delta C \leftarrow C(M_i) - C(M)$ ; /* Evaluate cost function. */
(8)      $x \leftarrow \text{random}$ ; /*  $x$  in the range [0,1]. */
(9)     if  $\Delta C < 0$  or  $e^{-\Delta C/T} > x$  then
(10)        $M \leftarrow M_i$ ; /* Accept new state. */
(11)       if  $C(M) < C(M^*)$  then
(12)          $M^* \leftarrow M$ ; /* Memorize the best solution. */
(13)       end if
(14)     end if
(15)   end for
(16)   CALCULATE.CONTROL( $T, \phi$ )
(17) until termination condition is satisfied;

```

**Algorithm 1:** Sequential simulated annealing for the CAC problem.

### 3.6. Termination Condition

The *stop criterion* for ESSA is either when the current temperature reaches  $T_f$ , when it ceases to make progress, or when a valid covering array is found. In the proposed implementation, a lack of progress exists if after  $\phi$  (frozen factor) consecutive temperature decrements the best-so-far solution is not improved.

### 3.7. Simulated Annealing Pseudocode

Algorithm 1 presents the simulated annealing heuristic as described above. The meaning of the three functions is obvious: INITIALIZE computes a start solution and initial values of the parameters  $T$  and  $L$ ; GENERATE selects a solution from the neighborhood of the current solution, using the neighborhood function  $\mathcal{N}_3(s, x)$ ; CALCULATE.CONTROL computes a new value for the parameter  $T$  (cooling schedule) and the number of consecutive temperature decrements with no improvement in the solution.

This is unlike the classical method which takes as a solution to the problem the last value obtained in the annealing chain [28]. We memorize the best solution found during the whole annealing process (see lines 2 and 12).

In the next section, it is presented three parallel simulated annealing approaches for solving the CAC problem.

## 4. Parallel Simulated Annealing

Parallelization is recognized like a powerful strategy to increase algorithms efficiency; however, simulated annealing parallelization is a hard task because it is essentially a sequential process.

In evaluating performance of a parallel simulated annealing (PSA), it needs to consider solution quality as well as execution speed. The execution speed may be quantified in terms of *speed-up* ( $\mathcal{S}$ ) and *efficiency* ( $\mathcal{E}$ ). The  $\mathcal{S}$  is defined as the ratio of the execution time (on one processor) by the sequential simulated annealing to that by the PSA (on  $P$  processors) for an equivalent solution quality. In the ideal case,  $\mathcal{S}$  would be equal to  $P$ . The  $\mathcal{E}$  is defined as the ratio of the actual  $\mathcal{S}$  to the ideal  $\mathcal{S}(P)$ .

Next, we propose three parallel implementations of the simulated annealing algorithm described in Section 3. For these cases, let  $P$  denote the number of processors and  $L$  the length of Markov chain.

#### **4.1. Independent Search Approach**

A common approach to parallelizing simulated annealing is the ISA [4, 6, 29]. In this approach, each processor independently perturbs the configuration, evaluates the cost, and decides on the perturbation. The processors  $P_i$ ,  $i = 0, 1, \dots, P - 1$  carry out the independent annealing searches using the same initial solution and cooling schedule as in the sequential algorithm. At each temperature,  $P_i$  executes  $N \times k \times v^2$  annealing steps. When each processor finishes, it sends its results to processor  $P_0$ . Finally, processor  $P_0$  chooses the final solution among the local solutions.

We have implemented a simulated annealing algorithm using ISA approach for constructing covering arrays. In the developed implementation, the processors do not interact during individual annealing processes until all processors find their final solution. Then, the best of the solutions is saved and the others are discarded.

#### **4.2. Semi-Independent Search Approach**

Aarts and van Laarhoven [4] introduced a new parallel simulated annealing algorithm named *division algorithm*. In the division algorithm, the number of iterations at each temperature is divided equally between the processors. After a change in temperature, each processor may simply start from the final solution obtained by that processor at the previous temperature. The best solution from all the processors is then taken to be the final solution. Another variant of this approach is to communicate the best solution from all the processors to each processor every time the temperature changes. Aarts and van Laarhoven [4] found no significant differences in the performance of these two variants.

We have developed an implementation of division algorithm; we named the implementation SSA. In SSA, parallelism is obtained by dividing the effort of generation a Markov chain over the available processors. A Markov chain is divided into  $P$  subchains of the length  $[L/P]$ . In this approach, the processors exchange local information including intermediate solutions and their costs. Then, each processor restarts from the best intermediate ones.

Compared to the ISA, communication overhead in this SSA approach would be increased. However, each processor can utilize the information from other processors such that the decrease in computations and idle times can be greater than the increase in communication overhead. For instance, a certain processor which is trapped in an inferior solution can recognize its state by comparing it with others and may accelerate the annealing procedure. That is, processors may collectively converge to a better solution.

### 4.3. Cooperative Search Approach

In order to improve the performance of the SSA approach, we propose the cooperative search approach (CSA); it used asynchronous communication among processors accessing the global state to eliminate the idle times. Each processor follows a separate search path, accesses the global state which consists of the current best solution and its cost whenever it finished a Markov subchain, and updates the state if necessary. Once a processor gets the global state, it proceeds to the next Markov subchain with any delay.

Unlike SSA, CSA has the following characteristics:

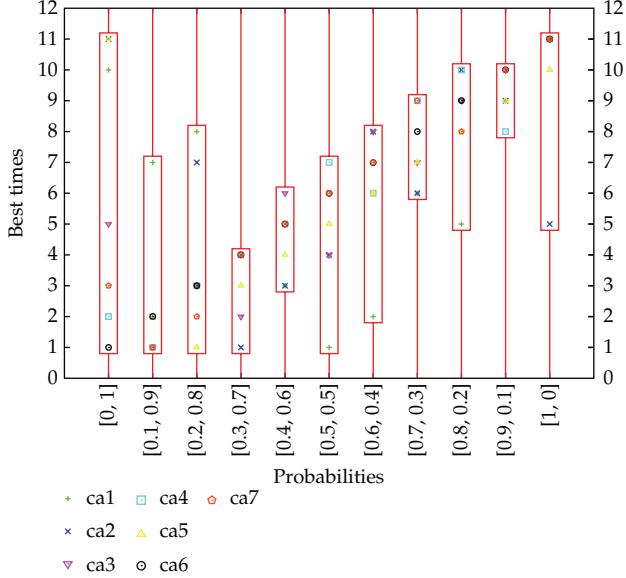
- (i) idle times can be reduced since asynchronous communications overlap a part of the computation;
- (ii) less communication overhead, an isolated access to the global state is needed by each processor at the end of each Markov subchain;
- (iii) the probability of being trapped in a local optimum can be smaller. This is because not all the processors start from the same state in each Markov subchain.

## 5. Experimental Results

This section presents an experimental design and results derived from testing the parallel ISA, SSA, and CSA algorithms described in the Section 4. In order to show the performance of these approaches, three experiments were developed. The first experiment had as purpose to fine-tune the probabilities of the neighborhood functions to be selected. The second experiment had as purpose to compare the three approaches in terms of parallel execution time. Among the three approaches, the CSA approach was the fastest. The third experiment evaluated the quality of the solutions of the CSA approach over a new benchmark proposed in this paper. The results were compared against the best-known solutions reported in the literature to construct covering arrays [30].

The three parallel approaches were implemented using C language and the message passing interface (MPI) library. The implementations were run on the *Tirant supercomputer* (The Tirant supercomputer: <http://www.uv.es/siuv/cas/zcalculo/res/informa.wiki>). Tirant comprises 256 JS20 compute nodes (blades) and 5 p515 servers. Every blade has two IBM Power4 processors at 2.0 GHz running Linux operating system with 4 GB of memory RAM and 36 GB of local disk storage. All the servers provide a total of nearly 10 TB of disk storage accessible from every blade through GPFS (global parallel file system). Tirant has in total 512 processors, 1 TB of memory RAM, and 19 TB of disk storage. The following parameters were used for all simulated annealing implementations:

- (1) Initial temperature  $T_i = 4.0$ ,
- (2) Final temperature  $T_f = 1.0E - 10$ ,
- (3) Cooling factor  $\alpha = 0.99$ ,
- (4) Maximum neighboring solutions per temperature  $L = N \times k \times v^2$ ,
- (5) Frozen factor  $\phi = 11$ ;
- (6) The neighborhood function  $\mathcal{N}_3(s, x)$  is applied using a probability  $\mathbb{P} = 0.3$ .



**Figure 1:** Performance of ESSA with the 11 combinations of probabilities.

### 5.1. Tuning of the Parameters of Parallel Simulated Annealing

It is well known that the performance of a simulated annealing algorithm is sensitive to parameter tuning. In this sense, we follow a methodology for a fine-tuning of the two neighborhood functions used in ESSA. The fine-tuning was based on the linear Diophantine equation (5.1), where  $x_i$  represents a neighborhood function, and its value set to 1,  $\mathbb{P}_i$  is a value in  $\{0.0, 0.1, \dots, 1.0\}$  that represents the probability of executing  $x_i$ , and  $q$  is set to 1.0 which is the maximum probability of executing any  $x_i$ :

$$\mathbb{P}_1 x_1 + \mathbb{P}_2 x_2 = q. \quad (5.1)$$

A solution to the given linear Diophantine equation must satisfy (5.2). This equation has 11 solutions, each solution is an experiment that tests the degree of participation of each neighborhood function in ESSA to accomplish the construction of a covering array. Every combination of the probabilities was applied by ESSA to construct the set of covering arrays shows in Table I(a), and each experiment was run 31 times; with the data obtained for each experiment, we calculate the median. A summary of the performance of ESSA with the probabilities that solved the 100% of the runs is shown in Table I(b):

$$\sum_{i=1}^2 \mathbb{P}_i x_i = 1.0 \quad (5.2)$$

Finally, given the results shown in Figure 1, the best configuration of probabilities was  $\mathbb{P}_1 = 0.3$  and  $\mathbb{P}_2 = 0.7$  because it found the covering arrays in smaller time (median value). The values  $\mathbb{P}_1 = 0.3$  and  $\mathbb{P}_2 = 0.7$  were kept fixed in the second experiment.

**Table 1:** (a) A set of 7 covering arrays configurations. (b) Performance of ESSA. Columns 3–9 show the time taken to construct each of the covering arrays.

		(a)						
		Covering arrays						
$\mathbb{P}_1$	$\mathbb{P}_2$	ca <sub>1</sub>	ca <sub>2</sub>	ca <sub>3</sub>	ca <sub>4</sub>	ca <sub>5</sub>	ca <sub>6</sub>	ca <sub>7</sub>
0	1	4789.763	3.072	46.989	12.544	3700.038	167.901	0.102
0.1	0.9	1024.635	0.098	0.299	0.236	344.341	3.583	0.008
0.2	0.8	182.479	0.254	0.184	0.241	173.752	1.904	0.016
0.3	0.7	224.786	0.137	0.119	0.222	42.950	1.713	0.020
0.4	0.6	563.857	0.177	0.123	0.186	92.616	3.351	0.020
0.5	0.5	378.399	0.115	0.233	0.260	40.443	1.258	0.035
0.6	0.4	272.056	0.153	0.136	0.178	69.311	2.524	0.033
0.7	0.3	651.585	0.124	0.188	0.238	94.553	2.127	0.033
0.8	0.2	103.399	0.156	0.267	0.314	81.611	5.469	0.042
0.9	0.1	131.483	0.274	0.353	0.549	76.379	4.967	0.110
1	0	7623.546	15.905	18.285	23.927	1507.369	289.104	2.297

## 5.2. Comparison of the ISA, SSA, and CSA Approaches

To test the performance of the ISA, SSA, and CSA approaches, we propose the construction of a covering array with  $N = 80$ ,  $t = 3$ ,  $k = 22$ , and  $v = 3$ . Each approach was executed 31 times (for providing statistical validity to experiment) using  $P = \{4, 8, 16, 32\}$ .

The performance of the algorithms has been compared based on median speedup as a function of the number of processors; the results are shown in Figure 2.

The ISA approach; had difficulty in handling the large problem instances, it does not scale well. The SSA approach provides reasonable results; however, because it is a synchronous algorithm, the idle and communication times are inevitable. The CSA approach is the one which offers the best results; it reduces the execution time of the SSA approach by employing asynchronous information exchange.

In the next subsection, it is presented the third experiment of this work, and the purpose is to measure the performance of the CSA algorithm against the best-known solutions reported in the literature.

## 5.3. Comparison with State-of-the-Art Procedures

The purpose of this experiment is to carry out a performance comparison of the bounds achieved by the CSA approach with respect to the best-known solutions reported in

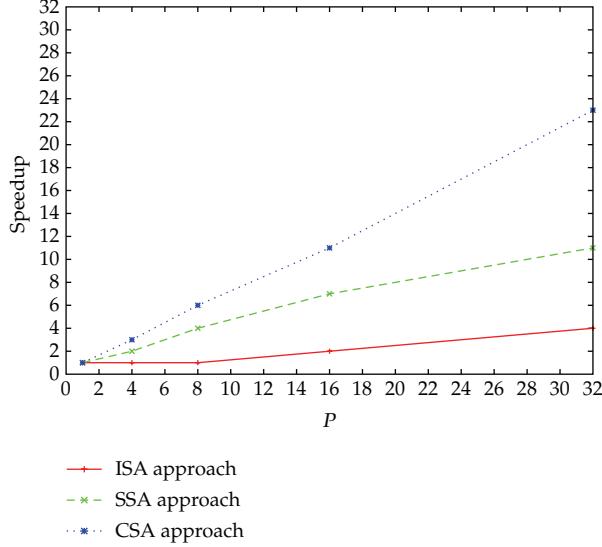


Figure 2: Median speedup for the ISA, SSA, and CSA approaches.

the literature [30], which were produced using the following state-of-the-art procedures: orthogonal array construction, Roux type constructions, doubling constructions, algebraic constructions, deterministic density algorithm (DDA), Tabu search, and IPOG-F.

For this experiment, we have fixed the maximum computational time expended by our PSA for constructing a covering array to 72 hours and 50 processors. We create a new benchmark composed of 182 covering arrays distributed as follows:

- (i) 47 covering arrays with strength  $t = 3$ , degree  $4 \leq k < 50$ , and order  $v = 3$ ,
- (ii) 46 covering arrays with strength  $t = 4$ , degree  $5 \leq k < 50$ , and order  $v = 3$ ,
- (iii) 45 covering arrays with strength  $t = 5$ , degree  $6 \leq k < 50$ , and order  $v = 3$ ,
- (iv) 44 covering arrays with strength  $t = 6$ , degree  $7 \leq k < 50$ , and order  $v = 3$ ,

The detailed results produced by this experiment are listed in Table 2. The first two columns in each subtable indicate the strength  $t$  and the degree  $k$  of the selected instances. Next two columns show, in terms of the size  $N$  of the covering arrays, the best-known solution reported in the literature and the improved bounds produced by the CSA approach. Last column depicts the difference between the best result produced by CSA approach and the best-known solution ( $\Delta = \beta - \vartheta$ ).

Figure 3 compares the results shown in Table 2 involving the CSA algorithm and the best-known solutions. The analysis of the data presented led us to the following observation. The solutions quality attained by the CSA approach is very competitive with respect to that produced by the state-of-the-art procedures summarized in column 3 ( $\vartheta$ ). In fact, it is able to improve on 134 previous best-known solutions.

## 6. Conclusions

The long execution time of simulated annealing due to its sequential nature hinders its application to realistically sized problems, in this case, the CAC problem when  $t > 3$ ,

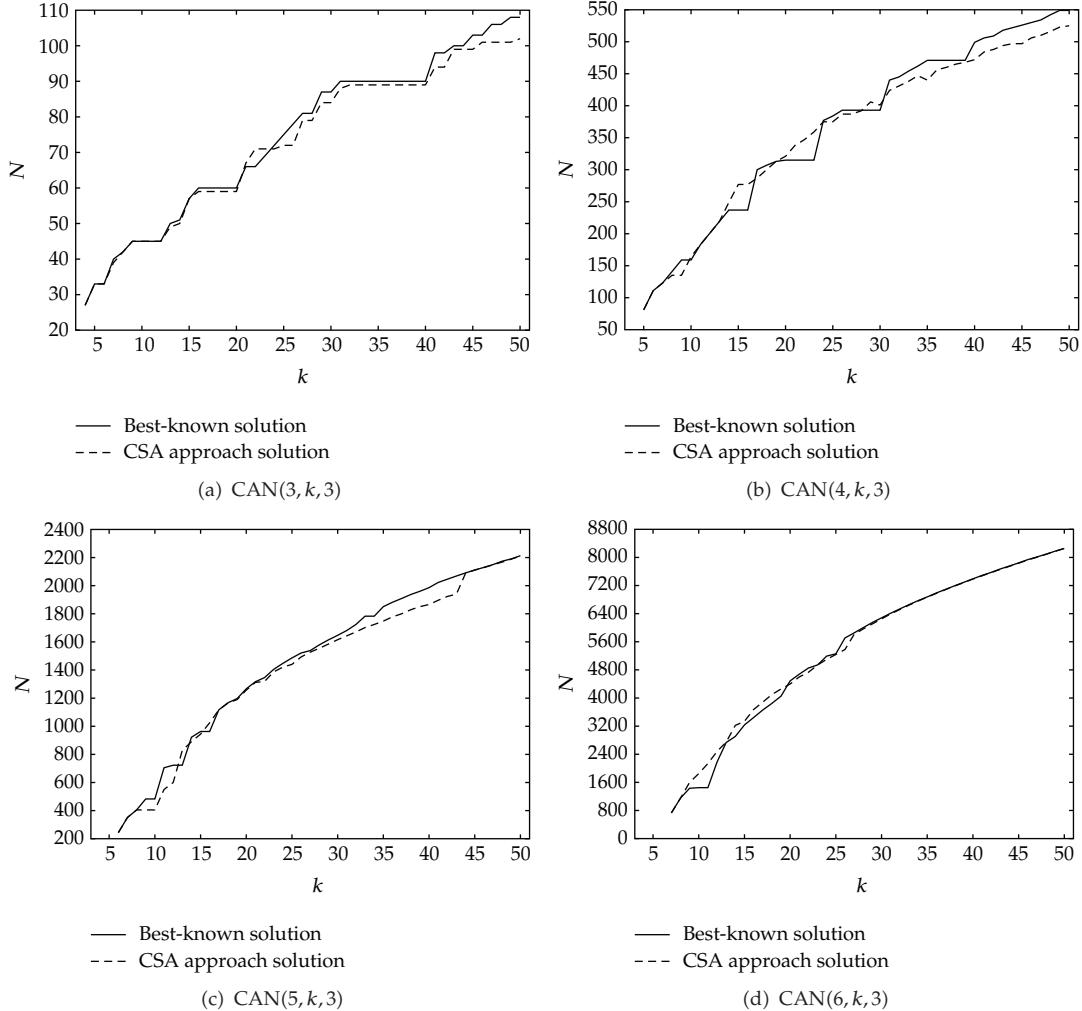
**Table 2:** It shows the improved bounds produced by CSA approach. Column  $\vartheta$  represents the best-known solution reported in the literature [30]. Column  $\beta$  represents the best solution in terms of  $N$  produced by CSA approach. Last column ( $\Delta$ ) depicts the difference between the best result produced by CSA approach and the best-known solution ( $\Delta = \beta - \vartheta$ ). (a) Improved bounds on CAN(3,  $k$ , 3); (b) improved bounds on CAN(4,  $k$ , 3); (c) improved bounds on CAN(5,  $k$ , 3); (d) improved bounds on CAN(6,  $k$ , 3).

(a)					
$t$	$k$	$\vartheta$	$\beta$	$\Delta$	
3	4	27	27	0	
3	5	33	33	0	
3	6	33	33	0	
3	7	40	<b>39</b>	-1	
3	8	42	42	0	
3	9	45	45	0	
3	10	45	45	0	
3	11	45	45	0	
3	12	45	45	0	
3	13	50	<b>49</b>	-1	
3	14	51	<b>50</b>	-1	
3	15	57	57	0	
3	16	60	<b>59</b>	-1	
3	17	60	<b>59</b>	-1	
3	18	60	<b>59</b>	-1	
3	19	60	<b>59</b>	-1	
3	20	60	<b>59</b>	-1	
3	21	66	67	1	
3	22	66	71	5	
3	23	69	71	2	
3	24	72	<b>71</b>	-1	
3	25	75	<b>72</b>	-3	
3	26	78	<b>72</b>	-6	
3	27	81	<b>79</b>	-2	
3	28	81	<b>79</b>	-2	
3	29	87	<b>84</b>	-3	
3	30	87	<b>84</b>	-3	
3	31	90	88	-2	
3	32	90	<b>89</b>	-1	
3	33	90	<b>89</b>	-1	
3	34	90	<b>89</b>	-1	
3	35	90	<b>89</b>	-1	
3	36	90	<b>89</b>	-1	
3	37	90	<b>89</b>	-1	
3	38	90	<b>89</b>	-1	
3	39	90	<b>89</b>	-1	
3	40	90	<b>89</b>	-1	
3	41	98	<b>94</b>	-4	
3	42	98	<b>94</b>	-4	
3	43	100	<b>99</b>	-1	
3	44	100	<b>99</b>	-1	
3	45	103	<b>99</b>	-4	
3	46	103	<b>101</b>	-2	
3	47	106	<b>101</b>	-5	
3	48	106	<b>101</b>	-5	
3	49	108	<b>101</b>	-7	
3	50	108	<b>102</b>	-6	

(b)				
$t$	$k$	$\vartheta$	$\beta$	$\Delta$
4	5	81	81	0
4	6	111	111	0
4	7	123	123	0
4	8	141	<b>135</b>	-6
4	9	159	<b>135</b>	-24
4	10	159	164	5
4	11	183	183	0
4	12	201	201	0
4	13	219	219	0
4	14	237	249	12
4	15	237	277	40
4	16	237	277	40
4	17	300	<b>287</b>	-13
4	18	307	<b>300</b>	-7
4	19	313	313	0
4	20	315	321	6
4	21	315	338	23
4	22	315	347	32
4	23	315	359	44
4	24	377	<b>375</b>	-2
4	25	384	<b>375</b>	-9
4	26	393	<b>387</b>	-6
4	27	393	<b>387</b>	-6
4	28	393	<b>392</b>	-1
4	29	393	406	13
4	30	393	401	8
4	31	440	<b>424</b>	-16
4	32	445	<b>431</b>	-14
4	33	454	<b>438</b>	-16
4	34	462	<b>447</b>	-15
4	35	471	<b>440</b>	-31
4	36	471	<b>456</b>	-15
4	37	471	<b>460</b>	-11
4	38	471	<b>465</b>	-6
4	39	471	<b>468</b>	-3
4	40	499	<b>472</b>	-27
4	41	506	<b>484</b>	-22
4	42	509	<b>488</b>	-21
4	43	518	<b>494</b>	-24
4	44	522	<b>497</b>	-25
4	45	526	<b>497</b>	-29
4	46	530	<b>506</b>	-24
4	47	534	<b>510</b>	-24
4	48	542	<b>516</b>	-26
4	49	549	<b>523</b>	-26
4	50	549	<b>525</b>	-24

(c)					
$t$	$k$	$\vartheta$	$\beta$	$\Delta$	
5	6	243	243	0	
5	7	351	351	0	
5	8	405	405	0	
5	9	483	<b>405</b>	-78	
5	10	483	<b>405</b>	-78	
5	11	705	<b>550</b>	-155	
5	12	723	<b>600</b>	-123	
5	13	723	828	105	
5	14	922	<b>890</b>	-32	
5	15	963	<b>944</b>	-19	
5	16	963	1025	62	
5	17	1117	1117	0	
5	18	1167	<b>1165</b>	-2	
5	19	1197	<b>1190</b>	-7	
5	20	1266	<b>1257</b>	-9	
5	21	1317	<b>1312</b>	-5	
5	22	1346	<b>1319</b>	-27	
5	23	1405	<b>1387</b>	-18	
5	24	1447	<b>1420</b>	-27	
5	25	1486	<b>1440</b>	-46	
5	26	1521	<b>1493</b>	-28	
5	27	1538	<b>1527</b>	-11	
5	28	1579	<b>1555</b>	-24	
5	29	1615	<b>1585</b>	-30	
5	30	1647	<b>1616</b>	-31	
5	31	1681	<b>1643</b>	-38	
5	32	1724	<b>1671</b>	-53	
5	33	1783	<b>1702</b>	-81	
5	34	1783	<b>1724</b>	-59	
5	35	1851	<b>1748</b>	-103	
5	36	1882	<b>1778</b>	-104	
5	37	1909	<b>1800</b>	-109	
5	38	1937	<b>1829</b>	-108	
5	39	1960	<b>1851</b>	-109	
5	40	1986	<b>1866</b>	-120	
5	41	2023	<b>1896</b>	-127	
5	42	2046	<b>1923</b>	-123	
5	43	2069	<b>1940</b>	-129	
5	44	2091	<b>2089</b>	-2	
5	45	2112	<b>2111</b>	-1	
5	46	2130	<b>2129</b>	-1	
5	47	2150	<b>2149</b>	-1	
5	48	2174	<b>2168</b>	-6	
5	49	2191	<b>2189</b>	-2	
5	50	2213	<b>2211</b>	-2	

(d)				
<i>t</i>	<i>k</i>	$\vartheta$	$\beta$	$\Delta$
6	7	729	729	0
6	8	1152	1152	0
6	9	1431	1600	169
6	10	1449	1849	400
6	11	1449	2136	687
6	12	2181	2482	301
6	13	2734	2744	10
6	14	2907	3220	313
6	15	3234	3338	104
6	16	3443	3672	229
6	17	3658	3882	224
6	18	3846	4098	252
6	19	4054	4256	202
6	20	4486	<b>4400</b>	-86
6	21	4678	<b>4600</b>	-78
6	22	4853	<b>4732</b>	-121
6	23	4942	<b>4941</b>	-1
6	24	5193	<b>5100</b>	-93
6	25	5257	<b>5238</b>	-19
6	26	5709	<b>5380</b>	-329
6	27	5853	<b>5810</b>	-43
6	28	6003	<b>5965</b>	-38
6	29	6150	<b>6110</b>	-40
6	30	6281	<b>6250</b>	-31
6	31	6413	<b>6393</b>	-20
6	32	6535	<b>6518</b>	-17
6	33	6656	<b>6642</b>	-14
6	34	6772	<b>6760</b>	-12
6	35	6877	<b>6871</b>	-6
6	36	6989	<b>6978</b>	-11
6	37	7092	<b>7086</b>	-6
6	38	7194	<b>7187</b>	-7
6	39	7293	<b>7284</b>	-9
6	40	7391	<b>7385</b>	-6
6	41	7490	<b>7478</b>	-12
6	42	7574	<b>7569</b>	-5
6	43	7672	<b>7661</b>	-11
6	44	7757	<b>7748</b>	-9
6	45	7845	<b>7836</b>	-9
6	46	7938	<b>7928</b>	-10
6	47	8013	<b>8005</b>	-8
6	48	8092	<b>8089</b>	-3
6	49	8179	<b>8176</b>	-3
6	50	8256	<b>8253</b>	-3



**Figure 3:** Graphical comparison of the quality solutions between CSA and the state of the art [30].

$5 < k \leq 100$ , and  $v = 3$ . A more efficient way to reduce execution time and make the simulated annealing a more promising method is to parallelize sequential simulated annealing. It is a challenging task. In fact, there are many approaches that may be considered in parallelizing simulated annealing. However, an inappropriate strategy used will likely result in poor performance.

In this paper, we have used three different approaches to do the parallelization of the simulated annealing algorithm. From the experimental results, we found that the ISA approach has the worst performance, and it does not scale well. The SSA approach offers reasonable execution times; compared to the ISA, communication overhead in the SSA approach would be increased when the size of the problem grows. However, each processor can utilize the information from other processors such that the decrease in computations and idle times can be greater than the increase in communication overhead. For instance, a certain processor which is trapped in an inferior solution can recognize its state by comparing it

with others and may accelerate the annealing procedure. That is, processors may collectively converge to a better solution.

The CSA approach is the one which offers the best results; it significantly reduces the execution time of the SSA approach by employing asynchronous information exchange. The quality solutions attained by the CSA approach are very competitive with respect to that produced by the state-of-the-art procedures; in fact, it is able to improve on 134 previous best-known solutions and equals the solutions for other 29 instances.

These experimental results confirm the practical advantages of using CSA algorithm in the software testing area. It is a robust algorithm yielding smaller test suites than other representative state-of-the-art algorithms, which allows reducing software testing costs.

The new bounds are available in Cinvestav covering array repository (CAR), which is available under request at <http://www.tamps.cinvestav.mx/~jtj/authentication.php>. We have verified all covering arrays described in this paper using the tool described in [31].

## Acknowledgments

The authors thankfully acknowledge the computer resources and assistance provided by Spanish Supercomputing Network (TIRANT-UV). This research work was partially funded by the following projects: CONACyT 58554; Cálculo de Covering Arrays; 51623-Fondo Mixto CONACyT; Gobierno del Estado de Tamaulipas.

## References

- [1] D. M. Cohen, S. R. Dalal, J. Parelius, and G. C. Patton, "The combinatorial design approach to automatic test generation," *IEEE Software*, vol. 13, no. 5, pp. 83–88, 1996.
- [2] K. Z. Zamli, M. F. J. Klaib, M. I. Younis, N. A. M. Isa, and R. Abdullah, "Design and implementation of a t-way test data generation strategy with automated execution tool support," *Information Sciences*, vol. 181, no. 9, pp. 1741–1758, 2011.
- [3] A. Hartman, "Software and hardware testing using combinatorial covering suites," in *Graph Theory, Combinatorics and Algorithms*, M. C. Golumbic and I. B. A. Hartman, Eds., vol. 34 of *Operations Research/Computer Science Interfaces*, pp. 237–266, Springer, New York, NY, USA, 2005.
- [4] E. H. L. Aarts and P. J. M. van Laarhoven, "Statistical cooling: a general approach to combinatorial optimization problems," *Philips Journal of Research*, vol. 40, no. 4, pp. 193–226, 1985.
- [5] D. Janaki Ram, T. H. Sreenivas, and K. G. Subramaniam, "Parallel simulated annealing algorithms," *Journal of Parallel and Distributed Computing*, vol. 37, no. 2, pp. 207–212, 1996.
- [6] S. Y. Lee and K. G. Lee, "Synchronous and asynchronous parallel simulated annealing with multiple Markov chains," *IEEE Transactions on Parallel and Distributed Systems*, vol. 7, no. 10, pp. 993–1008, 1996.
- [7] D. J. Chen, C. Y. Lee, C. H. Park, and P. Mendes, "Parallelizing simulated annealing algorithms based on high-performance computer," *Journal of Global Optimization*, vol. 39, no. 2, pp. 261–289, 2007.
- [8] Z. J. Czech, W. Mikanik, and R. Skinderowicz, "Implementing a parallel simulated annealing algorithm," in *Proceedings of the 8th International Conference on Parallel Processing and Applied Mathematics (PPAM'10)*, vol. 6067 of *Lecture Notes in Computer Science*, pp. 146–155, Springer, Berlin, Germany, 2010.
- [9] M. Chateauneuf and D. L. Kreher, "On the state of strength-three covering arrays," *Journal of Combinatorial Designs*, vol. 10, no. 4, pp. 217–238, 2002.
- [10] K. Meagher and B. Stevens, "Group construction of covering arrays," *Journal of Combinatorial Designs*, vol. 13, no. 1, pp. 70–77, 2005.
- [11] J. R. Lobb, C. J. Colbourn, P. Danziger, B. Stevens, and J. Torres-Jimenez, "Cover starters for covering arrays of strength two," *Discrete Mathematics*, vol. 312, no. 5, pp. 943–956, 2012.
- [12] D. T. Tang and L. S. Woo, "Exhaustive test pattern generation with constant weight vectors," *IEEE Transactions on Computers*, vol. C-32, no. 12, pp. 1145–1150, 1983.
- [13] J. Martinez-Pena, J. Torres-Jimenez, N. Rangel-Valdez, and H. Avila-George, "A heuristic approach

- for constructing ternary covering arrays using trinomial coefficients," in *Proceedings of the 12th Ibero-American Conference on Artificial Intel-Ligence—IBERAMIA*, vol. 6433 of *Lecture Notes in Computer Science*, pp. 572–581, Springer, Berlin, Germany, 2010.
- [14] C. J. Colbourn, S. S. Martirosyan, G. L. Mullen, D. Shasha, G. B. Sherwood, and J. L. Yucas, "Products of mixed covering arrays of strength two," *Journal of Combinatorial Designs*, vol. 14, no. 2, pp. 124–138, 2006.
  - [15] Y. W. Tung and W. S. Aldiwan, "Automating test case generation for the new generation mission software system," in *Proceedings of the IEEE Aerospace Conference*, vol. 1, pp. 431–437, IEEE Press, March 2000.
  - [16] R. C. Bryce and C. J. Colbourn, "The density algorithm for pairwise interaction testing," *Software Testing Verification and Reliability*, vol. 17, no. 3, pp. 159–182, 2007.
  - [17] Y. Lei and K. C. Tai, "In-parameter-order: a test generation strategy for pair-wise testing," in *Proceedings of the 3rd IEEE International Symposium on High-Assurance Systems Engineering (HASE'98)*, pp. 254–261, IEEE Computer Society, 1998.
  - [18] Y. Lei, R. Kacker, D. R. Kuhn, V. Okun, and J. Lawrence, "IPOG: a general strategy for T-way software testing," in *Proceedings of the 14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'07)*, pp. 549–556, IEEE Computer Society, Tucson, Ariz, USA, March 2007.
  - [19] A. H. Ronneseth and C. J. Colbourn, "Merging covering arrays and compressing multiple sequence alignments," *Discrete Applied Mathematics*, vol. 157, no. 9, pp. 2177–2190, 2009.
  - [20] K. J. Nurmiela, "Upper bounds for covering arrays by tabu search," *Discrete Applied Mathematics*, vol. 138, no. 1-2, pp. 143–152, 2004.
  - [21] L. Gonzalez-Hernandez, N. Rangel-Valdez, and J. Torres-Jimenez, "Construction of mixed covering arrays of variable strength using a tabu search approach," in *Proceedings of the 4th International Conference on Combinatorial Optimization and Applications (COCOA'10)*, vol. 6508 of *Lecture Notes in Computer Science*, pp. 51–64, Springer, Berlin, Germany, 2010.
  - [22] M. B. Cohen, C. J. Colbourn, and A. C. H. Ling, "Augmenting simulated annealing to build interaction test suites," in *Proceedings of the 14th International Symposium on Software Reliability Engineering (ISSRE'03)*, pp. 394–405, IEEE Computer Society, 2003.
  - [23] T. Shiba, T. Tsuchiya, and T. Kikuno, "Using artificial life techniques to generate test cases for combinatorial testing," in *Proceedings of the 28th Annual International Computer Software and Applications Conference (COMPSAC'04)*, vol. 1, pp. 72–77, IEEE Computer Society, September 2004.
  - [24] J. Torres-Jimenez and E. Rodriguez-Tello, "New bounds for binary covering arrays using simulated annealing," *Information Sciences*, vol. 185, no. 1, pp. 137–152, 2012.
  - [25] Y. Jun and S. Mizuta, "Detailed analysis of uphill moves in temperature parallel simulated annealing and enhancement of exchange probabilities," *Complex Systems*, vol. 15, no. 4, pp. 349–358, 2005.
  - [26] L. Cavique, C. Rego, and I. Themido, "Subgraph ejection chains and tabu search for the crew scheduling problem," *Journal of the Operational Research Society*, vol. 50, no. 6, pp. 608–616, 1999.
  - [27] M. M. Atiqullah, "An efficient simple cooling schedule for simulated annealing," in *Proceedings of the International Conference on Computational Science and its Applications (ICCSA'04)*, vol. 3045 of *Lecture Notes in Computer Science*, pp. 396–404, Springer, Berlin, Germany, 2004.
  - [28] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.
  - [29] Z. Czech, "Three parallel algorithms for simulated annealing," in *Parallel Processing and Applied Mathematics*, vol. 2328 of *Lecture Notes in Computer Science*, pp. 210–217, Springer, Berlin, Germany, 2006.
  - [30] C. J. Colbourn, "Covering array tables for t=2, 3, 4, 5, 6," 2012, <http://www.public.asu.edu/~ccolbou/src/tabby/catable.html>.
  - [31] H. Avila-George, J. Torres-Jimenez, N. Rangel-Valdez, A. Carrión, and V. Hernández, "Supercomputing and a Grid computing on the verification of coveringarrays," *The Journal of Supercomputing*. In press.

*Research Article*

## A VNS Metaheuristic with Stochastic Steps for Max 3-Cut and Max 3-Section

**Ai-fan Ling<sup>1,2</sup>**

<sup>1</sup> Research Center of Security and Future, School of Finance, Jiangxi University of Finance and Economics, Nanchang 330013, China

<sup>2</sup> Key Laboratory of Management, Decision and Information Systems, Academy of Mathematics and Systems Science, CAS, Beijing 100190, China

Correspondence should be addressed to Ai-fan Ling, aifanling@yahoo.com.cn

Received 15 February 2012; Accepted 30 May 2012

Academic Editor: John Gunnar Carlsson

Copyright © 2012 Ai-fan Ling. 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 heuristic algorithm based on VNS is proposed to solve the Max 3-cut and Max 3-section problems. By establishing a neighborhood structure of the Max 3-cut problem, we propose a local search algorithm and a variable neighborhood global search algorithm with two stochastic search steps to obtain the global solution. We give some numerical results and comparisons with the well-known 0.836-approximate algorithm. Numerical results show that the proposed heuristic algorithm can obtain efficiently the high-quality solutions and has the better numerical performance than the 0.836-approximate algorithm for the NP-Hard Max 3-cut and Max 3-section problems.

### 1. Introduction

Given a graph  $G(V; E)$ , with nodes set  $V$  and edges set  $E$ , the Max 3-cut problem is to find a partition  $S_0 \subset V$ ,  $S_1 \subset V$  and  $S_2 \subset V$ , of the set  $V$ , such that  $S_0 \cup S_1 \cup S_2 = V$ ,  $S_i \cap S_j = \emptyset (i \neq j)$  and the sum of the weights on the edges connecting the different parts is maximized. Similar to the Max cut problem, the Max 3-cut problem has long been known to be NP complete [1], even for any un-weighted graphs [2], and has also applications in circuit layout design, statistical physics, and so on [3]. However, due to the complexity of this problem, its research progresses is much lower than that of the Max cut problem. Based on the semidefinite programming relaxation proposed by Goemans and Williamson [4], Frieze and Jerrum [5] obtained a 0.800217-approximation algorithm for the Max 3-Cut problem. Recently, Goemans and Williamson [6] and Zhang and Huang [7] improved Frieze and

Jerrum's 0.800217-approximation ratio to 0.836 using a complex semidefinite programming relaxation of the Max 3-cut problem.

For the purpose of our analysis, we first introduce some notations. We denote the complex conjugate of  $y = a + ib$  by  $\bar{y} = a - ib$ , where  $i = \sqrt{-1}$  is the pure image number and the real part and image part of a complex number by  $\text{Re}(\cdot)$  and  $\text{Im}(\cdot)$ , respectively. For an  $n$  dimensional complex vector  $\mathbf{y} \in \mathbb{C}^n$  written as bold letter and  $n$  dimensional complex matrix  $Y \in \mathbb{C}^{n \times n}$ , we write  $\mathbf{y}^*$  and  $Y^*$  to denote their conjugate and transpose. That is,  $\mathbf{y}^* = \bar{\mathbf{y}}^T$  and  $Y^* = \bar{Y}^T$ . The set of  $n$  dimensional real symmetric (semidefinite positive) matrices and the set of  $n$  dimensional complex Hermitian (semidefinite positive) matrices are denoted by  $\mathcal{S}_n(\mathcal{S}_n^+)$  and  $\mathcal{H}_n(\mathcal{H}_n^+)$ , respectively. We sometimes use  $A \succcurlyeq 0$  to show  $A \in \mathcal{S}_n^+$  (or  $A \in \mathcal{H}_n^+$ ). For any two complex vector  $\mathbf{u}, \mathbf{v} \in \mathbb{C}^n$ , we write  $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u} \cdot \mathbf{v} = \mathbf{u}^* \mathbf{v}$  as their inner product. For any two complex matrices  $A, B \in \mathcal{H}_n$ , we write  $\langle A, B \rangle = A \cdot B$  as their inner product; that is,  $\langle A, B \rangle = A \cdot B = \text{Tr}(B^* A) = \sum_{i,j} \bar{b}_{ij} a_{ij}$ , where  $A = (a_{ij})$  and  $B = (b_{ij})$ .  $\|\cdot\|$  means the module of a complex number or the 2-norm of a complex vector or the  $F$ -norm of a complex matrix.

Let the third root of unity be denoted by  $\omega^0 = 1$ ,  $\omega = \omega^1 = e^{i(2\pi/3)}$ ,  $\omega^2 = e^{i(4\pi/3)}$ . Introduce a complex variable  $y_i \in \{1, \omega, \omega^2\}$ ,  $i = 1, \dots, n$ , then it is not hard to know that

$$\frac{2}{3} - \frac{1}{3} y_i \cdot y_j - \frac{1}{3} y_j \cdot y_i = \frac{2}{3} (1 - \text{Re}(y_i \cdot y_j)). \quad (1.1)$$

Denote  $S_k = \{i : y_i = \omega^k, k = 0, 1, 2\}$  and  $\mathbf{y} = (y_1, \dots, y_n)^T$ . Then the Max 3-cut problem can be expressed as

$$\begin{aligned} \text{M3C : } & \max f(\mathbf{y}) = \frac{2}{3} \sum_{i < j} w_{ij} (1 - \text{Re}(y_i \cdot y_j)) \\ & \text{s.t. } \mathbf{y} \in \{1, \omega, \omega^2\}^n, \end{aligned} \quad (1.2)$$

here  $\mathbf{y} \in \{1, \omega, \omega^2\}^n$  means that  $y_i \in \{1, \omega, \omega^2\}$ ,  $i = 1, \dots, n$ ,  $W = (w_{ij})$  is the weight-valued matrix of a given graph.

By relaxing the complex variable  $y_i$  into an  $n$  dimensional complex vector  $\mathbf{y}_i$ , we get a complex semidefinite programming (CSDP) relaxation of (M3C) as follows:

$$\begin{aligned} \text{CSDP : } & \max \frac{2}{3} \sum_{i < j} w_{ij} (1 - \text{Re}(\mathbf{y}_i \cdot \mathbf{y}_j)) \\ & \text{s.t. } \|\mathbf{y}_i\| = 1, \quad i = 1, 2, \dots, n, \\ & \quad A_{ij}^k \cdot Y \geq -1, \quad i, j = 1, 2, \dots, n, \quad k = 0, 1, 2 \\ & \quad Y \succcurlyeq 0, \end{aligned} \quad (1.3)$$

where  $Y_{ij} = \mathbf{y}_i \cdot \mathbf{y}_j$ ,  $A_{ij}^k = \omega^k \mathbf{e}_i \mathbf{e}_j^T + \omega^{-k} \mathbf{e}_j \mathbf{e}_i^T$  and  $\mathbf{e}_i$  denotes the vector with zeros everywhere except for an unit in the  $i$ th component. It is easily to verify that constraints  $A_{ij}^k \cdot Y \geq -1$  can be expressed as

$$\operatorname{Re}(\omega^k Y_{ij}) \geq -\frac{1}{2}, \quad k = 0, 1, 2. \quad (1.4)$$

To get an approximate solution of M3C, Goemans and Williamson [6] do not directly solve the CSDP, but solve an equivalent real SDP with following form (Although some softwares, such as SeDuMi [8] and the earlier version of SDPT3-4.0 [9], can deal with SDPs with complex data, this does not reduce the dimensions of problems):

$$\begin{aligned} \text{RSDP : } & \max \frac{1}{2} \begin{bmatrix} Q & O \\ O & Q \end{bmatrix} \cdot X \\ \text{s.t. } & \begin{bmatrix} \mathbf{e}_i \mathbf{e}_i^T & O \\ O & \mathbf{e}_i \mathbf{e}_i^T \end{bmatrix} \cdot X = 2, \quad i = 1, 2, \dots, n, \\ & \begin{bmatrix} \operatorname{Re}(A_{ij}^k) & -\operatorname{Im}(A_{ij}^k) \\ \operatorname{Im}(A_{ij}^k) & \operatorname{Re}(A_{ij}^k) \end{bmatrix} \cdot X \geq -2, \quad 1 \leq i < j \leq n, \quad k = 0, 1, 2 \\ & \begin{bmatrix} A_{ij}^0 & O \\ O & -A_{ij}^0 \end{bmatrix} \cdot X = 0, \quad 1 \leq i < j \leq n, \\ & \begin{bmatrix} O & A_{ij}^0 \\ A_{ij}^0 & O \end{bmatrix} \cdot X = 0, \quad 1 \leq i < j \leq n, \\ & \begin{bmatrix} O & \mathbf{e}_i \mathbf{e}_i^T \\ \mathbf{e}_i \mathbf{e}_i^T & O \end{bmatrix} \cdot X = 0, \quad i = 1, 2, \dots, n, \\ & X \in \mathcal{S}_+^{2n}, \end{aligned} \quad (1.5)$$

where  $Q = (1/3) \operatorname{diag}(We) - W$  is the Laplace matrix of given graph,  $O$  is an  $n$ -dimensional full zeros matrix.

In RSDP, the first, third, and forth classes of equality constraints ensure that  $X_{ii} = 1$ ,  $i = 1, 2, \dots, n$  and with the form

$$X = \begin{bmatrix} R & -S \\ S & R \end{bmatrix}. \quad (1.6)$$

The final two classes of equality constraints ensure that  $S_{ii} = 0$  ( $i = 1, \dots, n$ ) and  $S$  is a skew-symmetric matrix.

If  $X$  is an optimal solution of RSDP, then the complex matrix  $\hat{Y} = R + iS$  is an optimal solution of CSDP. Then one can randomly generate a complex vector  $\xi$ , such that  $\xi \sim N(0, \hat{Y})$ , and set

$$\hat{y}_i = \begin{cases} 1, & \text{if } \operatorname{Arg}(\xi_i) \in \left[0, \frac{2\pi}{3}\right), \\ \omega, & \text{if } \operatorname{Arg}(\xi_i) \in \left[\frac{2\pi}{3}, \frac{4\pi}{3}\right), \\ \omega^2, & \text{if } \operatorname{Arg}(\xi_i) \in \left[\frac{4\pi}{3}, 2\pi\right), \end{cases} \quad (1.7)$$

where  $\operatorname{Arg}(\cdot) \in [0, 2\pi]$  means the complex angle principal value of a complex number. Goemans and Williamson [6] had verified that, see also Zhang and Huang [7],

$$f(\hat{y}) \geq 0.836 \cdot (Q \cdot \hat{Y}). \quad (1.8)$$

The algorithm proposed by Goemans and Williamson [6] can obtain a very good approximate ratio, and RSDP can be solved by interior point algorithm, but the 0.836-approximate algorithm will be not practical in numerical study for the Max 3-cut problem. From RSDP, one can see that for a graph with  $n$  nodes, RSDP has  $2n + 5n(n - 1)/2$  constraints and  $3n(n - 1)/2$  slack variables via the inequality constraints. That is to say, RSDP has a  $2n$  dimensional unknown symmetrical semidefinite positive matrix variable and a  $3n(n - 1)/2$  dimensional unknown vector variable, and  $2n + 5n(n - 1)/2$  constraints, and has also many matrices without an explicit block diagonal structure although they are sparse. For instance, when  $n = 100$ , RSDP becomes a very-high-dimensional semidefinite programming problem with 14850 slack variables and 24950 constraints. Further, as we known, it is only a class of universal and medium-scale instances for Max 3-cut problems with 50 to 100 nodes. Hence, it will be very time consuming to solve such a RSDP relaxation of M3C using the current existing any SDP softwares. This leads that 0.836-approximate algorithm is not suitable for computational study of the Max 3-cut problem. This limitation for solving M3C based on CSDP (or RSDP) relaxation motivates us to find a new efficient and fast algorithm for the practical purpose for the Max 3-cut problem.

In the current paper, we first establish a definition of  $K$ -neighborhood structure of the Max 3-cut problem and design a local search algorithm to find the local minimizer. And then, we propose a variable neighborhood search (VNS) metaheuristic with stochastic steps which is originally considered by Mladenović and Hansen [10], by which we can find efficiently a high-quality global approximate solution of the Max 3-cut problem. Further, combining a greed algorithm, we extend the proposed algorithm to the Max 3-section problem. To the best of our knowledge, it is first time to consider the computational study of the Max 3-cut problem. In order to test the performance of the proposed algorithm, we compare the numerical results with Goemans and Williamson's 0.836-approximate algorithm.

This paper is organized as follows. In Section 2, we give some definitions and lemmas. In Section 3, we present the VNS metaheuristic for solving the Max 3-cut problem. The VNS is extended to the Max 3-section problem in Section 4. In Section 5, we give some numerical results and comparisons.

## 2. Preliminaries

In this section, we will establish some definitions and give some facts for our sequel purpose. For the third roots of unity,  $1, \omega, \omega^2$ , we can get the following fact:

$$\|1 - \omega\|^2 = \|\omega - \omega^2\|^2 = \|1 - \omega^2\|^2 = 3. \quad (2.1)$$

Denote  $\mathbb{S} = \{1, \omega, \omega^2\}^n$ . Then based on (2.1), for any  $\mathbf{y} \in \mathbb{S}$ , we may definite a  $K$ -neighborhood of  $\mathbf{y}$  as follows.

*Definition 2.1.* For any  $\mathbf{y} \in \mathbb{S}$  and any positive integer number  $K$  ( $1 \leq K \leq n$ ), one defines the  $K$ -neighborhood, denoted by  $N_K(\mathbf{y})$ , of  $\mathbf{y}$  as the set

$$N_K(\mathbf{y}) = \left\{ \mathbf{z} \in \mathbb{S} : \|\mathbf{z} - \mathbf{y}\|^2 = \sum_{i=1}^n \|z_i - y_i\|^2 \leq 3K \right\}. \quad (2.2)$$

In particular, if  $K = 1$ , we write the 1-neighborhood  $N_1(\mathbf{y})$  of  $\mathbf{y}$  as  $N(\mathbf{y})$ .

The boundary of the  $K$ -neighborhood  $N_K(\mathbf{y})$  is defined by  $\partial N_K(\mathbf{y}) = \{\mathbf{z} \in \mathbb{S} : \|\mathbf{y} - \mathbf{z}\|^2 = 3K\}$ . Clearly,  $N(\mathbf{y}) = \partial N(\mathbf{y})$ . If  $\mathbf{z} \in \partial N_K(\mathbf{y})$ , we call  $\mathbf{z}$  a  $K$ -neighbor of  $\mathbf{y}$ . From Definition 2.1, the difference of between points  $\mathbf{y}$  and its  $K$ -neighbor  $\mathbf{z}$  is that they have only  $K$  different components. By computing straightforwardly, we get the number of elements of  $\partial N_K(\mathbf{y})$ , that is  $|\partial N_K(\mathbf{y})| = 2^K C_n^K$ . Particularly,  $|\partial N(\mathbf{y})| = 2n$  when  $K = 1$ .

*Example 2.2.* Let  $\mathbf{y} = (\omega, \omega, \omega^2)^T \in \{1, \omega, \omega^2\}^3$ . Then  $(1, \omega, \omega^2)^T \in N(\mathbf{y})$ ,  $(1, \omega^2, \omega^2)^T \in \partial N_2(\mathbf{y}) \subset N_2(\mathbf{y})$ , and  $(1, \omega^2, \omega)^T \in \partial N_3(\mathbf{y}) \subset N_3(\mathbf{y})$ .

*Definition 2.3.* For any  $u \in \{0, 1, 2\}$ , define two maps from  $\{1, \omega, \omega^2\}$  to itself as follows:  $\tau_i(\omega^u) = \omega^{u+i} \in \{1, \omega, \omega^2\}$ ,  $i = 1, 2$ .

Clearly, for any  $u \in \{0, 1, 2\}$ ,  $\tau_i(\omega^u) \neq \omega^u$ ,  $i = 1, 2$  and  $\tau_1(\omega^u) \neq \tau_2(\omega^u)$ . Applying Definition 2.3, for any  $\mathbf{z} \in N(\mathbf{y})$  there exists an unique component,  $z_k$  say, of  $\mathbf{z}$ , such that  $z_k \neq y_k$  and either  $z_k = \tau_1(y_k)$  or  $z_k = \tau_2(y_k)$ , and other components of  $\mathbf{z}$  and  $\mathbf{y}$  are the same. For simplicity, for any  $\mathbf{z} \in N(\mathbf{y})$  with  $z_k \neq y_k$  and  $z_i = y_i$  ( $i = 1, \dots, n, i \neq k$ ), we denote by  $\mathbf{z} = \tau_1^k(\mathbf{y})$  or  $\mathbf{z} = \tau_2^k(\mathbf{y})$  corresponding to  $z_k = \tau_1(y_k)$  or  $z_k = \tau_2(y_k)$ . By Definitions 2.1 and 2.3, for any  $\mathbf{y} \in \mathbb{S}$ , we can structure its 1-neighborhood points using maps defined by Definition 2.3; that is, we have the following result.

**Lemma 2.4.** Let  $\tau_i(\cdot)$  ( $i = 1, 2$ ) be defined by Definition 2.3. Then, for any  $\mathbf{y} \in \mathbb{S}$  and any fixed positive integer number  $k$  ( $1 \leq k \leq n$ ), one has

$$\tau_i^k(\mathbf{y}) \in N(\mathbf{y}), \quad i = 1, 2, \quad (2.3)$$

that is,  $\tau_1^k(\mathbf{y})$  and  $\tau_2^k(\mathbf{y})$  are two 1-neighborhood points of  $\mathbf{y}$ .

*Definition 2.5.* A point  $\hat{\mathbf{y}} \in \mathbb{S}$  is called a  $K$ -local maximizer of the function  $f$  over  $\mathbb{S}$ , if  $f(\hat{\mathbf{y}}) \geq f(\mathbf{y})$ , for all  $\mathbf{y} \in N_K(\hat{\mathbf{y}})$ . Furthermore, if  $f(\hat{\mathbf{y}}) \geq f(\mathbf{y})$  for all  $\mathbf{y} \in \mathbb{S}$ , then  $\hat{\mathbf{y}}$  is called a global

maximizer of  $f$  over  $\mathbb{S}$ . A 1-local maximizer of the function  $f$  is also called a local maximizer of the function  $f$  over  $\mathbb{S}$ .

### 3. VNS for Max 3-Cut

#### 3.1. Local Search Algorithm

Let  $\mathbf{y}^0 = (y_1^0, \dots, y_n^0)^T \in \mathbb{S}$  be a feasible solution of problem M3C. If  $\mathbf{y}^0$  is not a local maximizer of  $f$ , then for all  $\mathbf{y} \in N(\mathbf{y}^0)$ , we may find a  $\tilde{\mathbf{y}} \in N(\mathbf{y}^0)$ , such that  $f(\tilde{\mathbf{y}}) = \max\{f(\mathbf{y}) : \mathbf{y} \in N(\mathbf{y}^0)\}$ . It is clear that  $f(\tilde{\mathbf{y}}) \geq f(\mathbf{y}^0)$ . If  $\tilde{\mathbf{y}}$  is not still a local maximizer of  $f$ , then replacing  $\mathbf{y}^0$  with  $\tilde{\mathbf{y}}$  and repeating the process until a point  $\hat{\mathbf{y}}$  satisfying  $f(\hat{\mathbf{y}}) = \max\{f(\mathbf{y}) : \mathbf{y} \in N(\hat{\mathbf{y}})\}$  is found, which indicates that  $\hat{\mathbf{y}}$  is a local maximizer of  $f$ .

For any positive integer number  $k$  ( $1 \leq k \leq n$ ), let  $\mathbf{y}^k = (y_1^k, \dots, y_n^k)^T = \tau_i^k(\mathbf{y}^0) \in N(\mathbf{y}^0)$  ( $i = 1, 2$ ); that is,

$$\begin{aligned} y_i^k &= y_i^0, \quad i = 1, 2, \dots, k-1, k+1, \dots, n; \\ y_k^k &\neq y_k^0. \end{aligned} \tag{3.1}$$

Denote

$$\delta(k) = f(\mathbf{y}^0) - f(\mathbf{y}^k). \tag{3.2}$$

Then, we have the following result whose proof is clear.

**Lemma 3.1.** Consider

$$\delta(k) = \begin{cases} \frac{2}{3} \sum_{i=1}^{k-1} w_{ik} \operatorname{Re} [y_i^0 \cdot (y_k^k - y_k^0)] + \frac{2}{3} \sum_{j=k+1}^n w_{kj} \operatorname{Re} [(y_k^k - y_k^0) \cdot y_j^0], & k > 1; \\ \frac{2}{3} \sum_{j=k+1}^n w_{kj} \operatorname{Re} [(y_k^k - y_k^0) \cdot y_j^0], & k = 1. \end{cases} \tag{3.3}$$

Based on Lemma 3.1, if we know the value of  $f(\mathbf{y}^0)$ , then we can obtain the value function  $f(\mathbf{y}^k)$  at next iterative point  $\mathbf{y}^k$  by calculating  $\delta(k)$  by (3.3), instead of calculating directly the values  $f(\mathbf{y}^k)$ , which reduces sharply the computational cost. By Definition 2.1, there exist two points satisfying (3.1) for fixed  $k$ ; that is, when  $\mathbf{y}^k \in N(\mathbf{y}^0)$  and (3.1) is satisfied, then either  $y_k^k = \tau_1(y_k^0)$  or  $y_k^k = \tau_2(y_k^0)$ . For our convenience, we denote  $\delta(k)$  by  $\delta_1(k)$  when  $y_k^k = \tau_1(y_k^0)$  and by  $\delta_2(k)$  when  $y_k^k = \tau_2(y_k^0)$ . In what follows, we describe the local search algorithm for the Max 3-cut problem denoted by LSM3C; by this algorithm, we can get a local maximizer of function  $f(\mathbf{y})$  over  $\mathbb{S}$ .

For **LSM3C**, one has the following.

- (1) Input any initial feasible solution  $\mathbf{y}^0$  of problem (M3C).
- (2) For  $k$  from 1 to  $n$ , set  $\mathbf{z}^{k1} = \tau_1^k(\mathbf{y}^0)$ , calculate  $\delta_1(k)$ , and set again  $\mathbf{z}^{k2} = \tau_2^k(\mathbf{y}^0)$ , calculate  $\delta_2(k)$ .

(3) Find  $\delta_{i^*}(k^*)$  by the following way:

$$\delta_{i^*}(k^*) = \min\{\delta_1(1), \delta_2(1), \dots, \delta_1(k), \delta_2(k), \dots, \delta_1(n), \delta_2(n)\}. \quad (3.4)$$

(4) If  $\delta_{i^*}(k^*) \geq 0$ , then set  $\hat{\mathbf{y}} = \mathbf{y}^0$ , return  $\hat{\mathbf{y}}$ , and stop. Otherwise, go to next.

(5) Set  $\mathbf{y}^0 = \tau_{i^*}^{k^*}(\mathbf{y}^0)$ ; go to Step 2.

### 3.2. Variable Neighborhood Stochastic Search

Let  $\hat{\mathbf{y}}$  be a local maximizer obtained by LSM3C and  $K_{\max}$  ( $1 < K_{\max} \leq n$ ) a fixed positive integer number. we now describe the variable neighborhood search (VNS) with stochastic steps, by which we can find an approximate global maximizer of problem (M3C). The proposed VNS algorithm actually has three phases: First, for any given positive integer number  $K < K_{\max}$ , a  $K$ -neighborhood point,  $\mathbf{y}$  say, is randomly selected; that is,  $\mathbf{y} \in N_K(\hat{\mathbf{y}})$ . Next, a solution,  $\hat{\mathbf{y}}$  say, is obtained by applying algorithm LSM3C to  $\mathbf{y}$ . Finally, the current solution jumps from  $\hat{\mathbf{y}}$  to  $\hat{\hat{\mathbf{y}}}$  if it improves the former one. Otherwise, the order  $K$  of the neighborhood is increased by one when  $K < K_{\max}$  and the above steps are repeated until some stopping condition is met. The VNS that is also called  $k$ -max [11] can be illustrated as follows.

For VNS-k, one has the following.

- (1) Arbitrary choose a point  $\mathbf{y}^0 \in \mathbb{S}$ , implement LSM3C starting from  $\mathbf{y}^0 \in \mathbb{S}$  and denote the obtained local maximizer by  $\hat{\mathbf{y}}$ . Set  $K = 1$ .
- (2) Randomly take a point  $\mathbf{y} \in \partial N_{I(K)}(\hat{\mathbf{y}})$  and implement again LSM3C from  $\mathbf{y}$ , and denote the obtained new local maximizer by  $\hat{\hat{\mathbf{y}}}$ .
- (3) If  $f(\hat{\hat{\mathbf{y}}}) > f(\hat{\mathbf{y}})$ , then set  $\hat{\mathbf{y}} = \hat{\hat{\mathbf{y}}}$  and  $K = 1$ ; go to Step 2.
- (4) If  $K < K_{\max} (\leq n)$ , set  $K = K + 1$ ; go to Step 2. Otherwise, return  $\hat{\mathbf{y}}$  as an approximate global solution of problem M3C and stop.

The subscript  $I(K)$  in Step 2 is a function of  $K$  and is also a positive integer number not greater than  $n$ .  $I(K)$  reflects the main skill of converting the current neighborhood of local maximizer  $\hat{\mathbf{y}}$  into another neighborhood of  $\hat{\mathbf{y}}$ . For a given  $K_{\max}$ , let  $m = \lfloor n/K_{\max} \rfloor$  and  $K_0 = n - mK_{\max}$ , where  $\lfloor a \rfloor$  means the integral part of  $a$ . We divide the  $n$  neighborhoods of  $\hat{\mathbf{y}}$ ,  $N(\hat{\mathbf{y}}), N_2(\hat{\mathbf{y}}), \dots, N_K(\hat{\mathbf{y}}), \dots, N_n(\hat{\mathbf{y}})$  into  $K_{\max}$  neighborhood blocks  $N_{I(1)}(\hat{\mathbf{y}}), \dots, N_{I(K_{\max})}(\hat{\mathbf{y}})$ , such that, for  $K = 1, 2, \dots, K_{\max} - K_0$ ,

$$N_{(K-1)m+1}(\hat{\mathbf{y}}) \subseteq N_{I(K)}(\hat{\mathbf{y}}) \subseteq N_{Km+1}(\hat{\mathbf{y}}), \quad (3.5)$$

and, for  $K = K_{\max} - K_0 + 1, \dots, K_{\max} - K_0 + j, \dots, K_{\max}$ ,

$$N_{(K-1)(m+1)+1}(\hat{\mathbf{y}}) \subseteq N_{I(K)}(\hat{\mathbf{y}}) \subseteq N_{K(m+1)}(\hat{\mathbf{y}}). \quad (3.6)$$

In order to obtain the  $K_{\max}$  neighborhood blocks of  $\hat{\mathbf{y}}$ ,  $N_{I(K)}(\hat{\mathbf{y}})$ ,  $K = 1, \dots, K_{\max}$ , we divide the set  $\{1, 2, \dots, n\}$  into  $K_{\max}$  disjoint subsets, where each subset of the first  $K_{\max} - K_0$  subsets

has  $m$  integers and each subset of the last  $K_0$  subsets has  $m + 1$  integers. For any integer  $K (\leq K_{\max})$ , let

$$I(K) = (K - 1) \cdot m + [c \cdot m] + 1, \quad K = 1, 2, \dots, K_{\max} - K_0, \quad (3.7)$$

or

$$\begin{aligned} I(K) &= (K_{\max} - K_0)m + [(m + 1) \cdot c] + 1 + (m + 1)(K - (K_{\max} - K_0) - 1) \\ &= [(m + 1) \cdot c] + 1 + (m + 1)(K - 1) - (K_{\max} - K_0), \\ K &= K_{\max} - K_0 + 1, \dots, K_{\max} - K_0 + j, \dots, K_{\max}. \end{aligned} \quad (3.8)$$

Then we can randomly choose a point  $\mathbf{y}$  in  $\partial N_{I(K)}(\hat{\mathbf{y}})$ , where  $c \in (0, 1)$  is a random number from uniformly distribution  $\mathcal{U}(0, 1)$ , such that  $N_{I(K)}(\hat{\mathbf{y}})$  satisfies (3.5) or (3.6).

VNS- $k$  stops when the maximum  $K$  neighborhood is reached. Additionally, we also consider another termination criterion of VNS based on the maximum CPU-time and denoted by VNS- $t$ . VNS- $t$  can obtain a better solution than VNS- $k$  since VNS- $t$  actually runs several times VNS- $k$  in the maximum allowing time  $t_{\max}$ , but it generally has to spend more computational time. The VNS- $t$  can be stated as follows.

For VNS- $t$ , one has the following.

- (1) Set  $t_{\text{CPU}} = 0$ , running VNS- $k$  for an arbitrary initial point  $\mathbf{y}^0 \in \mathbb{S}$ , and let a local optimal solution  $\hat{\mathbf{y}}$  be obtained.
- (2) If  $K = K_{\max} (\leq n)$ , go to Step 3.
- (3) If  $t_{\text{CPU}} < t_{\max}$ , then set  $K = 1$ ; go to Step 2 in VNS- $k$ . Otherwise, return  $\hat{\mathbf{y}}$  as an approximate global solution of problem M3C and stop.

We mention that it differs from the classical variable neighborhood search meta-heuristic that is originally proposed by Mladenović and Hansen [10]. In order to obtain a global optimal solution or a high-quality approximate solution of problem M3C, we use two stochastic steps in VNS. First, for a fixed  $K$ , a  $K$ -neighbor of  $\hat{\mathbf{y}}$  is chosen randomly. Second, by the definition of  $I(K)$ , when we change the neighborhood of  $\hat{\mathbf{y}}$  from  $N_{I(K-1)}$  to  $N_{I(K)}$ ,  $N_{I(K)}$  may take any a neighborhood among  $N_{(K-1)m+j}$ ,  $j = 1, 2, \dots, m$  of  $\hat{\mathbf{y}}$ , which is decided by random number  $c$ . In VNS, positive integer  $K_{\max}$  decides the maximum search neighborhood block of  $\hat{\mathbf{y}}$ , which also decides directly the CPU-time of VNS. Based on the second stochastic step, we may choose a relative small  $K_{\max}$  comparing with  $n$ . This can decrease our computational time.

#### 4. A Greedy Algorithm for Max 3-Section

When the number of nodes  $n$  is a multiple of three and the condition  $|S_0| = |S_1| = |S_2| = n/3$  is required, the Max 3-cut problem becomes the Max 3-section problem. Notice that  $1 + \omega + \omega^2 =$

0, then the Max 3-section problem can be formulated as the following programming problem M3S:

$$\begin{aligned} \text{M3S : max } & \frac{2}{3} \sum_{i < j} w_{ij} (1 - \operatorname{Re}(y_i \cdot y_j)) \\ \text{s.t. } & \sum_{i=1}^n y_i = 0, \\ & \mathbf{y} \in \mathbb{S}, \end{aligned} \tag{4.1}$$

and its CSDP relaxation is

$$\begin{aligned} \text{CSDP1 : max } & \frac{2}{3} \sum_{i < j} w_{ij} (1 - \operatorname{Re}(\mathbf{y}_i \cdot \mathbf{y}_j)) \\ \text{s.t. } & \mathbf{e}\mathbf{e}^T \cdot \mathbf{Y} = 0, \\ & \|\mathbf{y}_i\| = 1, i = 1, 2, \dots, n, \\ & A_{ij}^k \cdot \mathbf{Y} \geq -1, \quad i, j = 1, 2, \dots, n, k = 0, 1, 2 \\ & \mathbf{Y} \succcurlyeq 0, \end{aligned} \tag{4.2}$$

where  $\mathbf{e}$  is the column vector of all ones. Andersson [12] extended Frieze and Jerrum's random rounding method to M3S and obtained a  $(2/3) + O(1/n^3)$ -approximate algorithm, which is the current best approximate ratio for M3S; also see the recent research of Gaur et al. [13]. The author of the current paper considers a special the Max 3-Section problem and obtains a 0.6733-approximate algorithm; see Ling (2009) [14].

Clearly, the feasible region of problem M3S is a subset of  $\mathbb{S}$ , and the optimal value of problem M3S is not greater than that of problem M3C. Assume that we have get a global optimal solution or a high-quality approximate solution  $\hat{\mathbf{y}}$  of problem M3C. It is clear that  $\hat{\mathbf{y}}$  may not satisfy the condition  $\sum_{i=1}^n \hat{y}_i = 0$ . But we may adjust  $\hat{\mathbf{y}}$  to get a new feasible solution  $\mathbf{y}^s$  using a greedy algorithm, such that  $\mathbf{y}^s$  satisfies  $\sum_{i=1}^n y_i^s = 0$ . This is the motivation that we propose the greedy algorithm for the Max 3-section problem.

For the sake of our analysis, without loss of generality, we assume that the local maximizer  $\hat{\mathbf{y}}$  satisfies  $|S_0| = \max\{|S_0|, |S_1|, |S_2|\}$ . This means that  $S_0 = \{i : \hat{y}_i = 1\}$  is the subset of  $V$  with maximum cardinal number. If  $|S_k| = \max\{|S_0|, |S_1|, |S_2|\}$  ( $k \neq 0, k = 1, 2$ ), then we may set  $y_i^N = \bar{w}^k \hat{y}_i$ ,  $i = 1, \dots, n$ . The resulted new solution  $\mathbf{y}^N = (y_1^N, \dots, y_n^N)$  will not change the objective value since  $f(\hat{\mathbf{y}}) = f(\bar{w}^k \hat{\mathbf{y}})$  ( $k \neq 0, k = 1, 2$ ); moreover, the new partition  $\{S_0^N, S_1^N, S_2^N\}$  based on  $\mathbf{y}^N$  satisfies  $|S_0^N| = \max\{|S_0^N|, |S_1^N|, |S_2^N|\}$ . By our assumption, the partition  $S = \{S_0, S_1, S_2\}$  still exist four possible cases.

*Case 1.*  $|S_0| \geq |S_1| \geq n/3 \geq |S_2|$ .

*Case 2.*  $|S_0| \geq n/3 \geq |S_1| \geq |S_2|$ .

*Case 3.*  $|S_0| \geq |S_2| \geq n/3 \geq |S_1|$ .

*Case 4.*  $|S_0| \geq n/3 \geq |S_2| \geq |S_1|$ .

The sizes adjusting greedy algorithm of Cases 3 and 4 are similar to Cases 1 and 2. Hence, we mainly consider Cases 1 and 2 for adjusting the partition of  $V$  from  $S = \{S_0, S_1, S_2\}$  to  $\tilde{S} = \{\tilde{S}_0, \tilde{S}_1, \tilde{S}_2\}$  such that  $|\tilde{S}_k| = n/3$ ,  $k = 0, 1, 2$ . Denote

$$\begin{aligned}\delta_0(i) &= \sum_{j \in S_1 \cup S_2} w_{ij}, \quad i \in S_0, \\ \delta_{01}(i) &= \sum_{j \in S_1} w_{ij}, \quad i \in S_0, \quad \delta_{10}(i) = \sum_{j \in S_0} w_{ij}, \quad i \in S_1, \\ \delta_{02}(i) &= \sum_{j \in S_2} w_{ij}, \quad i \in S_0, \quad \delta_{20}(i) = \sum_{j \in S_0} w_{ij}, \quad i \in S_2, \\ \delta_{12}(i) &= \sum_{j \in S_2} w_{ij}, \quad i \in S_1, \quad \delta_{21}(i) = \sum_{j \in S_1} w_{ij}, \quad i \in S_2.\end{aligned}\tag{4.3}$$

Then, it follows from simple computation that

$$\begin{aligned}\delta_0(i) &= \delta_{01}(i) + \delta_{02}(i), \quad \text{for each } i \in S_0, \\ \sum_{i \in S_k} \delta_{kl}(i) &= \sum_{i \in S_l} \delta_{lk}(i), \quad k, l = 0, 1, 2, k \neq l, \\ f(\hat{y}) &= \sum_{i \in S_0} \delta_0(i) + \sum_{i \in S_1} \delta_{12}(i) \\ &= \sum_{i \in S_0} \delta_{01}(i) + \sum_{i \in S_0} \delta_{02}(i) + \sum_{i \in S_1} \delta_{12}(i) \\ &= d_{01} + d_{02} + d_{12},\end{aligned}\tag{4.4}$$

where  $d_{01} = \sum_{i \in S_0} \delta_{01}(i)$ ,  $d_{02} = \sum_{i \in S_0} \delta_{02}(i)$ ,  $d_{12} = \sum_{i \in S_1} \delta_{12}(i)$ .

In what follows, we describe the size adjusting greedy algorithms (SAGAs) for Cases 1 and 2, and denote the greedy algorithms for the two cases by SAGA1 and SAGA2, respectively.

For **SAGA1**, one has the following.

(1) Calculate

$$m_{02} = \frac{\sum_{i \in S_0} \delta_{02}(i)}{|S_0|}, \quad m_{12} = \frac{\sum_{i \in S_1} \delta_{12}(i)}{|S_1|}.\tag{4.5}$$

(2) If  $m_{02} \geq m_{12}$ , let  $S_1 = \{j_1, j_2, \dots, j_{|S_1|}\}$ , where  $\delta_{12}(j_l) \geq \delta_{12}(j_{l+1})$ ,  $l = 1, 2, \dots, |S_1|$ . Set  $\tilde{S}_1 = \{j_1, j_2, \dots, j_{n/3}\}$ ,  $\tilde{S}_2 = S_2 \cup (S_1 \setminus \tilde{S}_1)$  and renew to calculate

$$\delta'_{02}(i) = \sum_{j \in \tilde{S}_2} w_{ij},\tag{4.6}$$

for each  $i \in S_0$ . Let  $S_0 = \{i_1, i_2, \dots, i_{|S_0|}\}$ , where  $\delta'_{02}(i_k) \geq \delta'_{02}(i_{k+1})$ . Set  $\tilde{S}_0 = \{i_1, i_2, \dots, i_{n/3}\}$  and  $\tilde{S}_2 = \tilde{S}_2 \cup (S_0 \setminus \tilde{S}_0)$ .

- (3) If  $m_{02} < m_{12}$ , let  $S_0 = \{i_1, i_2, \dots, i_{|S_0|}\}$ , where  $\delta_{02}(i_k) \geq \delta_{02}(i_{k+1})$ ,  $k = 1, 2, \dots, |S_0|$ , set  $\tilde{S}_0 = \{i_1, i_2, \dots, i_{n/3}\}$ ,  $\hat{S}_2 = S_2 \cup (S_0 \setminus \tilde{S}_0)$ , and then renew to calculate

$$\delta'_{12}(i) = \sum_{j \in \hat{S}_2} w_{ij}, \quad (4.7)$$

for each  $i \in S_1$ . Set  $\tilde{S}_1 = \{j_1, j_2, \dots, j_{n/3}\}$  and  $\tilde{S}_2 = \hat{S}_2 \cup (S_1 \setminus \tilde{S}_1)$ , where  $\delta'_{12}(j_k) \geq \delta'_{12}(j_{k+1})$  here.

- (4) Return the current partition  $\tilde{S} = \{\tilde{S}_0, \tilde{S}_1, \tilde{S}_2\}$ ; stop.

For **SAGA2**, one has the following.

- (1) Calculate  $d_{01} = \sum_{i \in S_0} \delta_{01}(i)$ ,  $d_{02} = \sum_{i \in S_0} \delta_{02}(i)$ , and

$$m_{01} = \frac{d_{01}}{|S_0|}, \quad m_{02} = \frac{d_{02}}{|S_0|}. \quad (4.8)$$

- (2) If  $m_{01} \leq m_{02}$ , let

$$S_0 = \{i_1, i_2, \dots, i_{|S_0|}\}, \quad (4.9)$$

where  $\delta_{01}(i_k) \geq \delta_{01}(i_{k+1})$ ,  $k = 1, 2, \dots, |S_0|$ . Set

$$\hat{S}_0 = \{i_1, i_2, \dots, i_{|S_0|-q_1}\}, \quad \tilde{S}_1 = S_1 \cup (S_0 \setminus \hat{S}_0), \quad (4.10)$$

where  $q_1 = (n/3) - |S_1|$ . Renew to calculate

$$\delta'_{02}(i) = \sum_{j \in S_2} w_{ij}, \quad i \in \hat{S}_0. \quad (4.11)$$

and let

$$\hat{S}_0 = \left\{ i'_1, i'_2, \dots, i'_{|\hat{S}_0|} \right\}, \quad (4.12)$$

where  $\delta'_{02}(i'_k) \geq \delta'_{02}(i'_{k+1})$ ,  $k = 1, 2, \dots, |\hat{S}_0|$ . Set

$$\tilde{S}_0 = \{i'_1, i'_2, \dots, i'_{n/3}\}, \quad \tilde{S}_2 = S_2 \cup (\hat{S}_0 \setminus \tilde{S}_0). \quad (4.13)$$

- (3) If  $m_{01} > m_{02}$ , let

$$S_0 = \{i_1, i_2, \dots, i_{|S_0|}\}, \quad (4.14)$$

where  $\delta_{02}(i_k) \geq \delta_{02}(i_{k+1})$ ,  $k = 1, 2, \dots, |S_0|$ . Set

$$\hat{S}_0 = \{i_1, i_2, \dots, i_{|S_0|-q_2}\}, \quad \tilde{S}_2 = S_2 \cup (S_0 \setminus \hat{S}_0), \quad (4.15)$$

where  $q_2 = (n/3) - |S_2|$ . Renew to calculate

$$\delta'_{01}(i) = \sum_{j \in S_1} w_{ij}, \quad i \in \hat{S}_0. \quad (4.16)$$

and let

$$\hat{S}_0 = \left\{ i'_1, i'_2, \dots, i'_{|\hat{S}_0|} \right\}, \quad (4.17)$$

where  $\delta'_{01}(i'_k) \geq \delta'_{01}(i'_{k+1})$ ,  $k = 1, 2, \dots, |\hat{S}_0|$ . Set

$$\tilde{S}_0 = \left\{ i'_1, i'_2, \dots, i'_{n/3} \right\}, \quad \tilde{S}_1 = S_1 \cup (\hat{S}_0 \setminus \tilde{S}_0). \quad (4.18)$$

(4) Return the current partition  $\tilde{S} = \{\tilde{S}_0, \tilde{S}_1, \tilde{S}_2\}$ ; stop.

## 5. Numerical Results

This section describes the obtained experimental results for some instances of Max 3-cut and Max 3-Section problems using the proposed VNS metaheuristic. We also show a quantitative comparison with 0.836-approximate algorithm. The computational experiments are performed in an Intel Pentium 4 processor at 2.0 GHz, with 512 MB of RAM, and all algorithms are coded in Matlab. Because RSDP relaxation of M3C includes many slack variables, many constraints, and matrices variables without a block diagonal structures, in our numerical comparisons, we choose SDPT3-4.0 [9], one of the best and well-known solvers of semidefinite programming, to solve RSDP relaxation of M3C.

All our test problems are generated randomly by the following way. Let  $p \in (0, 1)$  be a constant and  $r \in (0, 1)$  a random number. If  $r \leq p$ , then there is an edge between nodes  $i$  and  $j$  with weight  $w_{ij}$ , that is, a random integer between 1 and 10. Otherwise,  $w_{ij} = 0$ ; that is, there is no edge between nodes  $i$  and  $j$ . Because of the limits of memory of SDPT3, when  $n > 200$ , RSDP becomes a huge semidefinite programming problem with not less than 59700 slack variables and 99900 constraints and is out of memory of SDPT3. Hence, in the numerical experiments, we consider 30 instances with  $p = 0.1, 0.3, 0.6$ , and  $n$  varying from 20 to 200.

Firstly, we check the influence of  $K_{\max}$  on the quality of solution obtained by VNS- $k$ . For a given graph, we take  $K_{\max} = 3, 5, 10, 15, 30$ ; Table 1 presents the results, where  $Wnp$  in the first column of this table and the following tables means that a graph is randomly generated with nodes  $n$  and density  $p$ ; for instance,  $W30.6$  presents a graph generated randomly with  $n = 30$  and  $p = 0.6$ . We find from Table 1 that the influence of  $K_{\max}$  to objective value denoted by  $Obj$  in Table 1 is slight when  $K_{\max} > 5$ , but the CPU time increases sharply as  $K_{\max}$  increases. This result is actually not surprising. Indeed, because  $I(K) > K$ , we choose

**Table 1:** The objective value obtained by VNS for M3C with different  $K_{\max}$ .

Wnp	$K_{\max} = 3$		$K_{\max} = 5$		$K_{\max} = 10$		$K_{\max} = 15$		$K_{\max} = 30$	
	Obj	t	Obj	t	Obj	t	Obj	t	Obj	t
W60.1	960	1.38	965	2.14	967	4.63	967	6.29	969	12.50
W100.3	6001	7.55	6010	12.66	6013	23.37	6013	29.28	6015	69.52
W120.1	3323	13.76	3335	34.07	3337	38.78	3339	50.48	3343	101.85

randomly a point  $\mathbf{y}$  in  $\partial N_{I(K)}(\hat{\mathbf{y}})$ , instead of  $\partial N_K(\hat{\mathbf{y}})$ . This avoids to choose too large  $K_{\max}$  which leads to more CPU-time cost. Hence, in sequel numerical comparisons, we fix  $K_{\max} = 5$  for all test problems.

Secondly, we compare VNS (VNS- $k$ , VNS- $t$ ) metaheuristic with 0.836-approximate algorithm for all test problems. To avoid the effect of initial points, for each test problem, after RSDP is solved, we run the round procedure of 0.836-approximate algorithm and VNS metaheuristic ten times, respectively.

Table 2 gives the result of numerical comparisons. In the numerical presentations of Table 2,  $\text{Obj}_{\text{rsdp}}$  is the optimal value of problem RSDP; that is, it is an upper bound of M3C.  $\text{Obj}_{\text{GM}}$  is the largest value obtained by 0.836-approximate algorithm in the ten tests.  $\text{Obj}_{\text{vns}}$  stands for the largest value obtained by VNS for M3C in the ten tests, respectively.  $m$  and  $s.v.$  are the number of constraints and slack variables (s.v.), respectively.  $t_{\text{GM}}$  and  $t_{\text{vns}-k}$  are the average time (second) associated with the two algorithms in the ten tests. For the maximum CPU time of VNS- $t$ , we take  $t_{\max} = 2t_{\text{vns}-k}$ , but the real CPU time of VNS- $t$  will be greater than  $t_{\max}$ . Additionally, for measuring the performance of solutions, we take

$$\rho = \frac{\text{Obj}_{\text{vns}} - \text{Obj}_{\text{rsdp}}}{\text{Obj}_{\text{rsdp}}} = \frac{\text{Obj}_{\text{vns}}}{\text{Obj}_{\text{rsdp}}} - 1 \quad (5.1)$$

for M3C and

$$\rho = \frac{\text{Obj}_{\text{vns+saga}} - \text{Obj}_{\text{rsdp}}}{\text{Obj}_{\text{rsdp}}} = \frac{\text{Obj}_{\text{vns+saga}}}{\text{Obj}_{\text{rsdp}}} - 1 \quad (5.2)$$

for M3S. Clearly,  $\rho$  can reflect how close to the solution obtained by VNS from the optimal solution of RSDP. One can see from Table 2 that (1) the VNS metaheuristic not only can obtain a better solution than 0.836-approximate algorithm for all test problems, but also that the elapsed CPU-time of VNS metaheuristic is much less than that of 0.836-approximate algorithm for all test problems, (2) the performance of solution can be improved by VNS- $t$  for most of test problems when the termination criterion of VNS is based on the maximum CPU-time, but VNS- $t$  spends more computational time than VNS- $k$ . The improved performance can be reflected by  $\nabla\rho = \rho_t - \rho_k$  in the final column of Table 2. Average speaking, VNS- $t$  improves 0.91 percentage point.

Finally, we consider the solution of M3S by combining VNS- $k$  and greedy sizes-adjusted algorithm SAGA stated in Section 4. Let  $\hat{\mathbf{y}}$  be an approximate solution of M3C obtained by VNS; we can obtain an approximate solution of M3S from SAGA. The numerical results are reported by Table 3 in which  $\text{Obj}_{\text{vns+saga}}$  stands for the largest value obtained by VNS- $k$  plus SAGA for M3S. Although our sizes-adjusted algorithm may decrease the objective value obtained by VNS, the changes of objective values are very slight from

**Table 2:** The numerical comparisons of 0.836-approximate algorithm with VNS metaheuristic.

Wnp	$m$ s.v.	Obj <sub>rsdp</sub>	0.836-algorithm			VNS				
			Obj <sub>GM</sub>	$t_{GM}$	Obj <sub>vns-k</sub>	$t_{vns-k}$	$\rho_k\%$	Obj <sub>vns-t</sub>	$\rho_t\%$	$\nabla\rho\%$
W20.1	990	121	117	22.93	119	0.28	-1.65	119	-1.65	0
W20.3	570	232	225	25.58	228	0.29	-1.72	228	-1.72	0
W20.6		492	464	14.27	479	0.40	-2.65	479	-2.24	0.41
W30.1	2235	147	139	161.56	144	0.81	-2.04	144	-2.04	0
W30.3	1305	566	521	140.12	550	0.97	-2.83	550	-2.30	0.53
W30.6		1181	1101	94.93	1151	0.92	-2.54	1151	-2.03	0.51
W45.1	5040	675	594	202.48	605	0.92	-10.38	612	-9.33	1.05
W45.3	2970	1300	1147	211.72	1192	1.02	-8.31	1300	-7.15	1.16
W45.6		2441	2272	217.57	2313	1.02	-5.25	2350	-3.73	1.52
W60.1	8970	1069	918	256.35	965	2.14	-9.73	981	-8.23	1.50
W60.3	5310	2494	2228	256.57	2301	3.13	-7.74	2332	-6.50	1.24
W60.6		4403	4054	266.68	4213	2.48	-4.32	4307	-2.18	2.14
W80.1	15960	1567	1357	537.58	1415	5.82	-9.71	1438	-8.23	1.48
W80.3	9480	4094	3666	546.23	3801	7.35	-7.16	3912	-4.45	2.71
W80.6		7746	7328	495.34	7423	6.31	-4.17	7481	-3.42	0.75
W100.1	24950	2526	2149	893.52	2262	12.73	-10.46	2302	-8.87	1.60
W100.3	14850	6418	5814	882.73	6010	12.66	-6.36	6113	-4.75	1.61
W100.6		11956	11212	865.28	11391	11.99	-4.73	11422	-4.47	0.26
W120.1	33990	3746	3222	1476.54	3335	34.07	-10.98	3392	-9.45	1.53
W120.3	21420	9173	8266	1498.18	8575	35.41	-6.52	8623	-6.00	0.52
W120.6		16748	15596	1567.43	16056	43.80	-4.14	16114	-3.79	0.35
W150.1	56175	5821	5020	2618.11	5208	69.20	-10.54	5257	-9.69	0.85
W150.3	33525	14432	13209	3021.87	13543	70.60	-6.15	13607	-5.72	0.43
W150.6		26310	25076	3172.22	25405	74.46	-3.44	25517	-3.01	0.43
W180.05	80910	4303	3578	9043.25	3726	133.10	-13.41	3812	-11.41	1.20
W180.1	48330	7236	6328	10225.54	6481	130.07	-10.44	6547	-9.52	0.92
W180.3		20147	18436	9887.36	19031	132.14	-5.54	19213	-4.64	0.90
W180.6		37292	35386	10004.11	35949	102.77	-3.61	36124	-3.13	0.48
W200.05	99900	5174	4306	25872.33	4484	71.90	-13.40	4509	-12.85	0.55
W200.1	59700	9271	8092	29003.28	8799	149.50	-5.10	8853	-4.51	0.59
W200.5		38831	36481	28774.17	37477	200.16	-3.49	37552	-3.29	0.20

Table 3. Particular, objective values of some problems do not decrease, instead increase, such as W150.3. We do not compare the obtained results with Andersson's 2/3-approximate algorithm. Because we find that all approximate solutions of M3S obtained by VNS plus SAGA still are better than that of 0.836-approximate algorithm with the exception of only W30.1 and W30.3.

## 6. Conclusions

A variable neighborhood stochastic metaheuristic was proposed to solve the Max 3-cut and Max 3-section problems in this paper. Our algorithms can solve Max 3-cut and Max 3-section problems with different sizes and densities. Although 0.836-approximate algorithm

**Table 3:** The numerical results of combining VNS- $k$  metaheuristic with SAGA for M3S.

Wnp	Obj <sub>rsdp</sub>	Obj <sub>GM</sub>	Obj <sub>vns+saga</sub>	$\rho_k$ (%)
W30.1	147	139	138	-6.12
W30.3	566	521	518	-8.49
W30.6	1181	1101	1151	-2.55
W45.1	675	594	605	-10.38
W45.3	1300	1147	1191	-8.39
W45.6	2441	2272	2313	-5.25
W60.1	1069	918	952	-10.95
W60.3	2494	2228	2301	-7.74
W60.6	4403	4054	4213	-4.32
W120.1	3746	3222	3290	-12.18
W120.3	9173	8266	8449	-7.90
W120.6	16748	15596	16056	-4.14
W150.1	5821	5020	5203	-10.62
W150.3	14432	13209	13551	-6.11
W150.6	26310	25076	25096	-4.62
W180.05	4303	3578	3726	-13.41
W180.1	7236	6328	6396	-11.61
W180.3	20147	18436	18823	-6.58
W180.6	37292	35386	35947	-3.61

has the very good theoretic results, in numerical aspects, our comparisons indicate that the proposed VNS metaheuristic is superior to the well-known 0.836-approximate algorithm and can efficiently obtain very high-quality solutions of the Max 3-cut and Max 3-section problems.

We mention that the proposed algorithm in fact can deal with higher dimensional G-set graphs problems created by Pro. Rinaldi using a graph generator, rudy. But, we cannot give numerical comparisons with 0.836-approximate algorithm since RSDP relaxations of these problems are out of memory of the current all SDP software. In additionally, if we increase  $K_{\max}$  or  $t_{\max}$  in numerical implementing, then the quality of solution of M3C will further be improved by VNS.

## Funding

This work is supported by the National Natural Science Foundations of China (no. 71001045, 10971162), the China Postdoctoral Science Foundation (no. 20100480491), the Natural Science Foundation of Jiangxi Province of China (no. 20114BAB211008), and the Jiangxi University of Finance and Economics Support Program Funds for Outstanding Youths.

## Acknowledgment

The authors would like to thank the editor and an anonymous referee for their numerous suggestions for improving the paper.

## References

- [1] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, R. Miller and J. Thatcher, Eds., pp. 85–103, Plenum Press, New York, NY, USA, 1972.
- [2] M. R. Garey, D. S. Johnson, and L. Stockmeyer, "Some simplified NP-complete graph problems," *Theoretical Computer Science*, vol. 1, no. 3, pp. 237–267, 1976.
- [3] F. Barahona, M. Grötschel, G. Reinelt, and M. Juenger, "An application of combinatorial optimization to statistical physics and circuit layout design," *Operations Research*, vol. 36, no. 3, pp. 493–513, 1988.
- [4] M. X. Goemans and D. P. Williamson, "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming," *Journal of the Association for Computing Machinery*, vol. 42, no. 6, pp. 1115–1145, 1995.
- [5] A. Frieze and M. Jerrum, "Improved approximation algorithms for MAX  $k$ -CUT and MAX BISECTION," in *Integer Programming and Combinatorial Optimization*, E. Balas and J. Clausen, Eds., vol. 920, pp. 1–13, 1995.
- [6] M. X. Goemans and D. P. Williamson, "Approximation algorithms for MAX-3-CUT and other problems via complex semidefinite programming," *Journal of Computer and System Sciences*, vol. 68, no. 2, pp. 442–470, 2004.
- [7] S. Zhang and Y. Huang, "Complex quadratic optimization and semidefinite programming," *SIAM Journal on Optimization*, vol. 16, no. 3, pp. 871–890, 2006.
- [8] J. F. Sturm, "Using SeDuMi 1.02, 'a MATLAB toolbox for optimization over symmetric cones,'" *Optimization Methods and Software*, vol. 11, no. 1–4, pp. 625–653, 1999.
- [9] K.-C. Toh, M. J. Todd, and R. H. Tütüncü, "SDPT3 version 4.0 (beta)—a MATLAB software for semidefinite-quadratic-linear programming," 2004, <http://www.math.nus.edu.sg/~mattohkc/sdpt3.html>.
- [10] N. Mladenović and P. Hansen, "Variable neighborhood search," *Computers & Operations Research*, vol. 24, no. 11, pp. 1097–1100, 1997.
- [11] P. Hansen, N. Mladenović, and J. A. Moreno Pérez, "Variable neighbourhood search: methods and applications," *Annals of Operations Research*, vol. 175, pp. 367–407, 2010.
- [12] G. Andersson, "An approximation algorithm for Max  $p$ -Section," *Lecture Notes in Computer Science*, vol. 1563, pp. 237–247, 1999.
- [13] D. R. Gaur, R. Krishnamurti, and R. Kohli, "The capacitated max  $k$ -cut problem," *Mathematical Programming*, vol. 115, no. 1, pp. 65–72, 2008.
- [14] A.-F. Ling, "Approximation algorithms for Max 3-section using complex semidefinite programming relaxation," in *Combinatorial Optimization and Applications*, vol. 5573 of *Lecture Notes in Computer Science*, pp. 219–230, 2009.

*Research Article*

## **Opposition-Based Barebones Particle Swarm for Constrained Nonlinear Optimization Problems**

**Hui Wang**

*School of Information Engineering, Nanchang Institute of Technology, Nanchang 330099, China*

Correspondence should be addressed to Hui Wang, wanghui.cug@yahoo.com.cn

Received 29 December 2011; Revised 9 April 2012; Accepted 10 May 2012

Academic Editor: Jianming Shi

Copyright © 2012 Hui Wang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a modified barebones particle swarm optimization (OBPSO) to solve constrained nonlinear optimization problems. The proposed approach OBPSO combines barebones particle swarm optimization (BPSO) and opposition-based learning (OBL) to improve the quality of solutions. A novel boundary search strategy is used to approach the boundary between the feasible and infeasible search region. Moreover, an adaptive penalty method is employed to handle constraints. To verify the performance of OBPSO, a set of well-known constrained benchmark functions is used in the experiments. Simulation results show that our approach achieves a promising performance.

### **1. Introduction**

Many engineering problems can be converted to constrained optimization problems. The aim of constrained optimization is to find a feasible solution with minimized cost (In this paper, we only consider minimization problems). A general constrained minimization optimization problem can be defined as follows.

$$\text{Minimize } f(x) \quad (1.1)$$

subject to

$$\begin{aligned} g_j(x) &\leq 0, \quad j = 1, 2, \dots, q, \\ h_j(x) &= 0, \quad j = q + 1, 2, \dots, m, \end{aligned} \quad (1.2)$$

where  $g(x)$  is inequality constraint,  $h(x)$  is the equality constraint,  $m$  is the number of constraints,  $q$  is the number of inequality constraints, and  $m - q$  is the number of equality constraints.

Particle swarm optimization (PSO) is a population-based stochastic search algorithm developed by Kennedy and Eberhart [1]. Although PSO shares many similarities with evolutionary algorithms (EAs), the standard PSO does not use evolution operators such as crossover and mutation. For PSO's simple concept, easy implementation, and effectiveness, it has been widely applied to many optimization areas.

Although PSO has shown a good performance over many optimization problems, it does not work well when solving complex problems. Especially for constrained optimization problems, the standard PSO could hardly search promising solutions. The possible reason is that constrained optimization problems are usually multimodal and having some constraints. PSO could easily fall into local minima and hardly search feasible solutions. To enhance the performance of PSO on constrained optimization problems, many improved PSO variants have been proposed in the past several years.

Meng et al. [2] used a quantum-inspired PSO (QPSO) to solve constrained economic load dispatch. The QPSO shows stronger search ability and quicker convergence speed, not only because of the introduction of quantum computing theory, but also due to two novel strategies: self-adaptive probability selection and chaotic sequences mutation. Coelho [3] presented a novel quantum-behaved PSO (QPSO) to solve constrained engineering design problems. The proposed approach embedded a Gaussian mutation into QPSO to prevent premature convergence to local minima. Sun et al. [4] proposed an improved vector PSO (IVPSO) based on multidimensional search, in which a simple constraint-preserving method is used to handle constraints. Liu et al. [5] used a hybrid PSO called PSO-DE to solve constrained numerical and engineering optimization problems. The PSO-DE integrates PSO with differential evolution (DE) to obtain a good performance. Venter and Haftka [6] proposed a new method to solve constrained optimization problems. The constrained, single objective optimization problem is converted into an unconstrained, biobjective optimization problem that is solved using a multiobjective PSO algorithm. Lu and Chen [7] presented an enhanced PSO by employing a dynamic inertia weight to avoid premature convergence. The inertia weight of every individual is dynamically controlled by the Euclidean distance between individual and the global best individual. Daneshyari and Yen [8] proposed a cultural-based constrained PSO to incorporate the information of the objective function and constraint violation. The archived information facilitates communication among swarms in the population space and assists in selecting the leading particles in three different levels: personal, swarm, and global levels.

There have been many modifications to the original PSO algorithm to improve the efficiency and robustness of the search. Although these modified PSO variants have shown good search abilities, their performance greatly depends on the control parameters in the velocity updating model, such as inertia weight ( $w$ ) and acceleration coefficients ( $c_1$  and  $c_2$ ). Recently, a parameter-free PSO, known barebones PSO (BPSO) [9], used Gaussian normal distribution to update the particles in the population. It does not involve inertia weight, acceleration coefficients, and velocity. Its performance has been found to be competitive, and a number of BPSO algorithms have been proposed in the past several years. Omran et al. [10] incorporate the idea of BPSO into DE. Krohling and Mendel [11] employed a jump strategy in BPSO to avoid premature convergence. Motivated by the idea of BPSO, this paper presents an improved BPSO, namely OBPSO, to solve constrained nonlinear optimization problems. In OBPSO, opposition-based learning (OBL) concept [12] is used for population

initialization and generation jumping. To verify the performance of OBPSO, a set of well-known constrained benchmark problems are used in the experiments. Results obtained by the proposed OBPSO are compared with those in the literature and discussed.

The rest of the paper is organized as follows. In Section 2, the standard PSO and barebones PSO are briefly introduced. Section 3 describes our proposed approach. Section 4 presents experimental simulations, results, and discussions. Finally, the work is concluded in Section 5.

## 2. Belief Descriptions of PSO and Barebones PSO

In traditional PSO, a member in the swarm, called a particle, represents a potential solution in the  $D$ -dimensional search space. Each particle has two vectors: velocity and position. It is attracted by its previous best particle ( $pbest$ ) and the global best particle ( $gbest$ ) during the evolutionary process. The velocity  $v_{ij}$  and position  $x_{ij}$  of the  $j$ th dimension of the  $i$ th particle are updated according to (2.1) [13]:

$$\begin{aligned} v_{ij}(t+1) &= w \cdot v_{ij}(t) + c_1 \cdot \text{rand1}_{ij} \cdot (pbest_{ij}(t) - x_{ij}(t)) \\ &\quad + c_2 \cdot \text{rand2}_{ij} \cdot (gbest_j(t) - x_{ij}(t)), \\ x_{ij}(t+1) &= x_{ij}(t) + v_{ij}(t+1), \end{aligned} \tag{2.1}$$

where  $i = 1, 2, \dots, N$  is the particle's index,  $N$  is the population size,  $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$  is the position of the  $i$ th particle;  $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$  represents the velocity of the  $i$ th particle; the  $pbest_i = (pbest_{i1}, pbest_{i2}, \dots, pbest_{iD})$  is the best previous position yielding the best fitness value for the  $i$ th particle;  $gbest = (gbest_1, gbest_2, \dots, gbest_D)$  is the global best particle found by all particles so far. The parameter  $w$ , called inertia factor, which is used to balance the global and local search abilities of particles [13],  $\text{rand1}_{ij}$  and  $\text{rand2}_{ij}$  are two random numbers generated independently within the range of  $[0, 1]$ ,  $c_1$  and  $c_2$  are two learning factors which control the influence of the social and cognitive components, and  $t = 1, 2, \dots$  indicates the iteration number.

Recently, Kennedy [9] developed the barebones PSO (BPSO). This new version of PSO eliminates the velocity term, and the position is updated as follows.

$$x_{ij}(t+1) = G\left(\frac{gbest_j(t) + pbest_{ij}(t)}{2}, |gbest_j(t) - pbest_{ij}(t)|\right), \tag{2.2}$$

where  $x_{ij}(t+1)$  is the position of the  $i$ th particle in the population, and  $G$  represents a Gaussian distribution with mean  $(gbest_j(t) + pbest_{ij}(t))/2$  and standard deviation  $|gbest_j(t) - pbest_{ij}(t)|$ .

Note that the particle positions are sampled by the above Gaussian distribution. The BPSO facilitates initial exploration, due to large deviation (initially,  $pbest$  will be far from the  $gbest$ ). As the number of generation increases, the deviation approaches to zero, by focussing on exploitation of the  $pbest$  and  $gbest$  [14].

### 3. Opposition-Based Barebones PSO (OBPSO)

#### 3.1. Opposition-Based Learning

Opposition-based learning (OBL) developed by Tizhoosh [12] is a new concept in computational intelligence. It has been proven to be an effective concept to enhance various optimization approaches [15–17]. When evaluating a solution  $x$  to a given problem, simultaneously computing its opposite solution will provide another chance for finding a candidate solution which is closer to the global optimum.

Let  $X = (x_1, x_2, \dots, x_D)$  be a candidate solution in a  $D$ -dimensional space, where  $x_1, x_2, \dots, x_D \in R$  and  $x_j \in [a_j, b_j]$ ,  $j \in 1, 2, \dots, D$ . The opposite solution  $X^* = (x_1^*, x_2^*, \dots, x_D^*)$  is defined by [15]

$$x_j^* = a_j + b_j - x_j. \quad (3.1)$$

By staying within variables' interval static boundaries, we would jump outside of the already shrunken search space and the knowledge of the current converged search space would be lost. Hence, we calculate opposite particles by using dynamically updated interval boundaries  $[a_j(t), b_j(t)]$  as follows [15].

$$x_{ij}^* = a_j(t) + b_j(t) - x_{ij}, \quad (3.2)$$

$$a_j(t) = \min(x_{ij}(t)), \quad b_j(t) = \max(x_{ij}(t)), \quad (3.3)$$

$$\begin{aligned} x_{ij}^* &= \text{rand}(a_j(t), b_j(t)), & \text{if } x_{ij}^* < x_{\min} \\ && \quad \parallel x_{ij}^* > x_{\max}, \\ i &= 1, 2, \dots, N, \quad j = 1, 2, \dots, D, \end{aligned} \quad (3.4)$$

where  $x_{ij}$  is the  $j$ th position element of the  $i$ th particle in the population,  $x_{ij}^*$  is the opposite particle of  $x_{ij}$ ,  $a_j(t)$  and  $b_j(t)$  are the minimum and maximum values of the  $j$ th dimension in current search space, respectively,  $\text{rand}(a_j(t), b_j(t))$  are random numbers within  $[a_j(t), b_j(t)]$ ,  $[x_{\min}, x_{\max}]$  is the box-constraint of the problem, and  $N$  is the population size, and  $t = 1, 2, \dots$  indicates the generations.

#### 3.2. Adaptive Constraint Handling

To handle the constraints in solving constrained optimization problems, this paper employs an adaptive penalty method (APM) which was early considered in [18–20]. It aims to help users avoid manually defining the coefficients of penalty functions. In the APM, each constraint of the candidate solutions is monitored. If a constraint seems to be more difficult to satisfy, then a larger penalty coefficient is added.

For each candidate solution, its  $j$ th constraint violation  $cv_j(x)$  is computed as follows:

$$cv_j(x) = \begin{cases} |h_j(x)| \\ \min\{0, g_j(x)\}, \end{cases} \quad (3.5)$$

where  $h_j(x)$  is the  $j$ th equality constraint, and  $g_j(x)$  is the  $j$ th inequality constraint.

The fitness value of candidate solution is defined by

$$F(x) = \begin{cases} f(x), & \text{if } x \text{ is feasible} \\ \hat{f}(x) + \sum_{j=1}^m (k_j \cdot cv_j(x)), & \text{otherwise,} \end{cases} \quad (3.6)$$

where  $m$  is the number of constraints,  $k_j$  is a penalty coefficient for each constraint, and  $\hat{f}(x)$  is defined by

$$\hat{f}(x) = \begin{cases} f(x), & \text{if } \bar{f}(x) \text{ is better than } f(x) \\ \bar{f}(x), & \text{otherwise,} \end{cases} \quad (3.7)$$

where  $\bar{f}(x)$  is the average objective function values in the current swarm, and it is computed as

$$\bar{f}(x) = \frac{1}{N} \cdot \sum_{i=1}^N f(x_i), \quad (3.8)$$

where  $N$  is the population size.

For the penalty coefficient  $k_j$ , it determines the scaled factor of the  $j$ th constraint. Every generation, the  $k_j$  is adaptively adjusted as follows.

$$k_j = \left| \bar{f}(x) \right| \cdot \frac{\overline{cv}_j(x)}{\sum_{l=1}^m (\overline{cv}_l(x))^2}, \quad (3.9)$$

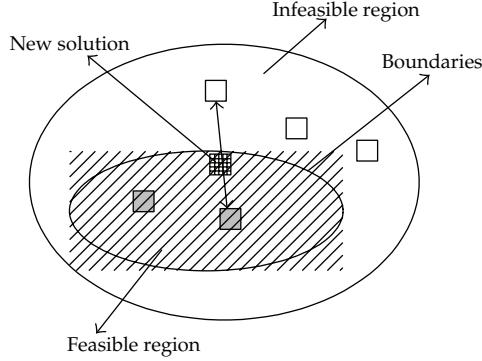
where  $\overline{cv}_l(x)$  is the violation of the  $l$ th constraint averaged over the current swarm, and it is computed by

$$\overline{cv}_j(x) = \frac{1}{N} \cdot \sum_{i=1}^N cv_{ij}(x), \quad (3.10)$$

where  $cv_{ij}(x)$  is the violation of  $i$ th particle on the  $j$ th constraint.

### 3.3. Boundary Search Strategy

For constrained optimization problems, the solution search space can be divided into two parts: feasible space and infeasible space. In some cases, the global optimum is located at the boundaries of feasible space. It is difficult to find this kind of solutions. Because many algorithms can hardly judge the boundaries of feasible space. To tackle this problem, this paper employs a boundary search strategy as follows.



**Figure 1:** The boundary search strategy.

If the current population contains  $M$  feasible solutions and  $N - M$  infeasible solutions ( $M < N$ ), then we randomly select one feasible solution  $x$  and one infeasible solution  $y$ . Based on  $x$  and  $y$ , a new solution  $z$  is generated by

$$z_j = 0.5 \cdot x_j + 0.5 \cdot y_j, \quad (3.11)$$

where  $j = 1, 2, \dots, D$ .

Note that the boundary search strategy works when the current population contains feasible and infeasible solutions. Figure 1 clearly illustrates the boundary search. As seen, if the new solution  $z$  is infeasible, then replace  $y$  with  $z$ . Because  $z$  is nearer to the boundary than  $y$ . If  $z$  is feasible, then replace  $x$  with  $z$ . Because  $z$  is nearer to the boundary than  $x$ .

### 3.4. The Proposed Approach

The proposed approach (OBPSO) uses a similar procedure to that of opposition-based differential evolution (ODE) for opposition-based population initialization and dynamic opposition [15]. To handle the constraints, we define a new fitness evaluation function as described in (3.6). Moreover, we also use a boundary search strategy to find the solutions located at the margin of the feasible region. The framework of OBPSO is shown in Algorithm 1, where  $P$  is the current population,  $OP$  is the population after using OBL,  $P_i$  is the  $i$ th particle in  $P$ ,  $OP_i$  is the  $i$ th particle in  $OP$ ,  $p_o$  is the probability of opposition,  $N$  is the population size,  $D$  is the dimension size,  $[a_j(t), b_j(t)]$  is the interval boundaries of current population, FEs is the number of fitness evaluations, and MAX.FEs is the maximum number of fitness evaluations.

## 4. Experimental Verifications

### 4.1. Test Problems

To verify the performance our proposed approach, we employ a set of 13 benchmark functions from the literature [21, 22]. The main characteristics of these benchmark functions are summarized in Table 1. For specific definitions of these functions, please refer to [23].

```

1 Uniform randomly initialize each particle in the swarm;
2 Initialize  $p_{best}$  and  $g_{best}$ ;
3 While FEs  $\leq$  MAX_FEs do
    /* Barebones PSO
4     for  $i = 1$  to  $N$  do
5         Calculate the position of the  $i$ th particle  $P_i$  according to (2.2);
6         Calculate the fitness value of  $P_i$  according to (3.6);
7         FEs++;
8     end
    /* Opposition-based learning
9     if  $\text{rand}(0,1) \leq p_o$  then
10        Update the dynamic interval boundaries  $[a_j(t), b_j(t)]$  in  $P$  according to (3.3);
11        for  $i = 1$  to  $N$  do
12            Generate the opposite particle of  $P_i$  according to (3.2)
13            Calculate the fitness value of  $P_i$  according to (3.6)
14            FEs++;
15        end
16        Select  $N$  fittest particles from  $\{P, OP\}$  as current population  $P$ ;
17    end
    /* Boundary search strategy
18    if  $M < N$  then
19         $K = \min\{M, N - M\}$ 
20        for  $i = 1$  to  $K$  do
21            Randomly select a feasible solution  $x$  and an infeasible solution  $y$ ;
22            Generate a new solution  $z$  according to (3.11);
23            Calculate the fitness value of  $z$  according to (3.6);
24            FEs++;
25            if  $z$  is feasible then
26                 $x = z$ ;
27            end
28            else
29                 $y = z$ ;
30            end
31        end
32    end
33    Update  $p_{best}$  and  $g_{best}$ ;
34 end

```

Algorithm 1: The proposed OBPSO algorithm.

#### 4.2. Comparison of OBPSO with Similar PSO Algorithms

In this section, we compare the performance of OBPSO with standard PSO, barebones PSO (BPSO), and OBPSO without boundary search strategy (OBPSO-1). To have a fair comparison, the same settings are used for common parameters. The population size  $N$  is set to 40. For PSO, The inertia weight  $w$  is set to 0.72984. The acceleration coefficients  $c_1$  and  $c_2$  are set to 1.49618. The maximum velocity  $V_{\max}$  was set to the half range of the search space for each dimension. For OBPSO-1 and OBPSO, the probability of opposition  $p_o$  is set to 0.3. For each test functions, both OBPSO and PSO stop running when the number of iterations reaches to 1,000.

Table 2 presents average results of the four PSO algorithms over 30 runs, where Mean represents the mean best function values. As seen, PSO outperforms BPSO on only one

**Table 1:** Summary of main characteristics of benchmark problems, where  $F$  means the feasible region,  $S$  is the whole search region, NE represents nonlinear equality, NI indicates nonlinear inequality, LI means linear inequality, and  $\alpha$  is the number of active constraints at optimum.

Problems	$D$	Type	$ F / S $	LI	NE	NI	$\alpha$
G01	13	Quadratic	0.011%	9	0	0	6
G02	20	Nonlinear	99.990%	1	0	1	1
G03	10	Polynomial	0.002%	0	1	0	1
G04	5	Quadratic	52.123%	0	0	6	2
G05	4	Cubic	0.000%	2	3	0	3
G06	2	Cubic	0.006%	0	0	2	2
G07	10	Quadratic	0.000%	3	0	5	6
G08	2	Nonlinear	0.856%	0	0	2	0
G09	7	Polynomial	0.512%	0	0	4	2
G10	8	Linear	0.001%	3	0	3	3
G11	2	Quadratic	0.000%	0	1	0	1
G12	3	Quadratic	4.779%	0	0	9 <sup>3</sup>	0
G13	5	Exponential	0.000%	0	3	0	3

problem G05. BPSO achieves better results than PSO on 7 problems. Both of them can find the global optimum on 5 problems. It demonstrates that the barebones PSO is better than standard PSO for these problems.

For the comparison of OBPSO with BPSO, both of them obtain the same results on 6 problems. For the rest 7 problems, OBPSO performs better than BPSO. It demonstrates that the opposition-based learning is helpful to improve the quality of solutions.

To verify the effects of the boundary search strategy, we compare the performance of OBPSO with OBPSO-1. For the OBPSO-1 algorithm, it does not use the proposed boundary search strategy. As seen, OBPSO outperforms OBPSO-1 on 6 problems, while they obtain the same results for the rest 7 problems. These results demonstrate the effectiveness of the boundary search strategy.

Figure 2 shows the evolutionary processes on four representative problems. It can be seen that OBPSO converges faster than other 3 PSO algorithms. The OBPSO-1 shows faster convergence rate than PSO and BPSO. This confirms that the opposition-based learning is beneficial for accelerating the evolution [15].

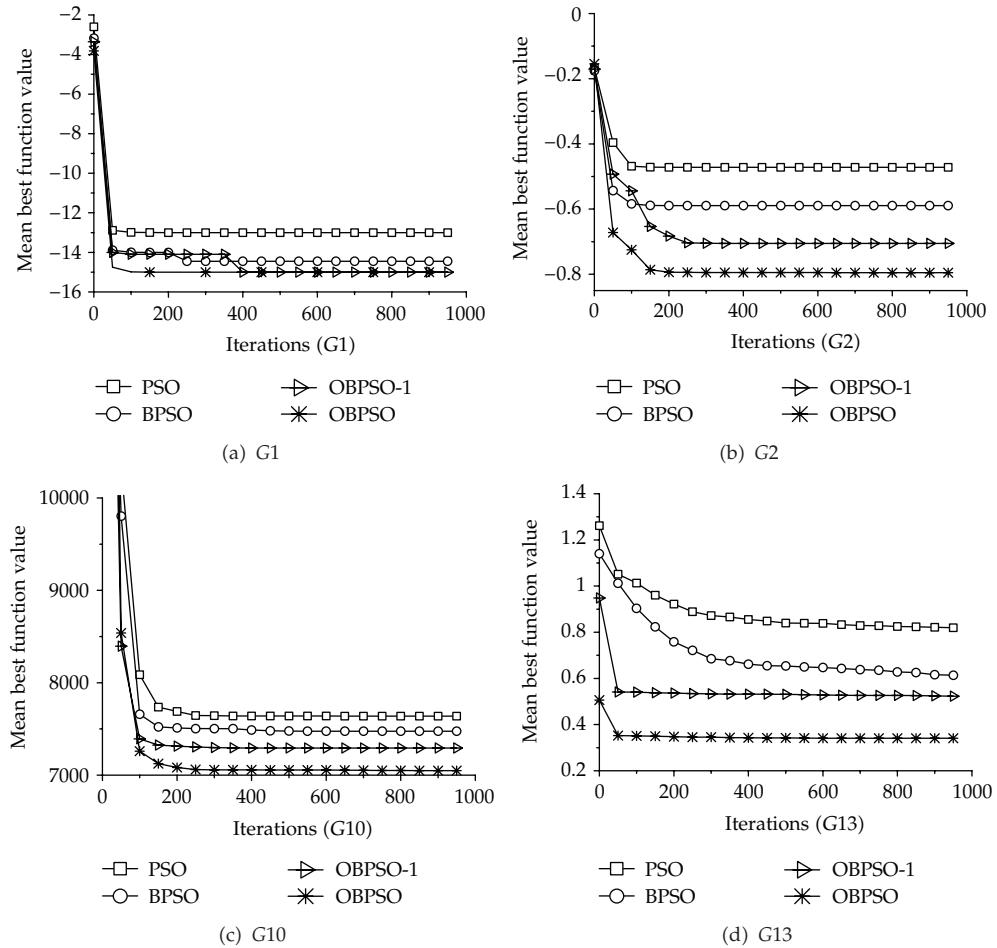
#### 4.3. Comparison of OBPSO with Other State-of-the-Art PSO Variants

In this section, we compare the performance of OBPSO with three other PSO variants on the test suite. The involved algorithms and parameter settings are listed as follows.

- (i) New vector PSO (NVPSO) [4].
- (ii) Dynamic-objective PSO (RVPSO) [21].
- (iii) Self-adaptive velocity PSO (SAVPSO) [22].
- (iv) Our approach OBPSO.

**Table 2:** Mean best function values achieved by PSO, BPSO, OBPSO-1, and OBPSO, and the best results among the four algorithms are shown in bold.

Functions	Optimum	PSO mean	BPSO mean	OBPSO-1 mean	OBPSO mean
G01	-15	-13.013	-14.46	<b>-15</b>	<b>-15</b>
G02	-0.803619	-0.47191	-0.58944	-0.70536	<b>-0.79973</b>
G03	-1.0	-1.00468	-1.00339	-1.00287	<b>-1.00126</b>
G04	-30665.539	<b>-30665.5</b>	<b>-30665.5</b>	<b>-30665.5</b>	<b>-30665.5</b>
G05	5126.4981	5213.14	5228.32	5154.76	<b>5126.68</b>
G06	-6961.814	<b>-6961.81</b>	<b>-6961.81</b>	<b>-6961.81</b>	<b>-6961.81</b>
G07	24.306	25.9185	25.3492	24.8576	<b>24.4196</b>
G08	-0.095825	<b>-0.095825</b>	<b>-0.095825</b>	<b>-0.095825</b>	<b>-0.095825</b>
G09	680.630	680.679	<b>680.630</b>	<b>680.630</b>	<b>680.630</b>
G10	7049.248	7639.4	7474.5	7292.82	<b>7049.2605</b>
G11	0.750	<b>0.749</b>	<b>0.749</b>	<b>0.749</b>	<b>0.749</b>
G12	-1.0	<b>-1.0</b>	<b>-1.0</b>	<b>-1.0</b>	<b>-1.0</b>
G13	0.05395	0.819146	0.61415	0.52312	<b>0.33837</b>



**Figure 2:** The evolutionary processes of PSO, BPSO, OBPSO-1, and OBPSO on four selected problems.

**Table 3:** Comparison results of OBPSO with other three PSO algorithms, where “ $w/t/l$ ” means that OBPSO wins in  $w$  functions, ties in  $t$  functions, and loses in  $l$  functions, compared with its competitors. The best results among the four algorithms are shown in bold.

Functions	Optimum	NVPSO [4] mean	RVPSO [21] mean	SAVPSO [22] mean	OBPSO mean
G01	-15	-13.871875	-14.7151	-14.4187	<b>-15</b>
G02	-0.803619	-0.336263	-0.74057	-0.413257	<b>-0.79973</b>
G03	-1.0	-1.00484	-1.0034	-1.0025	<b>-1.00126</b>
G04	-30665.539	<b>-30665.5</b>	<b>-30665.5</b>	<b>-30665.5</b>	<b>-30665.5</b>
G05	5126.4981	<b>5126.4957</b>	5202.3627	5241.0549	5126.68
G06	-6961.814	<b>-6961.81</b>	<b>-6961.81</b>	<b>-6961.81</b>	<b>-6961.81</b>
G07	24.306	25.1301	24.989	<b>24.317</b>	24.4196
G08	-0.095825	<b>-0.095825</b>	<b>-0.095825</b>	<b>-0.095825</b>	<b>-0.095825</b>
G09	680.630	680.634430	680.653	<b>680.630</b>	<b>680.630</b>
G10	7049.248	7409.065752	7173.2661	7049.2701	<b>7049.2606</b>
G11	0.750	<b>0.749</b>	<b>0.749</b>	<b>0.749</b>	<b>0.749</b>
G12	-1.0	<b>-1.0</b>	<b>-1.0</b>	<b>-1.0</b>	<b>-1.0</b>
G13	0.05395	0.465217	0.552753	0.681123	<b>0.33837</b>
$w/t/l$	—	7/5/1	8/5/0	6/6/1	—

The parameter settings of NVPSO are described in [4]. For RVPSO and SAVPSO, their parameter settings are given in [22]. For OBPSO, we use the same parameter values as described in the previous section. For each test functions, all algorithms stop running when the number of iterations reaches to the maximum value 1,000.

Table 3 presents average results of NVPSO, RVPSO, SAVPSO, and OBPSO over 30 runs, where Mean represents the mean best function values. The comparison results among OBPSO and other algorithms are summarized as  $w/t/l$  in the last row of the table, which means that OBPSO wins in  $w$  functions, ties in  $t$  functions, and loses in  $l$  functions, compared with its competitors.

From the results of Table 3, OBPSO outperforms NVPSO on 7 problems, while NVPSO only achieves better results on a single problem. For the rest 5 problems, both OBPSO and NVPSO can find the global optimum. OBPSO performs better than RVPSO on 8 problems, while both of them obtain the same results for the rest 5 problems. For the comparison of SAVPSO and OBPSO, both of them achieve the same results on 6 problems. For the rest 7 problems, OBPSO wins 6, while SAVPSO wins only 1.

## 5. Conclusion

This paper proposes a modified barebones particle swarm optimization to solve constrained nonlinear optimization problems. The proposed approach is called OBPSO which employs two novel strategies including opposition-based learning and boundary search. Compared to other improved PSO variants, OBPSO is almost a parameter-free algorithm (except for the probability of opposition). Moreover, an adaptive penalty method is used to handle constraints. Experimental verifications on a set of constrained benchmark functions show that OBPSO achieves a promising performance compared to four other PSO variants. The parameter  $p_o$  may affect the performance of OBPSO. To determine the best choice of  $p_o$ , different values of  $p_o$  will be investigated. This will be conducted in the future work.

## Acknowledgments

The authors would like to thank the editor and anonymous reviewers for their detailed and constructive comments that helped them to increase the quality of this work. This work is supported by the Science and Technology Plan Projects of Jiangxi Provincial Education Department (nos. GJJ12641, GJJ12633, and GJJ12307), and the National Natural Science Foundation of China (nos. 61070008, 61165004).

## References

- [1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks. Part 1*, pp. 1942–1948, December 1995.
- [2] K. Meng, H. G. Wang, Z. Y. Dong, and K. P. Wong, "Quantum-inspired particle swarm optimization for valve-point economic load dispatch," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 215–222, 2010.
- [3] L. D. S. Coelho, "Gaussian quantum-behaved particle swarm optimization approaches for constrained engineering design problems," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1676–1683, 2010.
- [4] C. L. Sun, J. C. Zeng, and J. S. Pan, "An new vector particle swarm optimization for constrained optimization problems," in *Proceedings of the International Joint Conference on Computational Sciences and Optimization (CSO '09)*, pp. 485–488, April 2009.
- [5] H. Liu, Z. Cai, and Y. Wang, "Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization," *Applied Soft Computing Journal*, vol. 10, no. 2, pp. 629–640, 2010.
- [6] G. Venter and R. T. Haftka, "Constrained particle swarm optimization using a bi-objective formulation," *Structural and Multidisciplinary Optimization*, vol. 40, no. 1–6, pp. 65–76, 2010.
- [7] H. Lu and X. Chen, "A new particle swarm optimization with a dynamic inertia weight for solving constrained optimization problems," *Information Technology Journal*, vol. 10, no. 8, pp. 1536–1544, 2011.
- [8] 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*, vol. 18, pp. 1–16, 2011.
- [9] J. Kennedy, "Bare bones particle swarms," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS '03)*, pp. 80–87, 2003.
- [10] M. G. H. Omran, A. P. Engelbrecht, and A. Salman, "Bare bones differential evolution," *European Journal of Operational Research*, vol. 196, no. 1, pp. 128–139, 2009.
- [11] R. A. Krohling and E. Mendel, "Bare bones particle swarm optimization with Gaussian or cauchy jumps," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '09)*, pp. 3285–3291, May 2009.
- [12] H. R. Tizhoosh, "Opposition-based learning: a new scheme for machine intelligence," in *Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation (CIMCA '05) and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (IAWTIC '05)*, pp. 695–701, November 2005.
- [13] Y. Shi and R. C. Eberhart, "A modified particle swarm optimizer," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '98)*, pp. 69–73, May 1998.
- [14] A. P. Engelbrecht, "Heterogeneous particle swarm optimization," in *Proceedings of the International Conference on Swarm Intelligence*, pp. 191–202, 2010.
- [15] R. S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-based differential evolution," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 64–79, 2008.
- [16] 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.
- [17] H. Wang, Z. Wu, and S. Rahnamayan, "Enhanced opposition-based differential evolution for solving high-dimensional continuous optimization problems," *Soft Computing*, vol. 15, no. 11, pp. 2127–2140, 2011.
- [18] A. C. C. Lemonge and H. J. C. Barbosa, "An adaptive penalty scheme for genetic algorithms in structural optimization," *International Journal for Numerical Methods in Engineering*, vol. 59, no. 5, pp. 703–736, 2004.

- [19] E. K. da Silva, H. J. C. Barbosa, and A. C. C. Lemonge, "An adaptive constraint handling technique for differential evolution with dynamic use of variants in engineering optimization," *Optimization and Engineering*, vol. 12, no. 1-2, pp. 31–54, 2011.
- [20] X. Pan, Y. Cao, and Q. Pu, "Improved particle swarm optimization with adaptive constraint handling for engineering optimization," *Journal of Information and Computational Science*, vol. 8, no. 15, pp. 3507–3514, 2011.
- [21] H. Y. Lu and W. Q. Chen, "Dynamic-objective particle swarm optimization for constrained optimization problems," *Journal of Combinatorial Optimization*, vol. 12, no. 4, pp. 409–419, 2006.
- [22] H. Y. Lu and W. Q. Chen, "Self-adaptive velocity particle swarm optimization for solving constrained optimization problems," *Journal of Global Optimization*, vol. 41, no. 3, pp. 427–445, 2008.
- [23] J. J. Liang, T. P. Runarsson, E. Mezura-Montes et al., "Problem definitions and evaluation criteria for the CEC 2006, special session on constrained real-parameter optimization," Tech. Rep., Nanyang Technological University, Singapore, 2006.

## Research Article

# A New Hybrid Nelder-Mead Particle Swarm Optimization for Coordination Optimization of Directional Overcurrent Relays

An Liu<sup>1,2</sup> and Ming-Ta Yang<sup>3</sup>

<sup>1</sup> Department of Computer Science and Information Engineering, St. John's University, No. 499, Section 4, Tam King Road, Tamsui District, New Taipei City 25135, Taiwan

<sup>2</sup> Graduate Institute of Computer and Communication Engineering, National Taipei University of Technology, No. 1, Section 3, Chung Hsiao East Road, Taipei 10608, Taiwan

<sup>3</sup> Department of Electrical Engineering, St. John's University, No. 499, Section 4, Tam King Road, Tamsui District, New Taipei City 25135, Taiwan

Correspondence should be addressed to Ming-Ta Yang, mtyang@mail.sju.edu.tw

Received 18 February 2012; Revised 26 April 2012; Accepted 10 May 2012

Academic Editor: Jianming Shi

Copyright © 2012 A. Liu and M.-T. Yang. 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.

Coordination optimization of directional overcurrent relays (DOCRs) is an important part of an efficient distribution system. This optimization problem involves obtaining the time dial setting (TDS) and pickup current ( $I_p$ ) values of each DOCR. The optimal results should have the shortest primary relay operating time for all fault lines. Recently, the particle swarm optimization (PSO) algorithm has been considered an effective tool for linear/nonlinear optimization problems with application in the protection and coordination of power systems. With a limited runtime period, the conventional PSO considers the optimal solution as the final solution, and an early convergence of PSO results in decreased overall performance and an increase in the risk of mistaking local optima for global optima. Therefore, this study proposes a new hybrid Nelder-Mead simplex search method and particle swarm optimization (proposed NM-PSO) algorithm to solve the DOCR coordination optimization problem. PSO is the main optimizer, and the Nelder-Mead simplex search method is used to improve the efficiency of PSO due to its potential for rapid convergence. To validate the proposal, this study compared the performance of the proposed algorithm with that of PSO and original NM-PSO. The findings demonstrate the outstanding performance of the proposed NM-PSO in terms of computation speed, rate of convergence, and feasibility.

## 1. Introduction

Transmission lines are exposed to the environment and stretch long distances, which increases the probability of failure far beyond that of other components of the network, including generators, transformers, and switchgear equipment.

A transmission network is usually divided according to function: (1) transmission lines between various major substations forming the backbone of the network; (2) subtransmission lines connecting substations to load centers or major users; (3) distribution lines between load centers and end users.

Lines from substations or load centers are often distributed in the form of radial feeders. Radial feeders only require the installation of relays and breakers at the ends of the various lines from which the power is sent. When a fault occurs in most radial feeders, the fault current will be greater than the load current with no reverse fault current. As a result, these types of radial feeders can be protected using nondirectional overcurrent relays.

If the protected line is installed with power supplies at both ends, such as in the case of loop networks, the fault current may be fed from the left or right in the event of reverse external failure. In this case, relay malfunctions may occur only if nondirectional overcurrent relays are used for protection, as these relays cannot be coordinated. Directional overcurrent relay (DOCR) is a method to improve protection. DOCR is designed to function only in the event of a unidirectional fault current.

The study of coordination problems in electrical power systems has become increasingly important in recent years. Economic considerations have propelled DOCR into widespread use as the primary protection for distribution systems and as the backup protection for transmission systems. When working with a DOCR system, operators must set time dial setting (TDS) and pick up current ( $I_p$ ) values according to the coordination relationship of the primary/backup (P/B) pairs to fully secure protection for the entire system.

In recent years, several optimization techniques have been proposed for the optimal coordination of DOCRs. Urdaneta et al. applied a minima optimization approach to determine the TDS values for preset  $I_p$  values for fixed and multiple power system network configurations [1]. Abyaneh et al. obtained optimum coordination by considering linear and nonlinear models of relay characteristics and changes in network configuration [2]. Birla et al. demonstrated the simultaneous optimization of all DOCR settings with nonlinear relay characteristics using a sequential quadratic programming method [3]. In [4], a genetic algorithm was selected as the tool to solve the DOCR coordination problem, which included nonlinear constraints. The results of [5] reveal that the advantage of the proposed interval method for the DOCR coordination problem provides robust support against uncertainty in the topology of the network. Bedekar and Bhide used a genetic algorithm and nonlinear programming method to systematically determine initial and final values of the time multiplier and plug settings for optimal DOCR coordination [6].

In general, PSO algorithms are not easily trapped in local optima; however, the convergence rate is slow, and optimization problems with constraints cannot be effectively solved. Zeineldin et al. proposed an approach using a modified particle swarm optimization (PSO) algorithm to calculate the optimal relay settings, formulating the coordination problem as a mixed-integer nonlinear programming problem [7]. In [8], the problem of setting the DOCR was formulated and solved as a linear programming problem; a modified PSO was also applied. The major goal of this study was to investigate the feasibility of applying a Nelder-Mead simplex search method and a particle swarm optimization (NM-PSO) methodology to address the coordination optimization of a DOCR distribution system.

We have divided the remainder of this paper into three sections. The first provides an introduction to the theoretical foundations of the research, involving the modeling of DOCR coordination problems. The proposed optimization algorithm includes a constraint handling method, an NM simplex search method, a PSO algorithm, and an NM-PSO method. We utilized IEEE 8- and 14-bus test systems to verify the feasibility of the proposed algorithm.

The results show that the proposed method, comprising a linear programming (LP) problem and a mixed-integer nonlinear programming (MINLP) problem, is capable of overcoming the relay coordination problem of a power system. The combined approach effectively increases the convergence rate of calculation and enhances the capability of the PSO when processing under constraints. Finally, we discuss the results and draw conclusions.

## 2. DOCR Coordination Problem

The main purpose of the DOCR coordination problem is to determine the TDS and  $I_p$  values of each relay in a power system. The optimal operating times of the primary relays are then minimized, and coordination pairs of the P/B relays and coordination constraints are obtained. The DOCR coordination optimization problem in a power system can be described as follows:

$$\min J = \sum_{i=1}^n w_i t_{ik}, \quad (2.1)$$

where  $n$  is the number of relays in zone  $k$  of a power network, and  $w_i$  is a coefficient indicating the probability of a fault occurring on the  $i$ th line in zone  $k$  of a power network. In general, the value of  $w_i$  is either 1 or 0. The variable  $t_{ik}$  indicates the operating time of relay  $i$  for a close-in fault in zone  $k$ .

The coordination constraints between the primary relay  $i$  and the backup relay(s)  $j$  are as follows:

$$t_{jk} - t_{ik} \geq \text{CTI}, \quad (2.2)$$

where  $t_{jk}$  reveals the operating time of relay  $j$ , and the relay is the backup relay of relay  $i$ . CTI is the minimum coordination time interval; its value ranges from 0.2 to 0.5 s. In this study, a CTI of 0.2 s was chosen.

The function for the nonlinear relay characteristics is based on IEEE standard C37.112-1996 and is represented as follows:

$$t_i = \text{TDS}_i \times \left( \frac{28.2}{(I_{fi}/I_{pi})^2 - 1} + 0.1217 \right), \quad (2.3)$$

where  $\text{TDS}_i$  and  $I_{pi}$  are the time dial setting and the pickup current setting of the  $i$ th relay, respectively.  $I_{fi}$  is the short-circuit fault current passing through the  $i$ th relay.

The constants and exponent in (2.3) define the shape of the extremely inverse trip characteristics.

The results of this research not only describe the methodology of DOCR coordination optimization but also demonstrate the feasibility of the TDS and  $I_p$  settings of the relays. In general, DOCR allows for a continuous TDS value, but a discrete  $I_p$  setting. To satisfy this requirement, this study explored both linear and nonlinear programming for DOCR coordination optimization. The variable TDS is optimized according to a predefined  $I_p$  for each DOCR, and this optimization problem can be viewed as a linear programming (LP) problem.

For the nonlinear programming (NLP) problem, variables TDS and  $I_p$  are optimized for each DOCR. In the LP or NLP problem of the DOCR coordination optimization, the TDS values can range continuously from 0.1000 to 1.1000, and the  $I_p$  values can range discretely between 10 and 1000 with a step size of 1, depending on the close-in fault current for each relay.

### 3. Proposed Optimization Algorithm

PSO is a random optimization technology developed by Eberhart and Kennedy in 1995 [9], who were inspired by simulating the intelligence of swarming bird flocks. PSO shares many similarities with evolutionary computation techniques such as genetic algorithms (GAs) [6]. The problem is initialized with a population of feasible random solutions; however, PSO contains no genetic operations, such as crossover or mutation.

Another important feature of PSO is that each particle has memory. PSO's information sharing mechanism differs greatly from that of GAs. In GAs, chromosomes mutually share information, and therefore, the movement of the population as it approaches the best area is relatively even. In PSO, the possible individual elements of PSO algorithms are called particles. The global best particle gives information to other particles and updates the movement direction and speed of each particle.

Based on the PSO method, we propose the NM-PSO method for solving the constrained optimization problem. The following section introduces the basic principles of NM-PSO, including constraint-handling methods, Nelder-Mead (NM) simplex search, and PSO.

#### 3.1. Constraint-Handling Methods

Constraint handling is a major concern when applying PSO algorithms to solve constrained optimization problems. This is because the traditional search operators of PSO algorithms are blind to constraints. Thus far, the most commonly used constraint handling methods for PSO are the penalty and repair methods.

The gradient-based repair method was addressed by [10, 11]. This method adopts gradient information derived from the constraint set to gradually repair an infeasible solution by directing the infeasible solution toward a feasible area. Because the constraints of the DOCR coordination optimization are not complicated, this method is highly suitable. Furthermore, because DOCR coordination optimization has no equality constraints, equality constraint equations can be ignored. This method is described below.

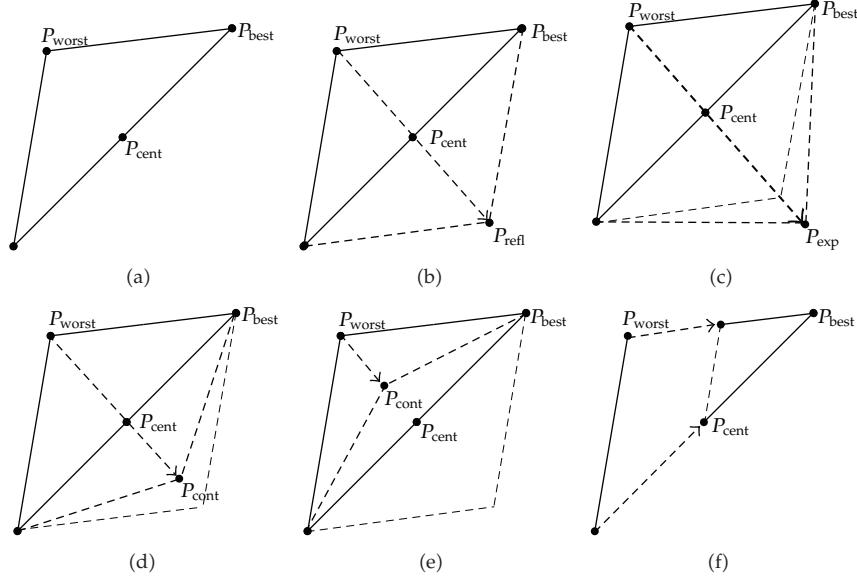
- (1) For a random solution, determine the degree of constraint violation  $\Delta V$  using the following equation:

$$V = [g]_{m \times 1} \Rightarrow \Delta V = [-g_j(x)]_{k \times 1} \quad \text{when } g_j(x) > 0, \quad j = 1, \dots, m, \quad (3.1)$$

where  $V$  is the vector of inequality constraints ( $g$ ), and  $k$  is the degree of constraint violation  $\Delta V$ .

- (2) Compute  $\Delta_x g$ , where  $\Delta_x g$  are the derivatives of these constraints with respect to the solution vector ( $n$  decision variables)

$$\Delta_x V = [\Delta_x g]_{k \times n}, \quad x = 1, \dots, k. \quad (3.2)$$



**Figure 1:** Illustration of the Nelder-Mead simplex method.

- (3) The relationship between changes in the constraint violation  $\Delta V$  and the solution vector  $\Delta x$  is expressed as

$$\Delta V = \Delta_x V \times \Delta x \implies \Delta x = \Delta_x V^{-1} \times \Delta V. \quad (3.3)$$

- (4) Compute the pseudoinverse  $\Delta_x V^{-1}$ .

- (5) Update the solution vector by

$$x^{t+1} = x^t + \Delta x = x^t + \Delta_x V^{-1} \times \Delta V. \quad (3.4)$$

The degree of constraint violation is adjusted according to the above procedure. In this algorithm, a “repair method” rapidly revises infeasible solutions to move them toward a feasible region. The number of constraint violations decreases and quickly vanishes with each iteration. Finally, a solution in the feasible region will be obtained.

### 3.2. The Nelder-Mead Simplex Search Method

When the search space is  $n$ -dimensional, the simplex consists of  $n+1$  solutions [12]. As shown in Figure 1(a), in a two-dimensional search plane, a simplex is a triangle. The fitness of each solution is considered at each step of the Nelder-Mead method, and the worst solution  $P_{\text{worst}}$  is identified. The centroid,  $P_{\text{cent}}$ , of the remaining  $n$  points is computed, and the reflection of  $P_{\text{worst}}$  is determined. This reflection yields a new solution,  $P_{\text{refl}}$ , which replaces  $P_{\text{worst}}$ , as shown in Figure 1(b). If the solution  $P_{\text{refl}}$  produced by this reflection has a higher fitness than any other solution in the simplex, the simplex is further expanded in the direction of  $P_{\text{refl}}$ , and  $P_{\text{worst}}$  is replaced with  $P_{\text{exp}}$ , as shown in Figure 1(c).

On the other hand, if  $P_{\text{refl}}$  has a comparatively low fitness, the simplex is contracted. Contraction can either be outward or inward, depending upon whether  $P_{\text{refl}}$  is better or worse than  $P_{\text{worst}}$ , respectively. The contraction operations (i.e.,  $P_{\text{worst}}$  is replaced with  $P_{\text{cont}}$ ) are shown in Figures 1(d) and 1(e). If neither contraction improves the worst solution in the simplex, the best point in the simplex is computed, and a shrinkage is then performed; all the points of the simplex are moved a little closer towards the best solution ( $P_{\text{best}}$ ), as shown in Figure 1(f).

### 3.3. Particle Swarm Optimization

In the past several years, PSO has been successfully applied in many fields [13, 14]. It has been demonstrated that the results of PSO are superior to other methods. The PSO procedure is reviewed below.

- (1) *Initialization.* It randomly generates a swarm of potential solutions called “particles” and assigns a random velocity to each.
- (2) *Velocity Update.* The particles are then “flown” through hyperspace by updating their own velocity. The velocity update of a particle is dynamically adjusted, subject to its own past flight and those of its companions. The velocity and position of the particles are updated by the following equations:

$$V_{\text{id}}^{\text{new}}(t+1) = c_o \times V_{\text{id}}^{\text{old}}(t) + c_1 \times \text{rand}() \times (p_{\text{id}}(t) - x_{\text{id}}^{\text{old}}(t)) + c_2 \times \text{rand}() \times (p_{\text{gd}}(t) - x_{\text{gd}}^{\text{old}}(t)), \quad (3.5)$$

$$x_{\text{id}}^{\text{new}}(t+1) = x_{\text{id}}^{\text{old}}(t) + V_{\text{id}}^{\text{new}}(t+1), \quad (3.6)$$

$$c_o = 0.5 + \frac{\text{rand}()}{3}, \quad (3.7)$$

where  $c_1$  and  $c_2$  are two positive constants;  $c_0$  is an inertia weight, and  $\text{rand}()$  is a random value between (0, 1). Zahara and Hu suggested  $c_1 = c_2 = 2$  and  $c_0 = [0.5 + (\text{rand}()/2)]$  [15]. However, many experiments have shown that using  $c_0 = [0.5 + (\text{rand}()/3)]$  provides better results. Equation (3.5) illustrates the calculation of a new velocity for each individual. The velocity of each particle is updated according to its previous velocity ( $V_{\text{id}}$ ), the particle’s previous best location ( $p_{\text{id}}$ ), and the global best location ( $p_{\text{gd}}$ ). Particle velocities for each dimension are clamped to a maximum velocity  $V_{\text{max}}$ . Equation (3.6) shows how each particle’s position is updated in the search space.

### 3.4. NM-PSO Method

The NM-PSO optimization method [16] integrates the constraint-handling methods, the Nelder-Mead simplex search method (traditional algorithm), and the PSO algorithm (evolutionary algorithm) [17]. The PSO optimal method resists easily falling into the local best solution, but it requires many particles in an optimal process, which reduces the speed of computation. The Nelder-Mead simplex search method improves the efficiency of PSO due to its capacity for rapid convergence. However, the drawback of this method is that

it easily falls into a local best solution. This drawback is improved by integrating the two algorithms. Combining the two algorithms and the gradient-based repair methods enables feasible optimal solutions to be found that satisfy the constraint conditions [18].

Using the advantages mentioned above, the NM-PSO method clearly overcomes the drawbacks of low convergence speed, the need for more particles, and the inability to deal with constraint conditions to accurately find optimal solutions.

### **3.5. Implementation of Proposed Method**

The following section introduces the NM-PSO algorithm procedure. Assume the problem to be solved is  $n$ -dimensional. First produce  $N$  ( $N \geq 2n+1$ ) particles to form a swarm. For every particle that violates the constraints, use the gradient repair method to direct the infeasible solution toward the feasible region. In most cases, the repair method does move the solution to the feasible region. Arrange the results of the objective function in order of good to bad and divide the  $N$  particles into  $n$  particles, the  $(n+1)$ th particle, and  $N-(n+1)$  particles and then create three groups. First calculate the top  $n$  particles and the  $(n+1)$ th particle using the NM simplex method. The updated best particle is obtained and the result saved. The PSO method adjusts the  $N$  particles by taking into account the position of the  $(n+1)$  best particle. Through the calculation of a simple NM algorithm, the probability of finding the optimal solution was increased. This procedure for adjusting the  $N$  particles involves selection of the global best particle, the selection of the neighboring best particles, and finally the velocity updates. The global best particle of the population is determined according to the sorted fitness values.

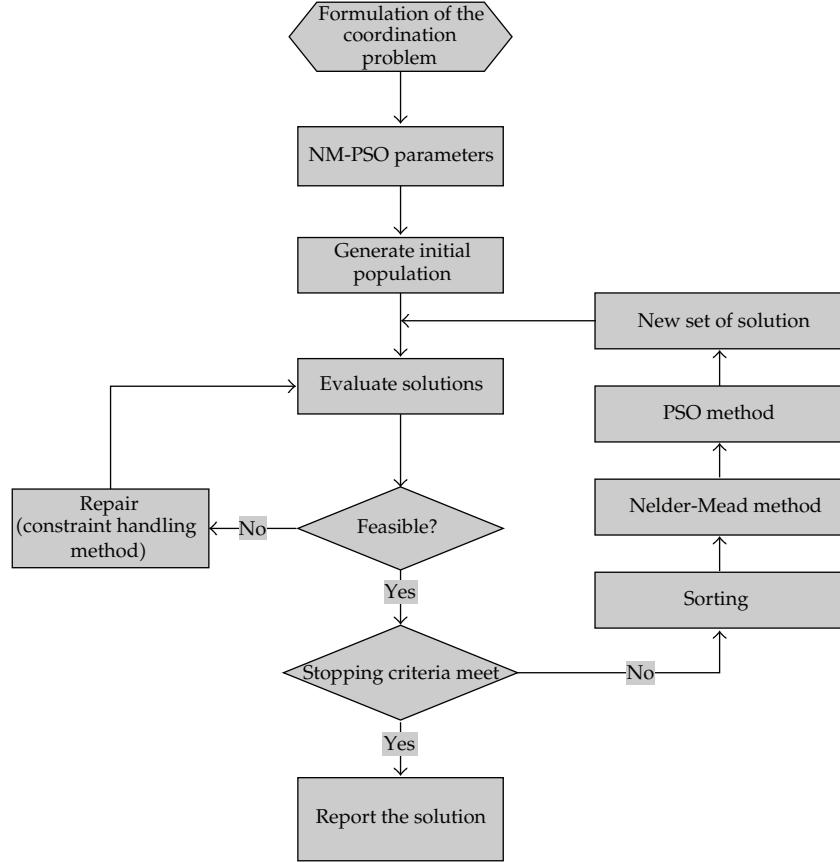
Unlike the original PSO calculation method proposed by [15], which updates the remaining particles ( $N - (n+1)$ ), we use the PSO algorithm to update all of the  $N$  particles. These two PSO algorithms combination NM methods are referred to in this paper as the original NM-PSO method and the proposed NM-PSO method, respectively. Repeat the entire NM-PSO optimization process until the condition is fulfilled. Figure 2 depicts the schematic representation of the proposed NM-PSO. Algorithm 1 shows the pseudocode of the NM-PSO algorithm embedded within the constraint-handling methods.

## **4. Case Study**

The appearance and parameters of the relevant line equipment of two typical test systems are introduced. We discuss the fault current and the corresponding DOCR relationship of the coordination pairs of P/B when a close-in three-phase short fault occurs in transmission lines.

Taking the above two test systems as examples, this study validated the feasibility of the proposed NM-PSO optimization algorithm to solve the DOCR optimal coordination problem. The results were compared with PSO and original NM-PSO algorithm. The results of the comparison demonstrate that the proposed NM-PSO algorithm is clearly better than PSO and original NM-PSO in terms of the objective function, the rate of convergence, and computation speed.

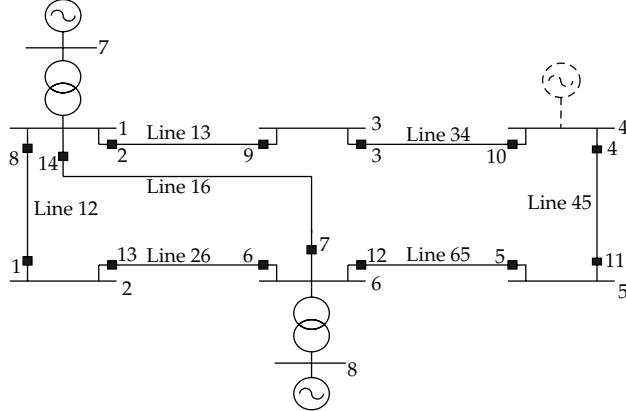
In this study, the multiples  $2 \times n + 1$ ,  $5 \times n + 1$ ,  $10 \times n + 1$ , and  $20 \times n + 1$  were adopted as the number of populations to demonstrate the influence of the number of particles on the proposed algorithm. To observe the process and changes of convergence in the objective function, the number of iterations was set at 300, to highlight the superior performance of the proposed system.



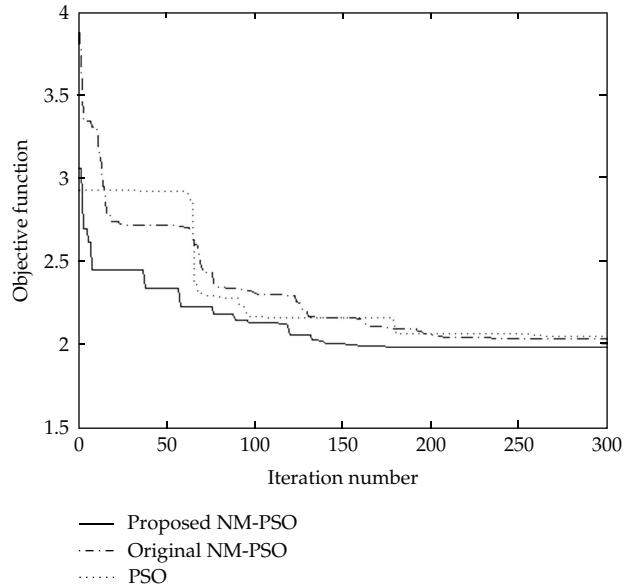
**Figure 2:** Flow chart of the proposed NM-PSO algorithm.

1. **Initialization.** Generate a population of size  $N$  ( $N > (n + 1)$ ).  
Repeat
2. **Constraint handling method**
  - 2.1 **The Gradient Repair Method.** Repair particles that violate the constraints by directing the infeasible solution toward the feasible region.
  - 2.2 Identify solutions (that fulfill the constraint conditions) and arrange them in the order of good to bad.
3. **Nelder-Mead Method.** Apply NM operator to the top  $n + 1$  particles and update the  $(n + 1)$ th particle.
4. **PSO Method.** Apply PSO operator for updating the  $N$  particles.
  - 4.1 **Selection.** Select the global best particle and the neighborhood best particle from the population.
  - 4.2 **Velocity Update.** Apply velocity updates to the  $N$  particles until the condition is fulfilled.

**Algorithm 1:** Pseudocode of the proposed hybrid NM-PSO algorithm.



**Figure 3:** One-line diagram for an IEEE 8-bus test system.



**Figure 4:** Convergence of PSO, original NM-PSO, and proposed NM-PSO to the optimal solution for an IEEE 8-bus test system (LP problem).

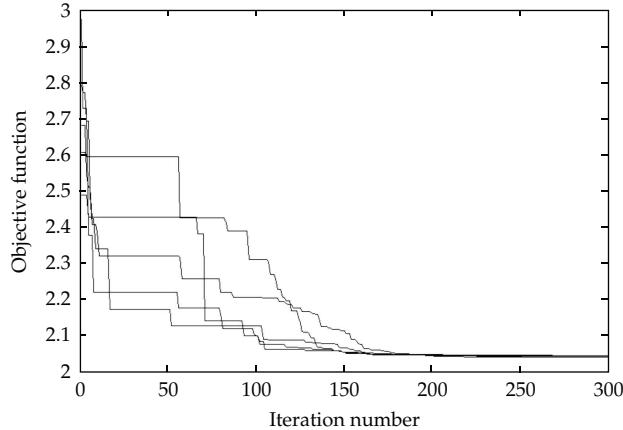
#### 4.1. IEEE 8-Bus Test System

As shown in Figure 3, the 8-bus test system consists of 9 lines, 2 transformers, and 14 DOCRs. All the DOCRs have the IEEE standard inverse-time characteristics mentioned in (2.3) above. The system parameters are the same as in [6]. At bus 4, there is a link to another network modeled by a short-circuit capacity of 400 MVA. The DOCR coordination problem can be formulated as an LP problem or an MINLP problem. Additionally, there are 20 inequality constraints corresponding to each relay pair.

Table 1 illustrates the fault currents of the DOCR coordination pairs of P/B of each phase in the event of a close-in three-phase short fault of the system. If a DOCR coordination

**Table 1:** P/B relays and the close-in fault currents for an IEEE 8-bus test system.

No.	Primary relay		Backup relay	
	No.	Current	No.	Current
1		3230	6	3230
8		6080	9	1160
8		6080	7	1880
2		5910	1	993
9		2480	10	2480
2		5910	7	1880
3		3550	2	3550
10		3880	11	2340
6		6100	5	1200
6		6100	14	1870
13		2980	8	2980
14		5190	9	1160
7		5210	5	1200
14		5190	1	993
7		5210	13	985
4		3780	3	2240
11		3700	12	3700
5		2400	4	2400
12		5890	13	985
12		5890	14	1870

**Figure 5:** Convergence of the NM-PSO for five different random initial populations for an IEEE 8-bus test system (LP problem).

optimization problem with known  $I_p$  values is assumed to be an LP problem, the results obtained using PSO, original NM-PSO, and proposed NM-PSO for the case of 300 iterations and a population size of 141 ( $10 \times n + 1$ , where  $n$  is the number of variables TDS of the 14 relays) are illustrated in Table 2. The results are also compared to those of Linprog (linear programming) obtained using the MATLAB optimization toolbox.

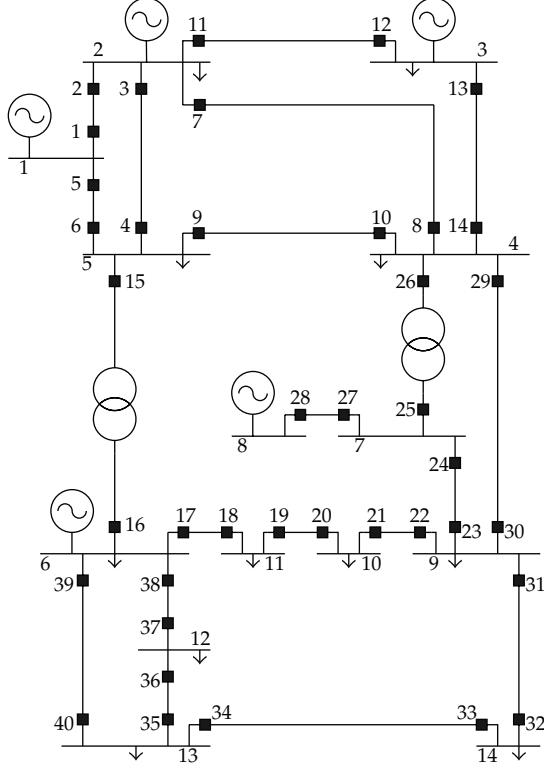
To validate the feasibility of the proposed method, the obtained TDS values and known  $I_p$  values were entered as constraints to obtain the results shown in Table 3. Since

**Table 2:** Optimal settings of the relays for an IEEE 8-bus test system (LP problem).

Algorithm Relay	$I_p$	Linprog TDS	PSO TDS	Original NM-PSO TDS	Proposed NM-PSO TDS
1	600	0.1007	0.1000	0.1000	0.1000
2	800	0.2485	0.2191	0.2260	0.2178
3	500	0.2294	0.2272	0.2369	0.2240
4	800	0.1115	0.1104	0.1234	0.1097
5	600	0.1003	0.1000	0.1075	0.1000
6	500	0.3858	0.3844	0.3882	0.3850
7	600	0.1103	0.1030	0.1233	0.1028
8	500	0.3575	0.3645	0.3587	0.3537
9	600	0.1001	0.1000	0.1007	0.1000
10	500	0.2947	0.3476	0.2963	0.2943
11	600	0.1794	0.2102	0.1958	0.1786
12	500	0.5591	0.6000	0.6194	0.5954
13	600	0.1007	0.1000	0.1000	0.1000
14	800	0.1094	0.1000	0.1030	0.1000
Obj-Fun		1.9640	2.0468	2.0278	1.9783

**Table 3:** Operating time of P/B relays for an IEEE 8-bus test system (LP problem).

Proposed NM-PSO method					
No.	Backup relay		Primary relay		Constraint value
	Operating time	No.	Operating time	No.	
6	0.3134	1	0.1130		0.2004
9	1.0422	8	0.1110		0.9312
7	0.3413	8	0.1110		0.2303
1	1.6338	2	0.1412		1.4926
10	0.3875	9	0.1875		0.2000
7	0.3413	2	0.1412		0.2001
2	0.3551	3	0.1551		0.2000
11	0.3762	10	0.1760		0.2002
5	0.9522	6	0.1203		0.8319
14	0.6439	6	0.1203		0.5236
8	0.3320	13	0.1313		0.2007
9	1.0422	14	0.0808		0.9614
5	0.9522	7	0.0515		0.9007
1	1.6338	14	0.0808		1.5530
13	1.6758	7	0.0515		1.6243
3	0.3585	4	0.1584		0.2001
12	0.3848	11	0.1578		0.2270
4	0.4000	5	0.2000		0.2000
13	1.6758	12	0.1943		1.4815
14	0.6439	12	0.1943		0.4496



**Figure 6:** One-line diagram for an IEEE 14-bus test system.

the proposed method satisfies all constraints (i.e.,  $CTI \geq 0.2$ ), the best coordination setting for DOCR can be efficiently completed.

As expected, the proposed NM-PSO yields better objective function results than PSO and original NM-PSO. Figure 4 shows that proposed NM-PSO nearly reached the global optimum after 189 iterations. The results of this proposed NM-PSO algorithm reveal better convergence.

To analyze the convergence consistency of the proposed NM-PSO algorithm when solving the LP problem in the case of different initial values, this study randomly performed the proposed method five times. As seen in Figure 5, the proposed NM-PSO algorithm can reduce the objective function to the same value after nearly 200 iterations. The convergence of the proposed NM-PSO is evidently not affected by different initial values.

#### 4.2. IEEE 14-Bus Test System

The IEEE 14-bus test system consists of 5 generators, 2 power transformers, 20 transmission lines, and 40 DOCRs, as shown in Figure 6. The system parameters are given in [19]. The voltage level and the power base of this system are 138 kV and 100 MVA, respectively.

It is assumed that the DOCRs all have the standard IEEE inverse-time characteristics as an IEEE 8-bus test system. Table 4 reveals the P/B relay pairs and the corresponding fault

**Table 4:** P/B relays and the close-in fault currents for an IEEE 14-bus test system.

Primary relay		Backup relay		Primary relay		Backup relay		Primary relay		Backup relay	
No.	Current	No.	Current	No.	Current	No.	Current	No.	Current	No.	Current
1	11650	6	654	8	3880	30	188	19	955	17	955
5	12400	2	1980	29	4720	7	1220	18	725	20	725
2	4260	4	750	29	4720	9	1990	22	1930	29	499
2	4260	12	875	29	4720	13	1070	22	1930	24	1160
2	4260	8	723	29	4720	25	449	22	1930	32	280
3	7310	1	3920	6	3830	3	1280	23	1200	21	434
3	7310	12	848	6	3830	10	1990	23	1200	29	499
3	7310	8	689	6	3830	16	560	23	1200	32	281
11	7180	4	725	4	3920	5	1370	30	1810	21	424
11	7180	1	3920	4	3920	16	562	30	1810	24	1130
11	7180	8	695	4	3920	10	1990	30	1810	32	275
7	7330	1	3920	15	4610	5	1360	31	2060	21	428
7	7330	4	716	15	4610	3	1280	31	2060	24	1150
7	7330	12	845	15	4610	10	1970	31	2060	29	494
13	3280	11	1380	9	3260	5	1390	27	2030	26	1230
12	3130	14	1250	9	3260	3	1310	27	2030	23	808
26	4640	9	2080	9	3260	16	569	25	1430	28	633
26	4640	13	1120	16	1490	40	201	25	1430	23	806
26	4640	7	1270	16	1490	18	388	24	1870	26	1230
26	4640	30	179	16	1490	37	51	24	1870	28	634
10	3110	13	1140	17	2210	15	1110	37	572	35	572
10	3110	7	1290	17	2210	37	51	36	781	38	781
10	3110	25	495	17	2210	40	199	35	1480	33	368
10	3110	30	190	39	2400	15	1120	35	1480	39	1110
14	4030	9	2090	39	2400	18	389	34	1390	36	284
14	4030	7	1270	39	2400	37	47	34	1390	39	1110
14	4030	25	489	38	2530	15	1110	40	654	36	285
14	4030	30	188	38	2530	40	191	40	654	33	370
8	3880	9	2090	38	2530	18	386	32	547	34	547
8	3880	25	489	21	564	19	564	33	783	31	783
8	3880	13	1120	20	1310	22	1310	x	x	x	x

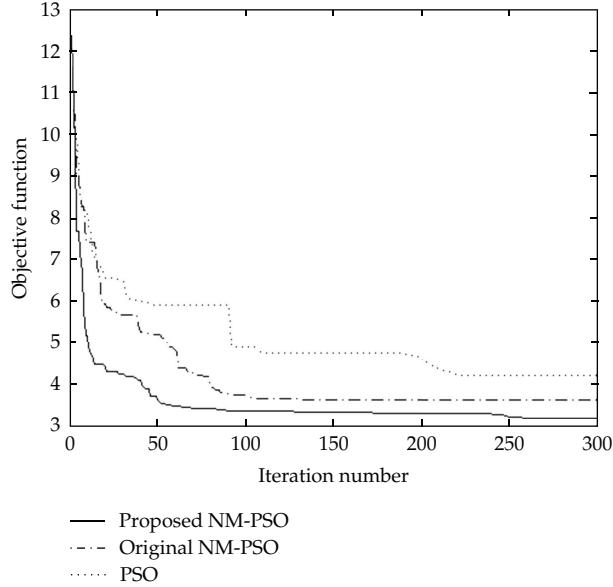
currents passing through them for a close-in fault in this network. There are 92 inequality constraints corresponding to each relay pair.

This study also used PSO, original NM-PSO, and proposed NM-PSO to solve the DOCR coordination optimization of the MINLP problem, which required obtaining the TDS and  $I_p$  values of each DOCR. The results after 300 iterations with a population size of 1601 ( $20 \times n + 1$ , where  $n$  is the number of TDS and  $I_p$  variables of the 40 relays) are shown in Table 5. The integer  $I_p$  can be directly applied in the current intelligent electronic device (IED) settings.

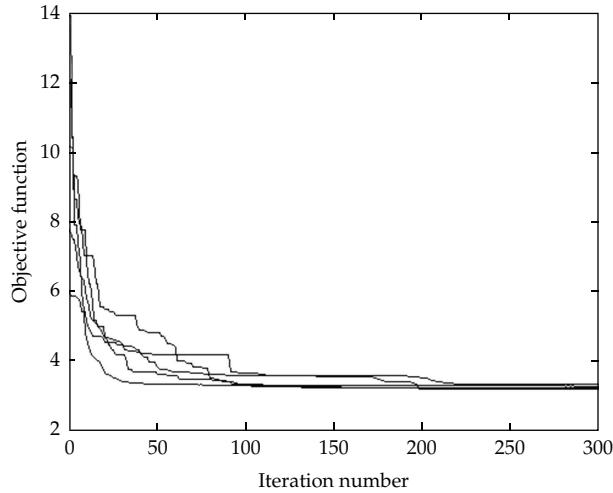
Figure 7 shows the results of the comparison demonstrate that the proposed NM-PSO algorithm is clearly better than PSO and original NM-PSO in terms of the objective function, the rate of convergence, and computation speed.

**Table 5:** Optimal settings of the relays for an IEEE 14-bus test system (MINLP problem).

Algorithm relay	PSO		Original NM-PSO		Proposed NM-PSO	
	$I_p$	TDS	$I_p$	TDS	$I_p$	TDS
1	881	0.1746	807	0.2024	736	0.3038
2	579	0.1000	515	0.1729	540	0.1211
3	408	0.2746	422	0.2108	437	0.1418
4	216	0.1239	261	0.1574	237	0.1184
5	758	0.1059	700	0.1860	700	0.1184
6	296	0.1607	227	0.2694	200	0.1020
7	532	0.2192	435	0.1187	419	0.1718
8	226	0.1335	206	0.1605	270	0.1452
9	444	0.1812	504	0.1454	537	0.1258
10	488	0.3347	528	0.1154	450	0.1523
11	477	0.1717	516	0.2881	550	0.1006
12	352	0.1015	295	0.2357	220	0.1414
13	394	0.1546	355	0.2212	350	0.1307
14	418	0.1000	362	0.2003	358	0.1250
15	333	0.1342	316	0.2404	359	0.1872
16	149	0.1451	126	0.2268	117	0.2302
17	216	0.3880	268	0.2414	177	0.3258
18	110	0.1172	101	0.1418	100	0.1301
19	164	0.2737	168	0.1379	108	0.3402
20	151	0.2322	170	0.1708	211	0.1057
21	100	0.3998	121	0.1308	115	0.1534
22	174	0.6243	339	0.1427	326	0.1473
23	119	0.5556	193	0.1457	150	0.2554
24	147	0.7269	307	0.1543	315	0.1437
25	133	0.2076	127	0.1483	102	0.3358
26	260	0.3107	351	0.1631	399	0.2024
27	150	0.1005	150	0.2364	150	0.1641
28	204	0.1230	127	0.3070	154	0.1779
29	245	0.2288	200	0.2564	200	0.2033
30	150	0.1000	100	0.2557	100	0.2124
31	241	0.1776	247	0.1765	164	0.2053
32	100	0.1066	100	0.1000	100	0.1001
33	177	0.1000	108	0.2252	100	0.1511
34	123	0.2518	100	0.2858	112	0.2285
35	100	0.3861	100	0.2349	100	0.3681
36	146	0.1132	121	0.1311	100	0.1300
37	36	0.1000	10	0.2869	11	0.3143
38	238	0.2182	245	0.1527	249	0.1892
39	268	0.2090	269	0.1850	266	0.1596
40	113	0.1000	100	0.2529	100	0.1485
Obj-Fun	4.2233		3.6362		3.1829	



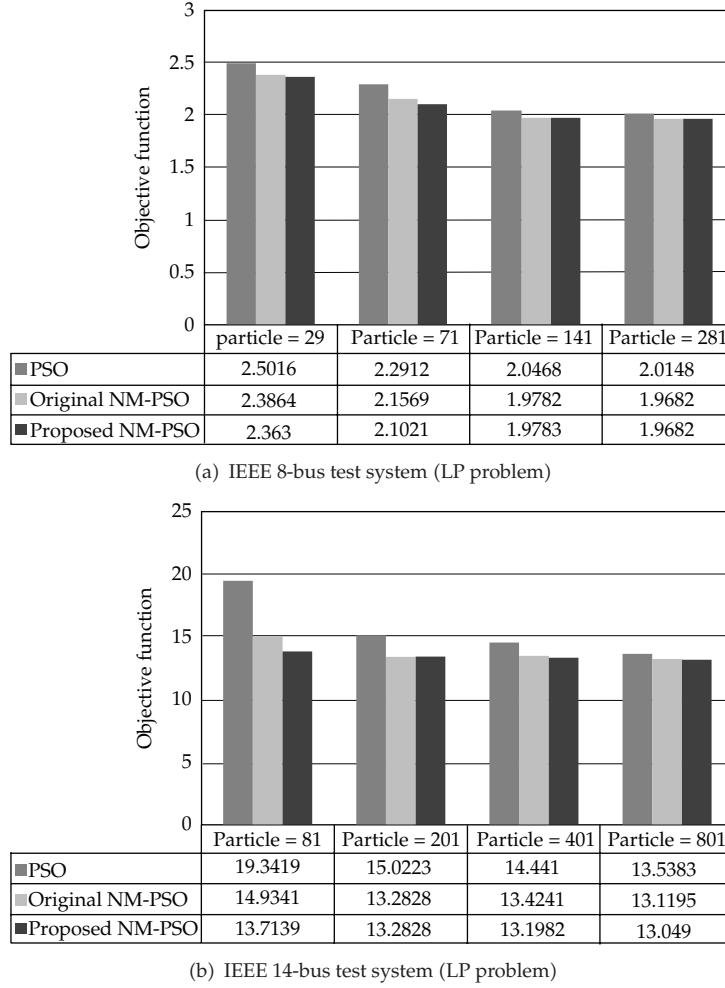
**Figure 7:** Convergence of PSO, original NM-PSO, and proposed NM-PSO to the optimal solution for an IEEE 14-bus test system (MINLP problem).



**Figure 8:** Convergence of the proposed NM-PSO for five different random initial populations for an IEEE 14-bus test system (MINLP problem).

To analyze the consistency in convergence of the proposed NM-PSO when solving MINLP problems, this study randomly executed the proposed method five times. As seen from Figure 8, the proposed NM-PSO can reduce the objective function to almost the same value after approximately 220 iterations. Results show that the convergence of the proposed NM-PSO is not seriously influenced by optimization problems of high complexity.

In addition to convergence rate, we investigated the final convergence values of the objective function (Obj-Fun). Figures 9(a) and 9(b) show the results of LP coordination



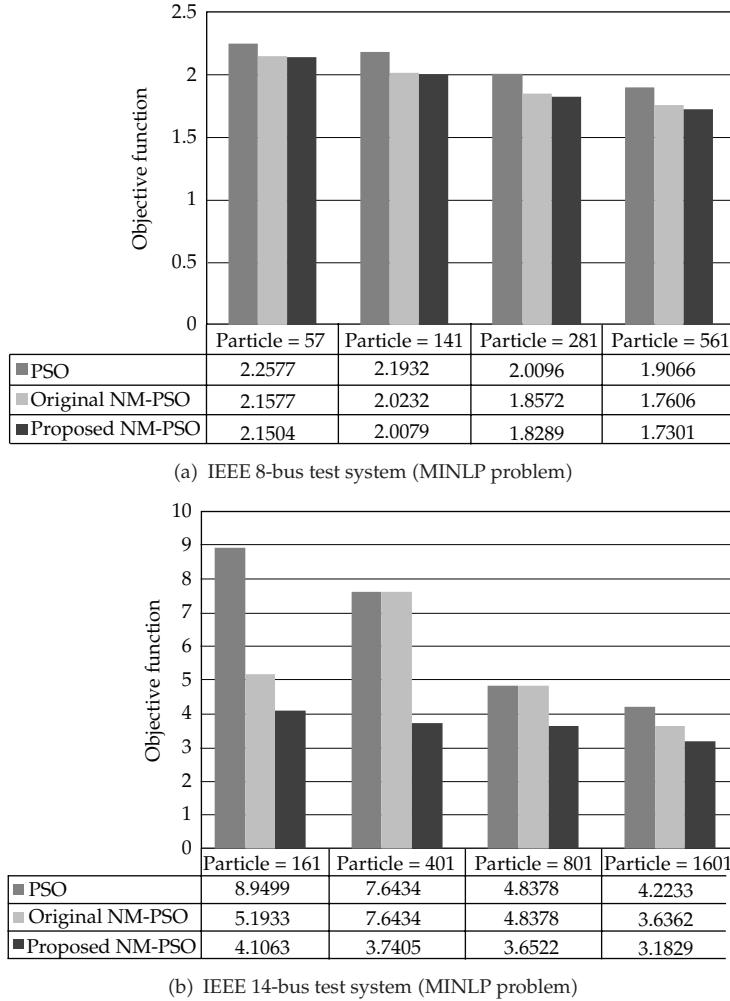
**Figure 9:** Objective function values by PSO, original NM-PSO, and proposed NM-PSO algorithms for the LP problem with  $n = 40$  for four different populations.

problems, for the same number of iterations (300) and for four different populations  $2 \times n + 1$ ,  $5 \times n + 1$ ,  $10 \times n + 1$ , and  $20 \times n + 1$ .

It can clearly be seen that the proposed NM-PSO method results in better Obj-Fun values than the PSO and original NM-PSO algorithm. For the more complicated MINLP problem shown in Figures 10(a) and 10(b), the proposed NM-PSO after 300 iterations results in even better Obj-Fun values in the case of particles  $2 \times n + 1$  than the Obj-Fun values of PSO algorithm in the case of particles  $20 \times n + 1$ . Hence, the proposed method produces better results than the PSO algorithm using fewer particles (less computation time).

## 5. Conclusions

In this paper, the DOCR coordination problem is formulated as a constrained optimization problem. It can be concluded that the proposed NM-PSO optimization algorithm is applicable



**Figure 10:** Objective function values by PSO, original NM-PSO, and proposed NM-PSO algorithms for the MINLP problem with  $n = 80$  for four different populations.

to the DOCR coordination optimization of a distribution system. In contrast to other methods in the literature that only find TDS, the algorithm proposed in this study obtains  $I_p$  and TDS values simultaneously, and the  $I_p$  values can be represented by integers for applications in the IED setting to complete a more comprehensive coordination optimization. The proposed method makes use of the advantages of both the NM and PSO methods, while overcoming the drawbacks associated with these methods. Regardless of whether LP or MINLP is used to solve a coordination optimization problem, we have demonstrated that the proposed algorithm performs better than PSO and original NM-PSO algorithm in terms of computation speed, rate of convergence, and objective function value. The reduction in the DOCR operating time in our results demonstrates that the proposed method can be adopted for determining the optimum settings of DOCRs.

## References

- [1] A. J. Urdaneta, R. Nadira, and L. G. Perez Jimenez, "Optimal coordination of directional overcurrent relays in interconnected power systems," *IEEE Transactions on Power Delivery*, vol. 3, pp. 903–911, 1988.
- [2] H. A. Abyaneh, M. Al-Dabbagh, H. K. Karegar, S. H. H. Sadeghi, and R. A. J. Khan, "A new optimal approach for coordination of overcurrent relays in interconnected power systems," *IEEE Transactions on Power Delivery*, vol. 18, no. 2, pp. 430–435, 2003.
- [3] D. Birla, R. P. Maheshwari, and H. O. Gupta, "A new nonlinear directional overcurrent relay coordination technique, and banes and boons of near-end faults based approach," *IEEE Transactions on Power Delivery*, vol. 21, no. 3, pp. 1176–1182, 2006.
- [4] A. S. Noghabi, J. Sadeh, and H. R. Mashhadi, "Considering different network topologies in optimal overcurrent relay coordination using a hybrid GA," *IEEE Transactions on Power Delivery*, vol. 24, no. 4, pp. 1857–1863, 2009.
- [5] A. S. Noghabi, H. R. Mashhadi, and J. Sadeh, "Optimal coordination of directional overcurrent relays considering different network topologies using interval linear programming," *IEEE Transactions on Power Delivery*, vol. 25, no. 3, pp. 1348–1354, 2010.
- [6] P. P. Bedekar and S. R. Bhide, "Optimum coordination of directional overcurrent relays using the hybrid GA-NLP approach," *IEEE Transactions on Power Delivery*, vol. 26, no. 1, pp. 109–119, 2011.
- [7] H. H. Zeineldin, E. F. El-Saadany, and M. M. A. Salama, "Optimal coordination of overcurrent relays using a modified particle swarm optimization," *Electric Power Systems Research*, vol. 76, no. 11, pp. 988–995, 2006.
- [8] M. M. Mansour, S. F. Mekhamer, and N. E. S. El-Kharbawe, "A modified particle swarm optimizer for the coordination of directional overcurrent relays," *IEEE Transactions on Power Delivery*, vol. 22, no. 3, pp. 1400–1410, 2007.
- [9] R. C. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, pp. 39–43, Nagoya, Japan, October 1995.
- [10] P. Chootinan and A. Chen, "Constraint handling in genetic algorithms using a gradient-based repair method," *Computers and Operations Research*, vol. 33, no. 8, pp. 2263–2281, 2006.
- [11] Y. Dong, J. F. Tang, B. D. Xu, and D. Wang, "An application of swarm optimization to nonlinear programming," *Computers and Mathematics with Applications*, vol. 49, no. 11-12, pp. 1655–1668, 2005.
- [12] J. H. Mathews, *Numerical Methods Using Matlab*, Prentice-Hall, Upper Saddle River, NJ, USA, 2004.
- [13] T. Niknam, H. D. Mojarrad, and M. Nayeripour, "A new hybrid fuzzy adaptive particle swarm optimization for non-convex economic dispatch," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 1, pp. 189–202, 2011.
- [14] C. H. Lin, J. L. Chen, and Z. L. Gaing, "Combining biometric fractal pattern and particle swarm optimization-based classifier for fingerprint recognition," *Mathematical Problems in Engineering*, vol. 2010, Article ID 328676, 14 pages, 2010.
- [15] E. Zahara and C. H. Hu, "Solving constrained optimization problems with hybrid particle swarm optimization," *Engineering Optimization*, vol. 40, no. 11, pp. 1031–1049, 2008.
- [16] E. Zahara and A. Liu, "Solving parameter identification problem by hybrid particle swarm optimization," in *Proceedings of the International MultiConference of Engineers and Computer Scientists 2010 (IMECS '10)*, pp. 36–38, March 2010.
- [17] A. Liu and E. Zahara, "Parameter identification problem using particle swarm optimization," in *Proceedings of the 5th International Conference on Natural Computation (ICNC '09)*, pp. 275–278, Tianjin, China, August 2009.
- [18] Y. T. Kao, E. Zahara, and I. W. Kao, "A hybridized approach to data clustering," *Expert Systems with Applications*, vol. 34, no. 3, pp. 1754–1762, 2008.
- [19] S. Kamel and M. Kodsi, "Modeling and simulation of IEEE 14 bus system with facts controllers," Tech. Rep., 2009.

*Research Article*

## **Evaluating the Performance of Taiwan Homestay Using Analytic Network Process**

**Yi-Chung Hu, Jen-Hung Wang, and Ru-Yu Wang**

*Department of Business Administration, Chung Yuan Christian University, 200 Chung Pei Road, Chungli 32023, Taiwan*

Correspondence should be addressed to Jen-Hung Wang, doraemonponpon@hotmail.com

Received 24 February 2012; Revised 20 April 2012; Accepted 21 April 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Yi-Chung Hu 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.

Homestay industry in Taiwan is not only thriving, but also its operation is moving gradually toward elaboration strategy and in a specialized-operation manner these years. Nevertheless, the evaluation frameworks of the earlier studies were sporadically constructed from an overall perspective of homestays. Moreover, the functions, operational model, and natures of homestays are dissimilar to those of hotels; therefore, if the evaluation criteria of homestays employ the ones of hotels, it would appear to be incoherent and incompatible. This study has accordingly developed and constructed a set of evaluation indicators tailor-made for homestay sector through discussion of literatures and interviewing experts so that the evaluation framework would be more comprehensive and more practical. In the process of interviewing experts, it was discovered that dependences lay on the aspects and criteria. Consequently, this research chose the ANP (analytic network process) to get the weights and, further, to acquire the homestay business performance through fuzzy theory. The result reveals, as regards key aspects, homestay proprietors and customer groups both weight the surroundings of the building and features, service quality, operation, and management most. In respect to overall homestay performance, customer groups consider it has reached the satisfactory level.

### **1. Introduction**

The growing work pressure has made travelling as the best way to relieve stress in a society pursuing high efficiency [1]. In the wake of increased incomes, better living, change of traditional values, and convenient traffic network as well as more leisure time, the demand of recreational activities are swelling in Taiwan. Recreational activities and tourism has steadily become part of modern life [2], which has driven a multispect development of Taiwan domestic tourism and recreational market. The most considerable development and evolvement lie in the accommodation business at tourism attractions, also known as homestay [3]. Homestays are the best Taiwan promoters. If any tourists from all over the

world visit Taiwan and stay in homestays, hosts of homestays not only can introduce them the local cultures and past, but also can act as tour guides as well as good public relation practitioners. Homestays, therefore, play a very essential role. Homestays accordingly are with crucial marketing functions and can drive local economic development. On these grounds, the authors have deemed homestay is an important and discuss-worthy issue.

Based on the World Tourism Organization data, concerning global tourism market, there are approximately seven hundred million people travelling abroad every year. The scale of global tourism sector including both domestic and international tour takes up over 10% of the global GDP (gross domestic product), which has reached an amount of 3.5 trillion US dollars. Tourism 2020 Vision, which was published by the World Tourism Organization in 2001, also indicated that the number of tourists in 2020 would reach 1.56 billion persons, showing a growth of 1.76 times comparing to the 565 million persons in 1995 [4]. In addition, according to the Taiwan Tourism Bureau [5], the total tourism revenue in Taiwan had reached TWD\$514 billion in 2010, and the number of visitors has been growing significantly for a decade. In 2011, the number of in-bound visitors to Taiwan was 6.09 millions, which was an all-time peak and showed a growth of 9.34% compared to the number in 2010. Visitors with "pleasure" purpose were 3.63 million persons, which was up by 11.95%. The ones with business purpose were 984.85 thousand persons, showing an increase of 5.02%. In view of the data above, the number of inbound visitors has kept rising, which would benefit the development of Taiwan industries of tourism, recreation, hospitality, and accommodation.

On top of that, the Taiwan Council for Economic Planning and Development revealed that Taiwan residents' expenditures in recreational and cultural consumption were increasing significantly in 2010 with a real growth rate at 9.88%, up by 0.98 percentage point compared to 2011. The annual growth rate of consumer expenditures in restaurants and lodging has reached 8.11%, up by 0.22 percentage point compared to 2001 [6]. Regarding the expected utility of the development plans of six emerging industries (biotech, tourism, green energy, medical and caring, boutique farming, culture, and creativeness), the Taiwan Council for Economic Planning and Development [7] proposed that the entire revenue of tourism will increase from TWD\$402.2 billion in 2008 to TWD\$550 billion in 2012, including TWD\$ 300 billion from tourism foreign exchange earning. Further, in 2012, it will attract private investment with TWD\$200 billion. In view of the above, it not only demonstrates leisure and entertainment activities certainly carrying weight in Taiwan people's life, but also highlighting immense business potentials in tourism market.

Taiwan is promoting itself to be a "Green Silicon-Valley Island" for sustainable tourism; thus, tourism has become one of the major development targets. Homestay sector has shot up among the trends of recreational farming and ecotourism. According to Taiwan Tourism Bureau [8], the number of approved homestays has swelled from 1,237 in January 2006 to 3,397 in January 2012, with an increase number of 2,160. The homestay sector, on this ground, is not only prosperous, but also with immeasurable business opportunities in tourism sector. In recent years, the operation of homestays has developed into an elaborate and specialized manner. With marketing and media promotion, it has become an important option when people engage in recreational activities and has further turned into an emerging sector with great potential [9]. The Regulations for the Management of Home Stay Facilities [10] have been implemented for years since December 12, 2001, and approved homestays would receive validation certificates. But good ones always mingling with black sheep, some opportunists have entered this sector and often left negative images, which would impact consumers' image about homestays [3]. The Chief Secretary of Taiwan Tourism Bureau, Wayne Liu (the present Deputy Director General), said that there would be a classified

evaluation for Taiwan's approved homestays in the future. The preliminary plan was to categorize them into fourteen groups, which are architecture, room features, landscape, ecology, culture, aboriginal, experiential activities, sport activities, bicycling, hot spring, tour guiding service, language, industrial heritage, and miscellaneous so that people could clearly know each homestay's features [11].

In order to safeguard visitors' lodging interests, choices of recreational commodities and its safety, as well as maintaining the quality of recreational activities while enjoying the vocation, appropriate evaluation items for Taiwan homestay quality validation should be set up by weighing up multi-aspect evaluating items so that applicable, impartial validation assessment and criteria could be accordingly established [12]. The earlier homestay studies were mainly about customer satisfaction [13–15], marketing strategies [16–18], experiential marketing [19–21], operation and management [22–26], and consumer behaviors [27–30], but few were from an overall perspective to construct homestay evaluation framework. Furthermore, the homestay sector is showing a thriving trend and it is hoped that an evaluation mechanism could be formed through this research; notwithstanding, the main distinguishable features of homestay operation different from those of hotels are that homestays lay more stress on (1) inexpensive price and help-your-self service, (2) not emphasizing luxury facilities, but aware of safety issue and hygiene facilities, (3) its service might not be full-scaled, but is with hospitality, local color, and homeyness. What is more important is that homestays make use of nature resources and local cultures in order to let visitors be able to experience the local social customs in person. On top of these, homestays can provide functions like sport, recreation, amusement, and so on to make visitors fully enjoy their leisure time [31]. On these accounts, it is obvious that the homestays and hotels vary in their functions, business model, and natures [3, 12, 32–34]; therefore, if the hotel evaluation criteria are applied indiscriminately to homestays, it would appear to be unconnected and illogical. This research hence deems that it is vital and crucial to develop and construct applicable homestay evaluation indicators, which is one of the motives.

On the other hand, beside homestay validation, such as "hospitable homestays", there is sporadic research into homestay evaluation indicators. For example, Ou and Gian [31] made researches into leisure farming homestay visitor characteristics and needs, whose measure aspects were facilities, service, landscape, operation, and management. Yen et al. [34] studied the homestay evaluation indicators, whose measure aspects were infrastructures, service quality, features of resources, and the association with communities. Chen [35] has constructed evaluation indicators for Taiwan green homestays, whose measure aspects were green building, sustainable landscape engineering, organic farming, environment education, community co-prosperity, and so forth. Shi [12] researched Taiwan homestay quality validation, whose measure aspects were surroundings and facilities, business features, service quality, and community participation. To conclude, it is found that most underlined the hardware, service quality, or exterior settings. It was discovered in the interview process that the forementioned measure aspects are essential to the current external context; however, with respect to evaluation, it should be brought back to the homestay's natures and geist, such as the interaction between the homestay hosts and guests. This study has thus added the aspect of "homestay geist and community co-prosperity" into the final framework and discussed its nuances, which is another motive.

At the end, it found that there is dependence among aspects and criteria of the homestay evaluation. Saaty [36] suggested employing ANP to solve the decision-making issues of dependence. Zadeh [37] introduced fuzzy set theory in 1965, which is to address the uncertainties arising from subjective belief by quantitative methods, and by fuzzifying, it

would provide a better promotability, error tolerance, and is better for applying to the real-world nonlinear system [38].

The following structures of this paper are: Section 2 is the constructing process of the research framework, mainly by literature review and interviewing experts; Section 3 is about methodology, including majority rule, ANP (analytic network process) and Fuzzy Theory; Section 4 is about empirical results; Section 5 is conclusion and suggestions.

## **2. Constructing the Framework**

### **2.1. Preliminary Framework**

This research has organized the evaluation aspects and criteria of the preliminary framework by literature review as Table 1, covering six aspects and 33 criteria.

### **2.2. Final Framework**

After settling the preliminary framework, 3 academic and field experts were invited. After interviewing them, a final framework was established accordingly. It is hoped that the final framework could be a comprehensive one and meet homestays' situation in real practice.

Experts in this research are as follows. Expert A is an academic expert, who has been teaching in a technology university for 8 years and whose expertise is leisure farming, long stay, and marketing and management of recreational business. His papers were all related to recreational industry and he had worked in agricultural public service for 18 years. Currently he is also holding a position as director in two recreational associations. Expert B and C are the ones in practice. They are not only holding positions as director in homestay-related association, but also running homestays. The operating experience of Experts B is approximately 10 years and that of Expert C is about 5 years. In conclusion, the expert group in this research is not only with expertise, but also with rich experiences. In other words, they are representative as expert. In addition, three interviews were conducted for this study and the framework was altered according to the expert group's opinions and suggestions (refer to the Appendix section). A final framework was accordingly formed as shown in Table 2, covering 5 aspects and 30 criteria. The definitions of aspects were elaborated in Table 3.

After finalizing the final framework, three experts were asked to fill up the relevancy among criteria, and a questionnaire was drawn up afterwards. This research considered that if there are too many questions, customers might have no patience or might be not willing to complete the questionnaire, and the result accuracy would thus decrease. In order to prevent this situation, the influential threshold of this research was set to 100 degree; that is, only if three experts all deem a criterion must fully influence another criterion, these two criteria would be considered to be with relevancy. The result is exhibited in Table 4.

## **3. Methodology**

### **3.1. Majority Rule**

Majority rule is a decision rule that selects alternatives which are the major consensus among most experts. Majority rule can be the more-than-half-vote rule or two-thirds rule [40]. On

**Table 1:** Preliminary framework.

Aspect	References	Criteria	References
Surroundings of the building and features	[32, 34]	To utilize natural ventilation sufficiently	[34, 35]
		To utilize plenty natural light	
		To use nontoxic paint	[35]
		To maintain the land's vitality and good condition in the process of design and construction	
		To incorporate the local heritage and landscape elements into design	[31, 35]
Facilities	[31, 34, 39]	The beautification and uniqueness of the interior design	[32, 34, 39]
		Greenization and uniqueness of the garden design	
		Cooking facilities such as kitchen and BBQ	[31, 34, 39]
		Parking space	[31, 32, 34, 39]
		Emergency lighting settings	[31, 39]
Management	[39]	Fire and safety	[31, 32, 34, 39]
		Medical aid	[31, 39]
		Room settings	[12, 31, 39]
		Room tidiness	[32, 34, 39]
		Room coziness	[30, 32, 34]
Service quality	[12, 31, 32, 34]	Hygiene and surrounding cleanliness	[32, 39]
		Local catering planning	[31, 39]
		Introduction of the homestay and resources	
		Overall ambience forming	[39]
		Reception service (such as booking, checking out)	[39]
Privacy and safety	[34]	Information	[31, 32, 34, 39]
		Guiding services	[31, 32, 34]
		Catering	[31, 32, 34, 39]
		Staff quality	[12]
		Service attitude	[30, 32, 34]
Business features	[12]	Food quality	[12, 32, 34]
		Interaction with guests	[12, 32]
		Transportation and pick-up service	[12, 32, 34, 39]
		Local activity arrangement	[31, 39]
		Lodging safety management	
Business features	[12]	Lodging risk management	[12]
		Room themes and features	
		Homestay features	

this basis, this research has set that all must agree have 100 degree (full influence) as the screening threshold when determining criteria relevancy.

**Table 2:** Final framework.

Aspect	References	Criteria	References
Surroundings of the building and features (A)	[32, 34]	To utilize natural ventilation sufficiently (A1) To utilize plenty natural light (A2) To use non-toxic paint (A3) To maintain the land's vitality and good condition in the process of design and construction (A4) To incorporate the local heritage and landscape elements into design (A5) The beautification and uniqueness of the interior design (A6) Greenization and uniqueness of the garden design (A7)	[34, 35] [35] [31, 35] [32, 34, 39]
Service quality (B)	[12, 31, 32, 34]	Service attitude (e.g., reception service, to treat lodgers with voice of the customers) (B1) Pick-up service (offering free pick-up service) (B2) Information service (e.g., local hot spot, tour route planning) (B3) Catering service and quality (e.g., the hosts prepare diversified breakfast in person, freshness of ingredients) (B4)	[30, 32, 34] [12, 32, 34, 39] [31, 32, 34, 39] [12, 31, 32, 34, 39]
Homestay facilities (C)	[31, 34, 39]	Cooking facilities (e.g., kitchen) (C1) Parking space (C2) Safety facilities (e.g., emergency lighting setting, fire prevention settings) (C3) Medical aid (e.g., first-aid box) (C4) Room settings (C5)	[31, 34, 39] [31, 32, 34, 39] [31, 32, 34, 39] [31, 39] [12, 31, 39]
Homestay operation and management (D)	[12, 39]	Room tidiness (D1) Room coziness (D2) Room privacy (D3) Safety (e.g., lodger insurance and room safety) (D4) Room themes and features, for instance, oceanic themes (D5) Homestay features (e.g., aboriginal culture) (D6) Overall ambience forming (D7) Overall tidiness and hygiene (D8)	[32, 34, 39] [30, 32, 34] [34] [12, Expert Interview] [12] [39] [32, 39]
Homestay geist and community co-prosperity (E)	[12, 35], [Expert Interview]	Degree of interaction between hosts and lodgers (E1) Guiding services (E2) Arranging local experiential activities and food (E3) Contribution for living quality of local community (E4) Initiating preserving actions toward local resources (E5) Promoting and preserving local cultural resources (E6)	[12, 32] [31, 32, 34] [30, 31, 39] [12, 34] [35] [Expert Interview]

**Table 3:** The definition of aspect.

Aspect	Definition
Surroundings of the building and features (A)	The entire exterior and interior design of the homestay. Local cultural features are demonstrated and incorporated into the homestay.
Service quality (B)	Customer satisfaction at services provided by homestay proprietors.
Homestay facilities (C)	Hardware of the homestay.
Homestay operation and management (D)	This section is about how the homestay proprietors manage, plan, and design its rooms and surroundings as well as how they guard roomers' lodging privacy and safety.
Homestay geist and community co-prosperity (E)	The homestay proprietors carry out the homestay operation in person and frequently interact with their guests and run the homestay business with a concept of incorporating into its local community in order to help local economic prosper.

### 3.2. ANP (Analytic Network Process)

In the real environment, there are many decision-making problems that cannot just take the pure hierarchical relationship to construct the framework, because the high- and the low-level element may exist the dependence relationship and interaction [41]. As a result, Saaty [41] advanced ANP which has dependence and feedback, and in 2001 [36], he recommended using ANP to solve the problem of interdependent relationships among the criteria or alternatives. In ANP, when nodes correspond to levels or components, that means there exists the network feedback in a system [42]. The elements in the nodes may influence some or all elements in other nodes. In the network, all the nodes can be source nodes, intermediate nodes, or sink nodes. The relationship in the network is represented by the arc, and the direction of arrow means dependence relationship [41]. When the two nodes have the external dependence, it will be represented by the two-way arrow; the nodes in the elements have the internal dependence, it will be represented by the circle arc [43].

ANP has four steps [41, 44] as follows.

*Step 1. Establish the model and the framework.* The question should be clearly described and decomposed a rational system.

*Step 2. Do the pairwise comparison to get the priority vector.* The decision makers are asked to answer the series of pairwise comparison of two random elements or criteria.

*Step 3. Construct the supermatrix.* The concept of super matrix is similar to the Markov chain (Saaty, 1996). To obtain the global priority vector of systematic dependence, we must put the local priority vector into each corresponding line element to get the super matrix.

*Step 4. Choose the best alternative.* The alternative getting the highest weight is the best alternative. In our study, we do not have the alternatives; therefore, we just finish from Step 1 to Step 3.

Table 4: Criteria dependence.

**Table 5:** Linguistic values representations for Likert's scale and fuzzy set.

Linguistic values	Likert's scale	Fuzzy numbers
Highly unsatisfactory	1	(0, 0, 20)
Unsatisfactory	2	(0, 20, 40)
Slightly unsatisfactory	3	(20, 40, 60)
Slightly satisfactory	4	(40, 60, 80)
Satisfactory	5	(60, 80, 100)
Highly satisfactory	6	(80, 100, 100)

### 3.3. Fuzzy Theory

Fuzzy sets theory was introduced by Zadeh [37] in 1965. This theory is about human beings' thoughts, inferences, and perceptions basically are within a certain degree of ambiguity. Therefore, a fuzzy logic is required to describe that something is good or bad or its condition in order to make up the shortcomings of two-value logic (yes-no, true-or-false choices) of traditional sets.

A linguistic variable is a variable with linguistic words or sentences in a natural language [45]. The performance can be treated as a linguistic variable defined in the closed interval [0, 100], whereas a set of the six linguistic values (highly unsatisfactory, unsatisfactory, slightly unsatisfactory, slightly satisfactory, satisfactory, highly satisfactory) is called a term set, denoted by  $T(\text{preference})$ , with respect to the performance. Each linguistic value can be represented by a triangular fuzzy number, which is a fuzzy set in the universe of discourse that is both convex and normal [45, 46].

If the membership function of  $\tilde{A}$  is  $\mu_{\tilde{A}}(X) : R \rightarrow [0, 1]$ , then  $\tilde{A}$  is a fuzzy number which should satisfy the following conditions [47]: (1)  $\mu_{\tilde{A}}(x)$  is convex; (2)  $\mu_{\tilde{A}}(x)$  is normal such that a real number  $x_0$  exists which makes  $\mu_{\tilde{A}}(x_0) = 1$ ; (3)  $\mu_{\tilde{A}}(x)$  is piecewise continuous. This study takes the triangular membership function which is the most common. Each linguistic value can be represented by a triangular fuzzy number  $(l, m, u)$  within the scale ranging from 0 to 100. An example of linguistic values described by triangular fuzzy numbers is demonstrated in Table 5.

In addition, while computing synthetic performance value, fuzzy numbers are often converted into a quantifiable value, which is defuzzification, also known as linguistic-numerical transformation [48]. In Table 5, for example, it is unsatisfactory is demonstrated as "20"  $((0 + 20 + 40)/3)$  and thereafter should be followed by the same analogy. The way to compute the average performance value of the criteria in this research is that assuming there are two interviewees, A and B, they are evaluating a single criterion. If A thinks it is unsatisfactory, the corresponding value is  $(0, 30, 60)$ . If B feels it is satisfactory, the corresponding value is  $(60, 80, 100)$ . As a result, their average performance value of this criterion is  $1/2 \odot ((0, 30, 60) \oplus (60, 80, 100)) = (30, 55, 80)$ , and the defuzzified value is  $1/3 \odot (30 \oplus 55 \oplus 80) = 55$ .

## 4. Empirical Result

### 4.1. Validity and Reliability

The framework of this research is based on literature review and expert interviews; therefore, it is with content validity and expert validity. This research has employed Cronbach's  $\alpha$

**Table 6:** Cronbach's  $\alpha$  value of aspect.

Aspect	Cronbach's $\alpha$
Surroundings of the building and features (A)	0.76
Service quality (B)	0.83
Homestay facilities (C)	0.75
Homestay operation and management (D)	0.90
Homestay geist and community co-prosperity (E)	0.93

coefficient to assess the scale reliability. Cronbach [49] suggested that an  $\alpha$  coefficient less than 0.35 would be deemed as low reliability, the one between 0.35 and 0.7 would be considered as mediocre reliability, and the one above 0.7 is high reliability. The result reveals that Cronbach's  $\alpha$  coefficients of all aspects are above 0.7 as in Table 6, which indicates the scales of this research are with good reliability.

#### 4.2. Scope, Objects, Questionnaire, and Sampling

Regarding researching scope, Hsinchu County was selected, including Zhubei City, Wufong township, Jianshih township, Emei township, Beipu township, Baoshan township, Cyonglin township, Xinfeng township (also known as Hong Mao Kang), Hengshan township, Hukou township, Jhudong township, Xinpu township, and Guansi township (also known as town of longevity). There are two aboriginal towns in Hsinchu, which are Wufong township and Jianshih township. Wufong township is habitat of Atayals and Saisiyats. Atayals and its Ubong Fiesta (worshipping ancestors) and Saisiyats and its Pastaai Fiesta (celebrating harvest) are all important local cultural assets. Jianshih township is also a place of Atayals, where has developed a homestay cluster with aboriginal cultures, such as Smangus. Both places possess abundant natural resources, such as hot and the springs, giant trees, Dabajian Mountain; in addition, the aborigines are giving major efforts to growing high-mountain crops. These places are having great potentials in developing tourism with their leisure farming and cultural tourism. Besides, Emei township, Xinpu township, and Beipu township all possess rich Hakka cultural traditions and heritages, which is also with niches and worthiness to develop tourism. There are bountiful stones in Hengshan township, such as basalts, igneous rocks, and Septarian Boulders, which have attracted numerous stone enthusiasts and have formed another culture; furthermore, the Neiwan branch railway is the artery to maintain its local prosperity. To conclude, Hsinchu has strong aboriginal and Hakka color as well as cultural heritages, which are not only worthwhile to explore, but also with niches in developing in-depth tours. Moreover, this research has classified approved Hsinchu homestays. According to the result, homestays featured with landscape, ecology, and building play the key part; therefore, the evaluation alternative must cover at least two of the forementioned features. With Expert C's strong recommendation, two representative homestays were selected for the evaluation alternative, and their lodging guests were the research objects. Four parts are included in the questionnaire design: (1) pairwise comparison of aspects and criteria, (2) lodging satisfaction, (3) linguistic value for satisfaction, and (4) lodgers' demographics. Concerning people usually lodge in homestays only at weekend or on holidays, this questionnaire was distributed from December 24 2011 to January 18 2012, covering Christmas Eve, Christmas, New Year, and so forth, 50 copies were retrieved in Homestay A, which are one for the homestay host and 49 for the lodgers. 30 copies were

**Table 7:** Demographics distribution.

	Homestay A customer Number	Homestay A customer %	Homestay B customer Number	Homestay B customer %
<b>Gender</b>				
Male	22	44.9	8	28.6
Female	27	55.1	20	71.4
<b>Age</b>				
20 and below	6	12.2	3	10.7
21–40	26	53.1	24	85.7
41–60	17	34.7	1	3.6
61 and above	0	0	0	0
<b>Marriage status</b>				
Married	24	49	10	35.7
Single	24	49	16	57.1
Divorced	1	2	1	3.6
Widower/widow	0	0	1	3.6
<b>Occupation</b>				
Military/public/education service	7	14.3	2	7.1
Agriculture, fishery, and manufacturing	1	2	1	3.6
Service sector	8	16.3	8	28.6
Commercial/financial sector	9	18.4	4	14.3
Electronics manufacturing	7	14.3	6	21.4
Student	8	16.3	3	10.7
Homemaker	0	0	1	3.6
Freelancer	3	6.1	1	3.6
Miscellaneous	6	12.2	2	7.1
<b>Education</b>				
Junior high school and below	0	0	0	0
Senior high school	9	18.4	6	21.4
University/college	28	57.1	17	60.7
Graduate and above	12	24.5	5	17.9
<b>Avg. monthly earning (NT\$)</b>				
20,000 and below	11	22.4	5	17.9
20,001–40,000	15	30.6	7	25
40,001–60,000	13	26.5	13	46.4
60,001 and above	10	20.4	3	10.7

retrieved in Homestay B, two for homestay hosts and 28 for lodgers. 80 copies are all valid samples.

#### 4.3. Demographics Analysis

According to Table 7, regarding gender, female plays the key role among the respondents. About age, the group of 21–40 years old is the most. For marriage status, most are married or unwed at Homestay A, and at Homestay B, most are unwed. With respect to occupation, Homestay A's guests majorly are working in commercial and financial sector and second major group is working in service sector, and the students. The group working in service

**Table 8:** An example of unweighted supermatrix.

	A	B	C	D	E
A	1		1	1	1
B		1	1	1	1
C		1	1	1	
D	1	1	1	1	1
E	1	1		1	1

**Table 9:** An example of weighted supermatrix.

	A	B	C	D	E
A	0.4286		0.25	0.2727	0.25
B		0.375	0.25	0.2727	0.25
C		0.125	0.25	0.0909	
D	0.4286	0.375	0.25	0.2727	0.25
E	0.1429	0.125		0.0909	0.25

sector is the largest and is followed by the one working in electronic manufacturing sector at Homestay B. About education level, most are holding college degree. Concerning average monthly earning, the group earning NT\$ 20,001–40,000 is the major one and is followed by the one earning NT\$ 40,001–60,000 at Homestay A, and most of the Homestay B's lodgers are from the one earning NT\$ 40,001–60,000.

#### 4.4. Analyze the Key Aspects and the Key Criteria

The study got unweighted supermatrix, weighted supermatrix, and limiting supermatrix by Super Decisions Software which is for ANP (analytic network process). Taking one of respondents for example, the three matrices are showed as from Tables 8, 9, and 10.

##### 4.4.1. Aspects

This research has verified that the CI (consistency index) value of each pairwise comparison's matrix is less than 0.1, and the values of limiting supermatrix were obtained by arithmetic average in order to get the rank of aspects as in Tables 11 and 12. According to Table 11, Homestay A's key aspects are homestay operation and management, surroundings of the building and features, and service quality. Key aspects of its customer group are homestay operation and management, service quality, and surroundings of the building and features. Both do not attach great importance to homestay facility and homestay geist and community co-prosperity. Based on Table 12, key aspects of Homestay B are service quality, homestay operation and management, and surroundings of the building and features. Its customer group's key aspects are homestay operation and management, service quality, and surroundings of the building and features. Both Homestay B and its customer group do not pay much attention to homestay facility as well as homestay geist and community co-prosperity.

**Table 10:** An example of limiting supermatrix.

	A	B	C	D	E
A	0.2452	0.2452	0.2452	0.2452	0.2452
B	0.2242	0.2242	0.2242	0.2242	0.2242
C	0.0773	0.0773	0.0773	0.0773	0.0773
D	0.3293	0.3293	0.3293	0.3293	0.3293
E	0.1240	0.1240	0.1240	0.1240	0.1240

**Table 11:** The rank of aspect importance of homestay A (limiting supermatrix value).

Aspect	Proprietor's	Rank	Customers'	Rank
A	0.2452	2	0.2080	3
B	0.2242	3	0.2280	2
C	0.0773	5	0.1585	4
D	0.3293	1	0.2637	1
E	0.1240	4	0.1419	5

#### 4.4.2. Criteria

This research has verified the CI (consistency index) value of each pairwise comparison's matrix is less than 0.1, and the values of limiting supermatrix were obtained by arithmetic average in order to get the rank of criteria as in Tables 13 and 14. According to Table 13, the top 5 key criteria of the Homestay A are to utilize natural ventilation sufficiently, room tidiness, to utilize plenty natural light, safety (e.g., lodger insurance and room safety), and room coziness, and the top 5 key criteria of lodgers are room coziness, to utilize plenty natural light, to utilize natural ventilation sufficiently, safety (e.g., lodger insurance and room safety), service attitude (e.g., reception service and treating lodgers with voice of the customers). In addition, the bottom 5 criteria that Homestay A values are room themes and features (e.g., oceanic themes), promoting and preserving local cultural resources, cooking facilities (e.g., kitchen), contribution for living quality of local community, and medical aid (e.g., first-aid box). The bottom 5 criteria that Homestay A lodgers value are room themes and features (e.g., oceanic themes), the beautification and uniqueness of the interior design, to incorporate the local heritage and landscape elements into design, contribution for living quality of local community, and medical aid (e.g., first-aid box).

According to Table 14, the top 5 key criteria of Homestay B are catering service and quality (e.g., the hosts prepare diversified breakfast in person, freshness of ingredients), service attitude (e.g., reception service, to treat lodgers with voice of the customers), homestay business features (e.g., aboriginal culture), room coziness, and pick-up service (e.g., offering free pick-up service). The ones of its lodgers are safety (e.g., lodger insurance and room safety), service attitude (e.g., reception service, to treat lodgers with voice of the customers), room coziness, pick-up service (e.g., offering free pick-up service), and catering service and quality (e.g., the hosts prepare diversified breakfast in person, freshness of ingredients). Furthermore, the bottom 5 criteria that the Homestay B value are to maintain the land's vitality and good condition in the process of design and construction, the beautification and uniqueness of the interior design, room themes and features (e.g., oceanic themes), medical aid (e.g., first-aid box), and contribution for living quality of local community. The bottom 5 criteria that Homestay B lodgers value are greenization and uniqueness of the

**Table 12:** The rank of aspect importance of Homestay B (limiting supermatrix value).

Aspect	Proprietor's	Rank	Customers'	Rank
A	0.1830	3	0.2132	3
B	0.2610	1	0.2240	2
C	0.1447	5	0.1803	4
D	0.2458	2	0.2552	1
E	0.1654	4	0.1273	5

**Table 13:** The rank of criteria importance of Homestay A (limiting supermatrix value).

Criteria	Proprietor's	Rank	Customers'	Rank
A1	0.0910	1	0.0624	3
A2	0.0741	3	0.0628	2
A3	0.0617	7	0.0487	11
A4	0.0213	16	0.0123	24
A5	0.0180	21	0.0088	28
A6	0.0231	15	0.0112	27
A7	0.0202	18	0.0125	23
B1	0.0304	13	0.0544	5
B2	0.0210	17	0.0537	6
B3	0.0158	23	0.0162	21
B4	0.0424	10	0.0501	10
C1	0.0053	28	0.0119	25
C2	0.0193	19	0.0505	9
C3	0.0140	24	0.0284	16
C4	0.0015	30	0.0035	30
C5	0.0269	14	0.0403	14
D1	0.0856	2	0.0534	7
D2	0.0669	5	0.0688	1
D3	0.0441	9	0.0428	13
D4	0.0708	4	0.0603	4
D5	0.0101	26	0.0113	26
D6	0.0651	6	0.0464	12
D7	0.0379	11	0.0275	17
D8	0.0450	8	0.0512	8
E1	0.0310	12	0.0307	15
E2	0.0170	22	0.0254	18
E3	0.0119	25	0.0173	20
E4	0.0033	29	0.0047	29
E5	0.0192	20	0.0197	19
E6	0.0061	27	0.0131	22

garden design, to maintain the land's vitality and good condition in the process of design and construction, to incorporate the local heritage and landscape elements into design, contribution for living quality of local community, and medical aid (e.g., first-aid box).

In conclusion, both homestays set store by room coziness, but do not attach much weight to the medical aid (e.g., first-aid box), room themes and features (e.g., oceanic themes) as well as contribution for living quality of local community. Customer groups of

**Table 14:** The rank of criteria importance of homestay B (limiting supermatrix value).

Criteria	Proprietor's	Rank	Customers'	Rank
A1	0.0310	15	0.0552	7
A2	0.0282	17	0.0545	8
A3	0.0423	10	0.0483	10
A4	0.0115	26	0.0097	27
A5	0.0134	23	0.0074	28
A6	0.0110	27	0.0102	25
A7	0.0125	24	0.0102	26
B1	0.0717	2	0.0632	2
B2	0.0621	5	0.0602	4
B3	0.0230	18	0.0164	21
B4	0.0720	1	0.0568	5
C1	0.0174	22	0.0145	23
C2	0.0567	7	0.0568	6
C3	0.0308	16	0.0328	15
C4	0.0044	29	0.0028	30
C5	0.0459	9	0.0391	14
D1	0.0348	13	0.0413	13
D2	0.0652	4	0.0614	3
D3	0.0317	14	0.0497	9
D4	0.0595	6	0.0760	1
D5	0.0097	28	0.0147	22
D6	0.0693	3	0.0433	12
D7	0.0176	21	0.0236	18
D8	0.0348	12	0.0434	11
E1	0.0515	8	0.0311	16
E2	0.0415	11	0.0262	17
E3	0.0183	19	0.0165	20
E4	0.0027	30	0.0046	29
E5	0.0181	20	0.0175	19
E6	0.0116	25	0.0125	24

both homestays regard service attitude (e.g., reception service, to treat lodgers with voice of the customers), room coziness, and safety (e.g., lodger insurance and room safety), but do not attach much importance to incorporating the local heritage and landscape elements into design, medical aid (e.g., first-aid box), and contribution for living quality of local community.

#### 4.5. Compare the Weight and the Performance of Criteria

According to Table 15, the top 5 key criteria of Homestay A customer group are room coziness, to utilize plenty natural light, to utilize natural ventilation sufficiently, safety (e.g., lodger insurance and room safety) and service attitude (e.g., reception service, to treat lodgers with voice of the customers). This customer group also deemed Homestay A performed well at most of these criteria, but not the one of service attitude (e.g., reception service, to treat lodgers with voice of the customers), of which the weight rank is 5 and performance is 7.

**Table 15:** The weight and the performance of criteria of Homestay A.

Criteria	Weight	Rank	Performance	Rank
A1	0.0624	3	247.0332	3
A2	0.0628	2	256.7336	2
A3	0.0487	11	186.0355	10
A4	0.0123	24	48.5820	23
A5	0.0088	28	34.2999	28
A6	0.0112	27	43.6726	25
A7	0.0125	23	51.1644	22
B1	0.0544	5	209.3067	7
B2	0.0537	6	160.3981	14
B3	0.0162	21	58.3286	21
B4	0.0501	10	195.9993	8
C1	0.0119	25	42.2324	26
C2	0.0505	9	187.3879	9
C3	0.0284	16	108.3439	17
C4	0.0035	30	12.9081	30
C5	0.0403	14	161.6255	12
D1	0.0534	7	217.9093	5
D2	0.0688	1	279.5624	1
D3	0.0428	13	161.3169	13
D4	0.0603	4	230.2385	4
D5	0.0113	26	39.5616	27
D6	0.0464	12	176.5603	11
D7	0.0275	17	111.5492	15
D8	0.0512	8	210.6188	6
E1	0.0307	15	111.0706	16
E2	0.0254	18	87.4298	18
E3	0.0173	20	58.7417	20
E4	0.0047	29	15.2610	29
E5	0.0197	19	66.3753	19
E6	0.0131	22	43.9704	24

Besides, the room coziness is the more outstanding one in the performance, of which the weight rank is 7 and the performance is 5. The bottom 5 criteria that customer does not attach much importance to are room themes and features (e.g., oceanic themes), beautification and uniqueness of the interior design, to incorporate the local heritage and landscape elements into design, contribution for living quality of local community, and medical aid (e.g., first-aid box). The Homestay A customer group felt Homestay A did not perform well on these criteria.

Based on Table 16, the top 5 key criteria of the Homestay B customer group are safety (e.g., lodger insurance and room safety), service attitude (e.g., reception service, to treat lodgers with voice of the customers), room coziness, pick-up service (e.g., offering free pick-up service), and catering service and quality (e.g., the hosts prepare diversified breakfast in person, freshness of ingredients). The customer group deemed Homestay B performed well in most of these criteria, but not the pick-up service (e.g., offering free pick-up service), whose weight rank is 4 and performance is 7. Besides, the performance in utilizing natural

**Table 16:** The weight and the performance of criteria of Homestay B.

Criteria	Weight	Rank	Performance	Rank
A1	0.0552	7	134.8743	5
A2	0.0545	8	131.8385	6
A3	0.0483	10	112.2302	10
A4	0.0097	27	21.9683	27
A5	0.0074	28	16.0411	28
A6	0.0102	25	24.2880	25
A7	0.0102	26	24.0641	26
B1	0.0632	2	154.7361	3
B2	0.0602	4	129.8464	7
B3	0.0164	21	38.5182	20
B4	0.0568	5	140.3985	4
C1	0.0145	23	32.8570	23
C2	0.0568	6	125.0572	8
C3	0.0328	15	72.5805	16
C4	0.0028	30	6.3272	30
C5	0.0391	14	93.9949	14
D1	0.0413	13	103.7234	12
D2	0.0614	3	156.1340	2
D3	0.0497	9	121.2171	9
D4	0.0760	1	186.3346	1
D5	0.0147	22	33.9942	22
D6	0.0433	12	102.4864	13
D7	0.0236	18	56.9343	18
D8	0.0434	11	107.4946	11
E1	0.0311	16	79.6609	15
E2	0.0262	17	63.3136	17
E3	0.0165	20	38.2953	21
E4	0.0046	29	11.3007	29
E5	0.0175	19	39.2847	19
E6	0.0125	24	29.9663	24

**Table 17:** The linguistic value after defuzzification.

	Highly unsatisfactory	Unsatisfactory	Slightly unsatisfactory	Slightly satisfactory	Satisfactory	Highly satisfactory
A	0	34	54	69	84	100
B		26	45	68	85	

ventilation sufficiently is exceedingly good, whose weight rank is 7 and performance is 5. The bottom 5 criteria that customer group values are greenization and uniqueness of the garden design, to maintain the land's vitality and good condition in the process of design and construction, to incorporate the local heritage and landscape elements into design, contribution for living quality of local community, and medical aid (e.g., first-aid box). The group also deemed that Homestay B did not perform well in these criteria.

To conclude, customer groups of both homestays think the homestay they stayed in performed well in utilizing natural ventilation sufficiently, room coziness, and safety (e.g.,

**Table 18:** The homestay performance.

Homestay A customer			Homestay B customer		
No.	Result	No.	Result	No.	Result
1	SU	26	S	1	S
2	SS	27	S	2	HS
3	SS	28	S	3	SS
4	S	29	S	4	S
5	S	30	HS	5	S
6	HS	31	S	6	S
7	S	32	HS	7	HS
8	SS	33	S	8	HS
9	HS	34	S	9	S
10	SU	35	SU	10	S
11	SS	36	SS	11	S
12	S	37	S	12	SS
13	S	38	SS	13	S
14	S	39	SU	14	S
15	S	40	SS	15	HS
16	S	41	S	16	S
17	HS	42	SS	17	SS
18	SS	43	S	18	S
19	SS	44	S	19	HS
20	SS	45	S	20	HS
21	SS	46	S	21	S
22	S	47	S	22	S
23	SU	48	SS	23	S
24	SS	49	S	24	S
25	S	Overall	S (78)	25	S
				26	S
				27	HS
				28	S
				Overall	S (85)

Note. U: unsatisfactory.

SU: slightly unsatisfactory.

SS: slightly satisfactory.

S: satisfactory.

lodger insurance and room safety), but not well in incorporating the local heritage and landscape elements into design, medical aid (e.g., first-aid box), and contribution for living quality of local community.

#### 4.6. Synthetic Performance of Homestay

The values of highly unsatisfactory, unsatisfactory, slightly unsatisfactory, slightly satisfactory, satisfactory, and highly satisfactory were obtained through defuzzification and arithmetic average as in Table 17. The customer group satisfaction with the lodging homestay was further analyzed, and the result is as Table 18. The computing method is to classify by a concept of “distance”; for example, a Homestay A customer’s overall performance value is

83, which is not far from the option of satisfactory (84); that means it is close up to satisfactory, and it will show as satisfactory. According to Table 18, for Homestay A, 5 customers felt highly satisfactory, 25 customers considered satisfactory, the number of slightly satisfactory is 14 and the number of feeling slightly unsatisfactory is 5. Regarding Homestay B, there were 7 customers feeling highly satisfactory, 18 thought it was a satisfactory experience, and 3 felt slightly satisfactory. Overall, both of them are satisfactory.

## 5. Conclusion and Suggestion

### 5.1. Conclusion

With respect to key aspects, both homestays and their customer groups attach much weight to surroundings of the building and features, service quality, and homestay operation and management, only vary in rank. Moreover, Homestay A and its customer group both value the aspect of homestay operation and management most, and Homestay B takes service quality most seriously and its customer group attaches greatest importance to the homestay operation and management.

With respect to key criteria, both Homestay A and its customer group hold similar ideas, and the difference lays merely on the rank. Homestay A regards utilizing natural ventilation sufficiently as the most important issue, but customer group values room coziness the most. A greater discrepancy lays on service attitude (e.g., reception service, to treat lodgers with voice of the customers), in which the Homestay A's rank is 13, but the one of customer group is 5, and the room tidiness, in which the Homestay A's rank is 2, but the one of customer group is 7. The key criteria of Homestay B and its customer group are mostly the same, which are only different in their rank. Homestay B pays greatest attention to catering service and quality (e.g., the hosts prepare diversified breakfast in person, freshness of ingredients), but its customer group lay greatest stress on safety (e.g., lodger insurance and room safety). Two large discrepancies between Homestay B and its customer group are safety (e.g., lodger insurance and room safety), for which the Homestay B's rank is 6, but its customer group's rank is 1, and homestay features (e.g., aboriginal culture), for which Homestay B's rank is 3, but the one of its customer group is 12.

Regarding importance rank of criteria and homestay performance, the top 5 key criteria of Homestay A's customer group are room coziness, to utilize plenty natural light, to utilize natural ventilation sufficiently, safety (e.g., lodger insurance and room safety), and service attitude (e.g., reception service, to treat lodgers with voice of the customers). Its lodgers all felt Homestay A performed great in most of these criteria, but not in service attitude (e.g., reception service, to treat lodgers with voice of the customers), whose weight rank is 5, but performance is 7. The performance in room tidiness also is not bad, whose weight rank is 7 and performance is 5. The top five key criteria of Homestay B's customer group are safety (e.g., lodger insurance and room safety), service attitude (e.g., reception service, to treat lodgers with voice of the customers), room coziness, pick-up service (offering free pick-up service), and catering service and quality (e.g., the hosts prepare diversified breakfast in person, freshness of ingredients). Its customers deemed Homestay B performed greatly in most of these criteria, but not in pick-up service (offering free pick-up service), whose weight rank is 4 but performance is 7. In addition, the criterion of utilizing natural ventilation sufficiently was doing well, whose weight rank is 7 and performance is 5. To summarize, both homestays performed outstandingly in utilizing natural ventilation sufficiently, room coziness, and safety (e.g., lodger insurance and room safety).

With regard to homestay overall performance, five customers felt highly satisfactory with Homestay A, 25 customers considered satisfactory, 14 were slightly satisfactory, and 5 are slightly unsatisfactory. For Homestay B, 7 customers felt highly satisfactory, 18 thought it was a satisfactory experience, and 3 were slightly satisfactory. To conclude, the overall performance of both homestays met the expectation of their customer groups.

### **5.2. *Suggestion***

Regarding aspects, both homestays and their customer groups attach greater importance to surroundings of the building and features, service quality, as well as homestay operation and management, but not to the homestay facilities and homestay geist and community co-prosperity, which is consistent with the real situation in Taiwan. Homestay operation in Taiwan is working toward a specialized model. Homestays tend to attract people with their magnificent structures and gorgeous settings, among which some even spent hundreds of millions to set up the facilities. Nevertheless, expert group suggested that is working against homestay's natures and geist. Homestay should be a sideline business, and the interaction between hosts and lodgers should be very tight, but not merely about lodging. On this ground, this study suggest it should start from educating consumers in order to make them understand what homestays are and what they should value is not merely about the exterior condition. Moreover, the interaction between homestay hosts and lodgers should be built up, such as taking the lodgers to have an in-depth travel in order to give a great impression to the customers and they would have a wanna-come-again idea. On top of those, government could work with local communities to form "homestay villages," which would be helpful for driving local economic development. Smangus tribe and Tao-Mi Eco Village, for instance, are both very successful examples of building homestay community in Taiwan.

Regarding customer groups' criteria importance rank and their lodging homestay's performance, the top five key criteria of Homestays A's customer group are room coziness, to utilize plenty natural light, to utilize natural ventilation sufficiently, safety (e.g., lodger insurance and room safety), and service attitude (e.g., reception service, to treat lodgers with voice of the customers). The leading four performance rank of Homestay A is consistent with its importance rank, but service attitude (e.g., reception service, to treat lodgers with voice of the customers) is not performing very well, whose weight rank is 5, but performance is 7. This study therefore suggests Homestay A can improve its service skills by on-the-job trainings or special courses. In addition, its pick-up service (offering free pick-up service) also did not perform well, whose weight rank is 6 and performance is 14. This study deems that it is because Homestay A does not offer this service, but its customer group apparently attaches great importance to it. Therefore, it is suggested that Homestay A could buy a shuttle bus or outsource this service to offer this service and improve the satisfaction. The customer group of Homestay B values safety (e.g., lodger insurance and room safety) most, and Homestay B performs best in this criterion as well. It also performs great in other key criteria, such as service attitude (e.g., reception service, to treat lodgers with voice of the customers), whose weight rank is 2 and performance is 3, and room coziness, whose weight rank is 3 and performance is 2, and catering service and quality (e.g., the hosts prepare diversified breakfast in person, freshness of ingredients), whose weigh rank is 5 and performance is 4. But Homestay B did not perform well in pick-up service (offering free pick-up service), whose weight rank is 4 and performance is 7. This study believes the main cause is same as Homestay A, which does not have this service; therefore, the improving measure is as forementioned. To conclude, the overall performance of both homestays is satisfactory;

**Table 19**

No.	Date	Expert opinions	Response and actions
(1)	2011/10/27	<p>Both Expert A and C thought the aspects and criteria in the preliminary framework are appropriate. It should not stress the exterior settings and facilities, but the homestay geist and natures in order to make the framework more distributive and comprehensive. On top of those, it should underline the connection between homestays and local communities. At the end, Expert A indicated that aspects and criteria with similar implications could be merged.</p>	<p>(1) This study has integrated and altered the aspects and criteria with similar implication as shown in Table 2.</p> <p>(2) A new aspect was added and was named "Homestay geist and community co-prosperity", in which there are five criteria: "interaction between hosts and guests", "guiding service", "arranging local experiential activities and food", "contribution for living quality of local community" and "initiating preserving actions toward local resources"</p>
(2)	2011/11/17	<p>Expert B pointed out that hardware of homestays are not on par with the one of hotels. Therefore, they should rely heavily on their natures and geist as well as the interaction with guests, and he also suggested underlining a marketing concept of "homestay village." At the end, the expert considered the framework after integrating and altering is appropriate, but if a new criterion of "promoting and preserving local cultural resources" is added under the aspect of homestay geist and community co-prosperity, it would be more comprehensive.</p>	<p>(1) A new criteria of "promoting and preserving local cultural resources" was added into the aspect of homestay geist and community co-prosperity</p>
(3)	2011/11/24	<p>(1) This research was carried out by emails to ask Expert A about the applicability of the after-integrate framework. He reckoned it is distributive and comprehensive.</p> <p>(2) Expert C thought some remarks should be included in each of the four criteria of the aspect of service quality. In addition, she suggested adding "employee" and "outdoor facilities" as new evaluation criteria.</p>	<p>(1) This research has added remarks to the criteria under the service quality aspect as demonstrated in Table 2.</p> <p>(2) Expert A and this research both deemed that the rooms of homestays are limited to 5 to 15 and homestay should be a sideline business; therefore, having employees is not an essential option. As a result, it would not be included in the framework.</p> <p>(3) Expert A and this research both deemed that homestay is a sideline business and its nature is that only if there are surplus rooms, it would become a homestay. Therefore, having outdoor facilities is not an essential option, which was not going to be included in the framework.</p>

however, if they can improve the criteria that did not have good performance, they would be able to achieve the level of highly satisfactory.

### **5.3. Contribution**

Researches on homestays most were about the customer satisfaction [13–15], marketing strategies [16–18], experiential marketing [19–21], operation management [22–26], or consumer behaviors [27–30], but few constructed the evaluation frameworks from perspectives of the entire homestay business operation. Moreover, the functions, operation models, and natures of homestays are different from hotels [3, 12, 32–34]; therefore, if the evaluation standards for hotels are applied to homestays, it would be incompatible and at variance. On these grounds, this research has been carried out through literature review and interviewing experts in order to develop and construct a set of evaluation indicators applicable to homestay business and to draw up an evaluation framework, which would be more comprehensive and tailor-made for the real practice. The expert group pointed out in the interviewing process that aspects like hardware facilities, service quality, and exterior settings are essential to the current demand, but they all stressed that the evaluation should be brought back to the natures and geist of homestay business, such as the interaction between homestay hosts and lodgers instead of only pursuing the exterior setting relentlessly. Based on this ground, the value of this study and the evaluation frameworks' differences between this study and the earlier ones [12, 31, 34, 35] are to add the criterion of homestay geist and community co-prosperity and to make a further discussion. At the end, it is hoped that the results of this study could be a reference for homestay proprietors to improve their business and for government to promote the homestay business in the future.

### **5.4. Limitation and Future Research**

This research only discussed the homestays in Hsinchu region. It is suggested that the scope of future researches can be expanded. Furthermore, this research focused on the features like landscapes, ecology, and building. Homestays with other features can be explored in the future, such as hot spring homestays.

## **Appendix**

See Table 19.

## **References**

- [1] J. Y. Rong, *Travel Agency: Principles, Practices, and Operations*, Yang-Chih, Taipei City, Taiwan, 2002.
- [2] T. L. Liou and J. W. Chen, "The study of influence of perceptions in tourism developing impact to tourism developing attitudes of residents of dahshi's old streets," *NPUST Humanities and Social Sciences Research*, vol. 3, no. 1, pp. 20–36, 2009.
- [3] S. Y. Lin, M. C. Lin, and G. Y. Liu, "Taiwan host B&B quality accreditation," *Quality Magazine*, vol. 46, no. 12, pp. 32–34, 2010.
- [4] Z. X. Chen and R. X. Chang, *Introduction to Leisure and Recreation Industry*, Yang-Chih, Taipei City, Taiwan, 2008.
- [5] Tourism Bureau M.O.T.C., 2012, <http://admin.taiwan.net.tw/public/public.aspx?no=315>.
- [6] Now News, 2011, <http://www.nownews.com/2011/05/10/91-2711236.htm>.

- [7] Council for Economic Planning and Development, 2011, <http://www.cepd.gov.tw/m1.aspx?sNo=0012621>.
- [8] Tourism Bureau M.O.T.C., 2012, [http://admin.taiwan.net.tw/travel/statistic\\_g.aspx?no=228](http://admin.taiwan.net.tw/travel/statistic_g.aspx?no=228).
- [9] N. Hing, V. McCabe, P. Lewis, and N. Leiper, "Hospitality trends in the asia-pacific: a discussion of five key sectors," *International Journal of Contemporary Hospitality Management*, vol. 10, no. 7, pp. 264–271, 1998.
- [10] Tourism Bureau M.O.T.C., *Regulations for the Management of Home Stay Facilities*, 2001.
- [11] Taiwan Awakening News Networks, 2010, <http://news.sina.com.tw/article/20100714/3492341.html>.
- [12] M. J. Shi, *The study of quality accreditation for Taiwan's bed and breakfast [Unpublished M.S. thesis]*, Asia University, Taichung City, Taiwan, 2008.
- [13] M. K. Chien, K. J. Hsieh, and C. F. Chiu, "An empirical study of promotion strategies, customer's satisfaction, and consumer's behaviors—example of liuchiu bed and breakfast operators," *Journal of Sport, Leisure and Hospitality Research*, vol. 6, no. 3, pp. 113–127, 2011.
- [14] C. H. Hsu and J. W. Chen, "A study on guest experience, expectation, and satisfaction level at resort B&Bs—using makung area as an example," *Journal of Island Tourism Research*, vol. 3, no. 3, pp. 63–87, 2010.
- [15] S. K. Sun, C. L. Liou, and S. T. Chuang, "Analyses on homestay satisfaction: a case study in Baiho," *Review of Agricultural Extension Science*, vol. 24, pp. 1–16, 2008.
- [16] C. J. Hsieh, Y. S. Lin, and F. J. Ling, "A study on the home stay manager's decision-making style of marketing," *Chinese Journal of Agribusiness Management*, vol. 10, pp. 90–120, 2004.
- [17] L. C. Lee, C. Q. Song, and H. C. Kang, "Internet marketing strategies for home stays," *Bio and Leisure Industry Research*, vol. 6, no. 3, pp. 120–140, 2008.
- [18] Y. F. Lin and Y. H. Lin, "Blog marketing for B&B," *Journal of Management Practices and Principles*, vol. 3, no. 3, pp. 52–77, 2009.
- [19] H. C. Hsu and K. M. Cheng, "The shaping of travel image from the perspective of the experiential marketing: hostel managers in taiwan taken as example," *Journal of Sport, Leisure and Hospitality Research*, vol. 5, no. 1, pp. 187–197, 2010.
- [20] S. C. Lin, C. Y. Tsai, and L. W. Chiou, "Improving guests' behavioral intentions based on lodging experiences—using hualien bed & breakfast as an example," *Journal of Tourism and Travel Research*, vol. 2, pp. 73–92, 2007.
- [21] C. C. Shen, P. W. Wang, and C. H. Chen, "A study on the intermediate effect variables of tourist's experience to loyalty—the case of the Fencihu B&B," *Bio and Leisure Industry Research*, vol. 3, no. 2, pp. 85–109, 2005.
- [22] P. C. Chang, Y. T. Hsieh, and Y. H. Zheng, "Operation and management research of B&B by using qualitative study method—take Paiho Town as an example," *Review of Tourism and Hospitality*, vol. 1, no. 1, pp. 71–91, 2008.
- [23] J. M. Lai, C. C. Chang, and S. C. Chang, "A study o key success factors in management of bed-and-breakfast in Nantou," *Journal of Sport and Recreation Research*, vol. 5, no. 3, pp. 15–32, 2011.
- [24] M. Y. Wang and C. Y. Ma, "The case study of the strategy of how to run bed and breakfast business," *Journal of Commercial Modernization*, vol. 4, no. 2, pp. 11–22, 2007.
- [25] R. G. Wang, B. Y. Liao, and M. L Shih, "The correlation between personality traits, leadership and business performance of guesthouse operators—Taitung County as example," *Chinese Journal of Agribusiness Management*, vol. 16, pp. 1–20, 2010.
- [26] C. C. Wu, "The case study of key success factors of homestay—the resource based theory perspective," *Journal of Management Practices and Principles*, vol. 4, no. 4, pp. 142–163, 2010.
- [27] S. F. Chen and T. Y. Chiu, "Consumer behavior of B&B tourists: case in hualien," *Journal of Dahan Institute of Technology*, vol. 21, pp. 155–173, 2006.
- [28] C. H. Hsu and C. W. Shen, "An analysis about current condition of operating household hotel and tourists' consuming characteristicsUW8212;taking taitung county as an example," *Journal of Sport and Recreation Management*, vol. 2, no. 1, pp. 145–158, 2005.
- [29] Y. C. Lin, "A study of home-stay demand differences among various home-stay consumption segments," *Journal of Sport and Recreation Research*, vol. 4, no. 3, pp. 85–105, 2010.
- [30] M. H. Wang, J. L. Chen, and C. J. Ye, "Exploring consumer behavior of home stay in hualien county," *Journal of Outdoor Recreation Study*, vol. 19, no. 4, pp. 1–30, 2006.
- [31] S. J. Ou and H. G. Gian, "Visitors' characteristics and demands of pension in agriculture tourism areas," *Horticulture NCHU*, vol. 22, no. 2, pp. 135–147, 1997.

- [32] Z. Q. Gao, *The study on evaluation indicators for special B&B Inn—the example of seashore type [Unpublished M.S. thesis]*, Kai Nan University, Lujhu Township, Taiwan, 2011.
- [33] S. Y. Lin, "Analysis of product levels of leisure industry—a case study of Taiwan B&B service quality," *Journal of Customer Satisfaction*, vol. 1, no. 1, pp. 145–168, 2005.
- [34] C. H. Yen, C. J. Chang, and L. L. Jian, "A study of evaluation indicators for taiwan home stay facility," *Journal of Farmers' Organizations*, vol. 8, pp. 133–175, 2006.
- [35] Y. J. Chen, *An investigation on evaluation indicators for Eco-inn selecting in Taiwan [Unpublished M.S. thesis]*, National Taipei College of Nursing, Taipei City, Taiwan, 2008.
- [36] T. L. Saaty, *Decision Making with Dependence and Feedback: The Analytic Network Process*, RWS Publications, Pittsburgh, Pa, USA, 2nd edition, 2001.
- [37] L. A. Zadeh, "Fuzzy Sets," *Information and Computation*, vol. 8, pp. 338–353, 1965.
- [38] M. C. Su and X. D Chang, *Machine Learning: Neural Networks, Fuzzy Systems, and Genetic Algorithms*, Quanhua, Taipei City, Taiwan, 3rd edition, 2004.
- [39] General Chamber of Commerce of the R.O.C., 2006, <http://60.244.127.66/cgi-bin/big5/p1215/ko04>.
- [40] J. Y. Teng, *Project Evaluation: Methods and Applications*, The Center of Planning and Management in National Taiwan Ocean University, Keelung City, Taiwan, 2nd edition, 2005.
- [41] T. L. Saaty, *Decision Making with Dependence and Feedback: The Analytic Network Process*, Pittsburgh, Pa, USA, RWS Publications, 1996.
- [42] T. L. Saaty, *The Analytic Hierarchy Process*, McGraw-Hill, New York, NY, USA, 1980.
- [43] J. Sarkis, "Quantitative models for performance measurement systems—alternate considerations," *International Journal of Production Economics*, vol. 86, no. 1, pp. 81–90, 2003.
- [44] L. M. Meade and J. Sarkis, "Analyzing organizational project alternatives for agile manufacturing processes: an analytical network approach," *International Journal of Production Research*, vol. 37, no. 2, pp. 241–261, 1999.
- [45] H. J. Zimmermann, *Fuzzy Set Theory and Its Applications*, Kluwer, Boston, Mass, USA, 1996.
- [46] W. Pedrycz and F. Gomide, *An Introduction to Fuzzy Sets: Analysis and Design*, Complex Adaptive Systems, MIT Press, Cambridge, Mass, USA, 1998.
- [47] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Application*, vol. 144 of *Mathematics in Science and Engineering*, Academic Press, New York, NY, USA, 1980.
- [48] M. Delgado, F. Herrera, E. Herrera-Viedma, and L. Martínez, "Combining numerical and linguistic information in group decision making," *Information Sciences*, vol. 107, no. 1–4, pp. 177–194, 1998.
- [49] L. J. Cronbach, "Coefficient alpha and the internal structure of tests," *Psychometrika*, vol. 16, no. 3, pp. 297–334, 1951.

*Research Article*

# Mixed Mortar Element Method for $P_1^{NC}/P_0$ Element and Its Multigrid Method for the Incompressible Stokes Problem

**Yaqin Jiang<sup>1,2</sup> and Jinru Chen<sup>1</sup>**

<sup>1</sup> Jiangsu Key Laboratory for NSLSCS, School of Mathematical Sciences, Nanjing Normal University, Nanjing 210046, China

<sup>2</sup> School of Sciences, Nanjing University of Posts and Telecommunications, Nanjing 210046, China

Correspondence should be addressed to Yaqin Jiang, yqjiangnj@163.com

Received 15 November 2011; Revised 19 April 2012; Accepted 20 April 2012

Academic Editor: Dongdong Ge

Copyright © 2012 Y. Jiang and J. Chen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We discuss a mortar-type  $P_1^{NC}/P_0$  element method for the incompressible Stokes problem. We prove the inf-sup condition and obtain the optimal error estimate. Meanwhile, we propose a  $\mathcal{W}$ -cycle multigrid for solving this discrete problem and prove the optimal convergence of the multigrid method, that is, the convergence rate is independent of the mesh size and mesh level. Finally, numerical experiments are presented to confirm our theoretical results.

## 1. Introduction

As we all know, the application of viscous incompressible flows is of considerable interest. For example, the design of hydraulic turbines, or rheologically complex flows appears in many processes which are involved in plastics and molten metals. Therefore, in recent decades, many engineers and mathematicians have concentrated their efforts on the Stokes problem, especially the problem that can be handled by the finite element methods. In [1], Girault and Raviart provided a fairly comprehensive treatment of the most recent development in the finite-element method. Some new divergence-free elements were proposed to solve Stokes problem recently (see [2, 3] and others). Due to this development in the finite-element theory, many numerical algorithms were established to solve the Stokes equations. Among these algorithms, multigrid methods and domain decomposition methods for the Stokes equations are very prevalent. In [4], the authors constructed an efficient smoother. Based on the smoother, the multigrid methods have been greatly developed (see [5, 6]). Meanwhile,

a FETI-DP method was extended to the incompressible Stokes equations in [7, 8], a BDDC algorithm for this problem was developed too in [9] and others.

In the last twenty years, mortar element methods have attracted much attention and it was first introduced in [10]. This method is a nonconforming domain decomposition method with nonoverlapping subdomains. In mortar finite-element methods, the meshes on adjacent subdomains may not match with each other across the interfaces of the subdomains. The coupling of the finite-element functions on adjacent meshes is done by enforcing the so-called mortar condition across the interfaces (see [10] for details). There have been considerable researches on the mortar element methods (see [11–13] and others).

In [12], the author discussed the mortar-type conforming element ( $P_2/P_1$  element) method for the Stokes problem, and then Chen and Huang proposed the mortar-type nonconforming element ( $Q_1^{\text{rot}}/Q_0$  element) method for the problem in [5]. It is well known that the rotated  $Q_1$  element is a rectangle element, and it is not a flexible finite element since it is only suitable for the rectangular or L-shape-bounded domain. Moreover, the rotated  $Q_1$  element is a quadratic element and is not as convenient as the linear elements in calculating.

In this paper, we apply the mortar element method coupling with  $P_1$  nonconforming finite element to the incompressible Stokes problem. The  $P_1$  nonconforming finite element is a triangular element and it is suitable for more extensive polygonal domain than the rotated  $Q_1$  element. Moreover, owing to its linearity, the computational work is less than the rotated  $Q_1$  element. We prove the so-called inf-sup condition and obtain the optimal error estimate. When solving the discrete problem, we also present a  $\mathcal{W}$ -cycle multigrid algorithm, but the analysis about the convergence of the multigrid is different from [5]. We only prove that the prolong operator satisfies the criterion which proposed in [14] and we obtain the optional convergence with simpler analysis than that in [5]. Meanwhile, we do some numerical experiments which were realized in [5]. From numerical results, we note that the number of iterations is less than the rotated  $Q_1$  element method when achieving the same relative error.

The rest of this paper is organized as follows. In Section 2, we review the Stokes problem and introduce the mortar element method for  $P_1$  nonconforming element. Section 3 gives verification of the inf-sup condition and error estimate. The multigrid algorithm and the convergence analysis are given in Sections 4 and 5, respectively. The last section presents some numerical experiments. Throughout this paper, we denote by “C” a universal constant which is independent from the mesh size and level, whose values can differ from place to place.

## 2. Preliminaries

We only consider the incompressible flow problem, the steady-state Stokes problem, so that we can compare the results with those in [5].

The partial differential equations of the model problem is

$$\begin{aligned} -\Delta \mathbf{u} + \nabla p &= f \quad \text{in } \Omega, \\ \operatorname{div} \mathbf{u} &= 0 \quad \text{in } \Omega, \\ \mathbf{u} &= \mathbf{0} \quad \text{on } \partial\Omega, \end{aligned} \tag{2.1}$$

where  $\Omega$  is bounded convex polygonal domain in  $R^2$ ,  $\mathbf{u}$  represents the velocity of fluid,  $p$  is pressure, and  $\mathbf{f}$  is external force. Define

$$L_0^2(\Omega) = \left\{ q \in L^2(\Omega) \mid \int_{\Omega} q \, dx = 0 \right\}. \quad (2.2)$$

The mixed variational formulation of problem (2.1) is to find  $(\mathbf{u}, p) \in (H_0^1(\Omega))^2 \times L_0^2(\Omega)$  such that

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) &= \langle \mathbf{f}, \mathbf{v} \rangle, \quad \forall \mathbf{v} \in (H_0^1(\Omega))^2, \\ b(\mathbf{u}, q) &= 0, \quad \forall q \in L_0^2(\Omega), \end{aligned} \quad (2.3)$$

where the bilinear formulations  $a(\cdot, \cdot)$  on  $(H_0^1(\Omega))^2 \times (H_0^1(\Omega))^2$ ,  $b(\cdot, \cdot)$  on  $(H_0^1(\Omega))^2 \times L_0^2(\Omega)$  and the dual parity  $\langle \cdot, \cdot \rangle$  on  $(L^2(\Omega))^2 \times (L^2(\Omega))^2$  are given, respectively, by

$$a(\mathbf{u}, \mathbf{v}) = \int_{\Omega} \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, dx, \quad b(\mathbf{v}, q) = - \int_{\Omega} \operatorname{div} \mathbf{v} q \, dx, \quad \langle \mathbf{f}, \mathbf{v} \rangle = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, dx. \quad (2.4)$$

It is well known that the bilinear form  $b(\cdot, \cdot)$  satisfies the inf-sup condition, that is, there exists a positive constant  $\beta$  for any  $q \in L_0^2(\Omega)$  such that

$$\sup_{\mathbf{v} \in (H_0^1(\Omega))^2} \frac{b(\mathbf{v}, q)}{\|\mathbf{v}\|_{(H^1(\Omega))^2}} \geq \beta \|q\|_{L^2(\Omega)}. \quad (2.5)$$

According to the assumption on  $\Omega$  and the saddle point theory in [15], we know that if  $\mathbf{f} \in (L^2(\Omega))^2$ , then there exists a unique solution  $(\mathbf{u}, p) \in (H_0^1(\Omega) \cap H^2(\Omega))^2 \times (L_0^2(\Omega) \cap H^1(\Omega))$  satisfying

$$\|\mathbf{u}\|_{(H^2(\Omega))^2} + \|p\|_{H^1(\Omega)} \leq C \|\mathbf{f}\|_{(L^2(\Omega))^2}. \quad (2.6)$$

We now introduce a mortar finite-element method for solving problem (2.3). First, we partition  $\Omega$  into nonoverlapping polygonal subdomains such that

$$\overline{\Omega} = \bigcup_{i=1}^N \overline{\Omega}_i, \quad \Omega_i \cap \Omega_j = \emptyset \quad \text{if } i \neq j. \quad (2.7)$$

They are arranged, so that the intersection of  $\Omega_i \cap \Omega_j$  for  $i \neq j$  is an empty set or an edge, or a vertex; that is, the partition is geometrically conforming. Denote by  $\gamma_m$  the common open edge to  $\Omega_i$  and  $\Omega_j$ , then the interface  $\Gamma = \bigcup_{i=1}^N \partial \Omega_i \setminus \partial \Omega$  is broken into a set of disjoint open straight segments  $\gamma_m$  ( $1 \leq m \leq M$ ), that is,

$$\Gamma = \bigcup_{m=1}^M \bar{\gamma}_m, \quad \gamma_m \cap \gamma_n = \emptyset \quad \text{if } m \neq n. \quad (2.8)$$

By  $\gamma_{m(i)}$  we denote an edge of  $\Omega_i$  called mortar and by  $\delta_{m(j)}$  an edge of  $\Omega_j$  that geometrically occupies the same place called nonmortar.

With each  $\Omega_i$ , we associate a quasuniform triangulation  $\mathcal{T}_h(\Omega_i)$  made of elements that are triangles. The mesh size  $h_i$  is the diameter of largest element in  $\mathcal{T}_h(\Omega_i)$ . We define  $h = \max_{1 \leq i \leq N} h_i$ ,  $\mathcal{T}_h = \bigcup_{i=1}^N \mathcal{T}_h(\Omega_i)$ . Let CR nodal points be the nonconforming nodal points, that is, the midpoints of the edges of the elements in  $\mathcal{T}_h(\Omega_i)$ . Denote the set of CR nodal points belonging to  $\bar{\Omega}_i$ ,  $\partial\Omega_i$  and  $\partial\Omega$  by  $\Omega_{ih}^{\text{CR}}$ ,  $\partial\Omega_{ih}^{\text{CR}}$  and  $\partial\Omega_h^{\text{CR}}$ , respectively.

For each triangulation  $\mathcal{T}_h(\Omega_i)$  on  $\Omega_i$ , the  $P_1$  nonconforming element velocity space and piecewise constant pressure space are defined, respectively, as follows:

$$\begin{aligned} X_h(\Omega_i) &= \left\{ \mathbf{v}_i \in \left(L^2(\Omega_i)\right)^2 \mid \mathbf{v}_i|_\tau \text{ is linear } \forall \tau \in \mathcal{T}_h(\Omega_i), \right. \\ &\quad \left. \mathbf{v}_i \text{ is continuous at midpoint of } \tau, \mathbf{v}(m_i) = \mathbf{0} \forall m_i \in \partial\Omega_h^{\text{CR}} \right\}, \quad (2.9) \\ Q_h(\Omega_i) &= \left\{ q_i \in L^2(\Omega_i) \mid q_i|_\tau \text{ is a constant for } \tau \in \mathcal{T}_h(\Omega_i) \right\}. \end{aligned}$$

Then the product space  $\tilde{X}_h(\Omega) = \prod_{i=1}^N X_h(\Omega_i)$  is a global  $P_1$  nonconforming element space for  $\mathcal{T}_h$  on  $\Omega$ .

For any interface  $\gamma_m = \gamma_{m(i)} = \delta_{m(j)}$  ( $1 \leq m \leq M$ ), there are two different and independent triangulations  $\mathcal{T}_h(\gamma_{m(i)})$  and  $\mathcal{T}_h(\delta_{m(j)})$ , which produce two sets of CR nodes belonging to  $\gamma_m$ : the midpoints of the elements belonging to  $\mathcal{T}_h(\gamma_{m(i)})$  and  $\mathcal{T}_h(\delta_{m(j)})$  denoted by  $\gamma_{m(i)}^{\text{CR}}$  and  $\delta_{m(j)}^{\text{CR}}$ , respectively.

In order to introduce the mortar condition across the interfaces  $\gamma_m$ , we need the auxiliary test space  $S_h(\delta_{m(j)})$  which is defined by

$$\begin{aligned} S_h(\delta_{m(j)}) &= \left\{ \mathbf{v} \in \left(L^2(\delta_{m(j)})\right)^2 \mid \mathbf{v} \text{ is piecewise constant on elements of triangulation } \mathcal{T}_h(\delta_{m(j)}) \right\}. \quad (2.10) \end{aligned}$$

For each nonmortar edge  $\delta_{m(j)}$ , define the  $L^2$ -projection operator:  $Q_{h,\delta_{m(j)}} : (L^2(\gamma_m))^2 \rightarrow S_h(\delta_{m(j)})$  by

$$(Q_{h,\delta_{m(j)}} \mathbf{v}, \mathbf{w})_{L^2(\delta_{m(j)})} = (\mathbf{v}, \mathbf{w})_{L^2(\delta_{m(j)})}, \quad \forall \mathbf{w} \in S_h(\delta_{m(j)}). \quad (2.11)$$

Now we can define the mortar-type  $P_1$  nonconforming element space as follows:

$$\begin{aligned} X_h(\Omega) &= \left\{ \mathbf{v} \in \tilde{X}_h(\Omega) \mid \mathbf{v}|_{\Omega_i} \in X_h(\Omega_i), Q_{h,\delta_{m(j)}}(\mathbf{v}|_{\delta_{m(j)}}) = Q_{h,\delta_{m(j)}}(\mathbf{v}|_{\gamma_{m(i)}}), \right. \\ &\quad \left. \forall \gamma_m = \gamma_{m(i)} = \delta_{m(j)} \subset \Gamma \right\}, \quad (2.12) \end{aligned}$$

the condition of the equality in (2.12) which the velocity function  $\mathbf{v}$  satisfies is called mortar condition.

The global  $P_0$  element pressure space on  $\Omega$  is defined by

$$Q_h(\Omega) = \left\{ q \in L_0^2(\Omega) \mid q|_{\Omega_i} \in Q_h(\Omega_i) \right\}. \quad (2.13)$$

We now establish the discrete system for problem (2.3) based on the mixed finite-element spaces  $X_h(\Omega) \times Q_h(\Omega)$ .

We first define the following formulations:

$$\begin{aligned} a_{h_i}(\mathbf{u}_h^i, \mathbf{v}_h^i) &= \sum_{\tau \in \mathcal{T}_h(\Omega_i)} \int_{\tau} \nabla \mathbf{u}_h^i \cdot \nabla \mathbf{v}_h^i \, dx, \quad \forall \mathbf{u}_h^i, \mathbf{v}_h^i \in X_h(\Omega_i), \\ b_{h_i}(\mathbf{v}_h^i, p_h^i) &= - \sum_{\tau \in \mathcal{T}_h(\Omega_i)} \int_{\tau} \operatorname{div} \mathbf{v}_h^i \cdot p_h^i \, dx, \quad \forall \mathbf{v}_h^i \in X_h(\Omega_i), \forall p_h^i \in Q_h(\Omega_i). \end{aligned} \quad (2.14)$$

Let

$$a_h(\mathbf{u}_h, \mathbf{v}_h) = \sum_{i=1}^N a_{h_i}(\mathbf{u}_h, \mathbf{v}_h), \quad b_h(\mathbf{v}_h, p_h) = \sum_{i=1}^N b_{h_i}(\mathbf{v}_h, p_h). \quad (2.15)$$

Then the discrete approximation of problem (2.3) is to find  $(\mathbf{u}_h, p_h) \in X_h(\Omega) \times Q_h(\Omega)$  such that

$$\begin{aligned} a_h(\mathbf{u}_h, \mathbf{v}_h) + b_h(\mathbf{v}_h, p_h) &= \langle \mathbf{f}, \mathbf{v}_h \rangle, \quad \forall \mathbf{v}_h \in X_h(\Omega), \\ b_h(\mathbf{u}_h, q_h) &= 0, \quad \forall q_h \in Q_h(\Omega). \end{aligned} \quad (2.16)$$

In the next section, we prove that the discrete problem (2.16) has a unique solution and we obtain error estimate.

### 3. Existence, Uniqueness, and Error Estimate of the Discrete Solution

According to the Brezzi theory, the well-posedness of problem (2.16) depends closely on the characteristics of both bilinear forms  $a_h(\cdot, \cdot)$  and  $b_h(\cdot, \cdot)$ . We equip the space  $X_h(\Omega)$  with the following norm:

$$\|\mathbf{v}\|_h^2 := \sum_{i=1}^N \|\mathbf{v}\|_{h,i}^2, \quad \|\mathbf{v}\|_{h,i}^2 := a_{h_i}(\mathbf{v}, \mathbf{v}). \quad (3.1)$$

We can find in [1] that the local space family  $\{X_h^0(\Omega_i), Q_h^0(\Omega_i)\}$  is div-stable; that is, there exists a constant  $\tilde{\beta}$  independent of  $h_i$  such that

$$\sup_{\tilde{\mathbf{v}}_h \in X_h^0(\Omega_i)} \frac{b_h(\tilde{\mathbf{v}}_h, \tilde{q}_h)}{\|\tilde{\mathbf{v}}_h\|_{h,i}} \geq \tilde{\beta} \|\tilde{q}_h\|_{L^2(\Omega_i)}, \quad \forall \tilde{q}_h \in Q_h^0(\Omega_i), \quad (3.2)$$

where  $X_h^0(\Omega_i) = \{\mathbf{v} \in X_h(\Omega_i) \mid \mathbf{v}(m_i) = \mathbf{0}, \forall m_i \in \partial\Omega_{i,h}^{\text{CR}}\}$ ,  $Q_h^0(\Omega_i) = Q_h(\Omega_i) \cap L_0^2(\Omega_i)$ .

In order to prove that the global space family  $X_h(\Omega) \times Q_h(\Omega)$  is div-stable, it is necessary to define the global spaces as

$$\check{Q}_h(\Omega) = \left\{ \check{q} = \prod_{i=1}^N \check{q}_i \in R^N, (\check{q}, 1) = \sum_{i=1}^N \check{q}_i |\Omega_i| = 0 \right\}. \quad (3.3)$$

We first prove that the family  $\{X_h(\Omega), \check{Q}_h(\Omega)\}$  is div-stable.

**Lemma 3.1.** *The following inf-sup condition holds:*

$$\sup_{\mathbf{v}_h \in X_h(\Omega)} \frac{b_h(\mathbf{v}_h, \check{q})}{\|\mathbf{v}_h\|_h} \geq \check{\beta} \|\check{q}\|_{L^2(\Omega)} \quad \forall \check{q} \in \check{Q}_h(\Omega), \quad (3.4)$$

where the constant  $\check{\beta}$  does not depend on  $h$ .

*Proof.* We decompose the space  $(H_0^1(\Omega))^2$  by  $(H_0^1(\Omega))^2 = \prod_{i=1}^N V(\Omega_i)(V(\Omega_i)) = (H_0^1(\Omega))^2|_{\Omega_i}$  and define a local interpolation operator  $\pi_i: V(\Omega_i) \rightarrow X_h(\Omega_i)$  as

$$\pi_i \mathbf{v}(m_i) = \frac{1}{|e_i|} \int_{e_i} \mathbf{v} ds, \quad (3.5)$$

where  $e_i$  is an edge of  $\tau \in \mathcal{T}_h(\Omega_i)$ ,  $m_i$  is the midpoint of  $e_i$ . Then we can define a global interpolation operator  $\pi: (H_0^1(\Omega))^2 \rightarrow \tilde{X}_h(\Omega)$  as follows:

$$\pi \mathbf{v} = (\pi_1 \mathbf{v}_1, \pi_2 \mathbf{v}_2, \dots, \pi_N \mathbf{v}_N), \quad \mathbf{v}_i = \mathbf{v}|_{\Omega_i}, \quad \forall \mathbf{v} \in (H_0^1(\Omega))^2. \quad (3.6)$$

Define the operator  $\Xi_{h,\delta_{m(j)}}: \tilde{X}_h(\Omega) \rightarrow \tilde{X}_h(\Omega)$  by

$$(\Xi_{h,\delta_{m(j)}} \mathbf{v})(m_i) = \begin{cases} Q_{h,\delta_{m(j)}}(\mathbf{v}|_{Y_{m(i)}} - \mathbf{v}|_{\delta_{m(j)}})(m_i), & m_i \in \delta_{m(j)}^{\text{CR}}, \\ 0, & \text{otherwise.} \end{cases} \quad (3.7)$$

We can deduce that for any  $\mathbf{v} \in (H_0^1(\Omega))^2$ , there exists a  $\mathbf{v}_h^* \in X_h(\Omega)$  satisfying

$$b(\mathbf{v} - \mathbf{v}_h^*, \check{q}) = 0. \quad (3.8)$$

In fact, we can set  $\mathbf{v}_h^* = \pi\mathbf{v} + \sum_{m=1}^M \Xi_{h,\delta_{m(j)}}(\pi\mathbf{v})$ . Obviously  $\mathbf{v}_h^* \in X_h(\Omega)$  and

$$\begin{aligned} b(\mathbf{v} - \mathbf{v}_h^*, \check{q}) &= -\sum_{i=1}^N \sum_{\tau \in \mathcal{T}_h(\Omega_i)} \int_{\tau} \operatorname{div}(\mathbf{v} - \mathbf{v}_h^*) \check{q} \, dx = -\sum_{i=1}^N \sum_{\tau \in \mathcal{T}_h(\Omega_i)} \int_{\partial\tau} (\mathbf{v} - \mathbf{v}_h^*) \cdot \mathbf{n} \check{q} \, ds \\ &= -\sum_{\tau \in \mathcal{T}_h} \int_{\partial\tau} (\mathbf{v} - \pi\mathbf{v}) \cdot \mathbf{n} \check{q} \, ds + \sum_{\tau \in \mathcal{T}_h} \int_{\partial\tau} \sum_{m=1}^M \Xi_{h,\delta_{m(j)}}(\pi\mathbf{v}) \cdot \mathbf{n} \check{q} \, ds \\ &= \sum_{j=1}^M \int_{\delta_{m(j)}} Q_{h,\delta_{m(j)}} \left( (\pi\mathbf{v})|_{\gamma_{m(i)}} - (\pi\mathbf{v})|_{\delta_{m(j)}} \right) \cdot \mathbf{n} \check{q}_j \, ds \\ &= \sum_{j=1}^M \int_{\delta_{m(j)}} \left( (\pi\mathbf{v})|_{\gamma_{m(i)}} - (\pi\mathbf{v})|_{\delta_{m(j)}} \right) \cdot \mathbf{n} \check{q}_j \, ds \\ &= \sum_{j=1}^M \int_{\delta_{m(j)}} \left( \mathbf{v}|_{\gamma_{m(i)}} - \mathbf{v}|_{\delta_{m(j)}} \right) \cdot \mathbf{n} \check{q}_j \, ds \\ &= 0. \end{aligned} \quad (3.9)$$

On the other hand

$$\|\mathbf{v}^*\|_h \leq \|\pi\mathbf{v}\|_h + \left\| \Xi_{h,\delta_{m(j)}} \pi\mathbf{v} \right\|_h' \quad (3.10)$$

by norm equivalence we have

$$\begin{aligned} \|\pi\mathbf{v}\|_h^2 &= \sum_{\tau} |\pi\mathbf{v}|_{H^1(\tau)}^2 \leq C \sum_{\tau} (\pi\mathbf{v}(m_i) - \pi\mathbf{v}(m_j))^2 \\ &= C \sum_{\tau} \left( \frac{1}{|e_i|} \int_{e_i} \mathbf{v} \, ds - \frac{1}{|e_j|} \int_{e_j} \mathbf{v} \, ds \right)^2 \\ &= C \sum_{\tau} \left( \frac{1}{|e_i|} \int_{e_i} (\mathbf{v} - \bar{\mathbf{v}}) \, ds - \frac{1}{|e_j|} \int_{e_j} (\mathbf{v} - \bar{\mathbf{v}}) \, ds \right)^2 \\ &\leq C \sum_{\tau} \left( \frac{1}{|e_i|^2} \left( \int_{e_i} (\mathbf{v} - \bar{\mathbf{v}}) \, ds \right)^2 + \frac{1}{|e_j|^2} \left( \int_{e_j} (\mathbf{v} - \bar{\mathbf{v}}) \, ds \right)^2 \right), \end{aligned} \quad (3.11)$$

where  $m_i, m_j$  are the midpoints of the edges of  $\tau$ , and  $\bar{\mathbf{v}}$  is the integral average of  $\mathbf{v}$  in  $\tau$ , by Hölder inequality, trace theorem, and Friedrichs' inequality we can get

$$\begin{aligned} \frac{1}{|e_i|^2} \left( \int_{e_i} (\mathbf{v} - \bar{\mathbf{v}}) ds \right)^2 &\leq \frac{1}{|e_i|} \int_{e_i} (\mathbf{v} - \bar{\mathbf{v}})^2 ds \leq Ch^{-1} \int_{\partial\tau} (\mathbf{v} - \bar{\mathbf{v}})^2 ds \\ &\leq C \left( h^{-2} \int_{\tau} (\mathbf{v} - \bar{\mathbf{v}})^2 dx + |\mathbf{v} - \bar{\mathbf{v}}|_{H^1(\tau)}^2 \right) \\ &\leq C |\mathbf{v}|_{H^1(\tau)}^2, \end{aligned} \quad (3.12)$$

and combining (3.11), we obtain

$$\|\pi\mathbf{v}\|_h \leq C \|\mathbf{v}\|_h. \quad (3.13)$$

Using norm equivalence we derive

$$\begin{aligned} \left\| \Xi_{h,\delta_{m(j)}} \pi\mathbf{v} \right\|_h^2 &\leq C \sum_{m_i \in \delta_{m(j)}^{\text{CR}}} \left( \Xi_{h,\delta_{m(j)}} \pi\mathbf{v}(m_i) \right)^2 \\ &= C \sum_{m_i \in \delta_{m(j)}^{\text{CR}}} \left( Q_{h,\delta_{m(j)}} \left( (\pi\mathbf{v})|_{\gamma_{m(i)}} - (\pi\mathbf{v})|_{\delta_{m(j)}} \right)(m_i) \right)^2 \\ &\leq Ch^{-1} \left\| Q_{h,\delta_{m(j)}} \left( (\pi\mathbf{v})|_{\gamma_{m(i)}} - (\pi\mathbf{v})|_{\delta_{m(j)}} \right)(m_i) \right\|_{0,\gamma_m}^2 \\ &\leq Ch^{-1} \left\| (\pi\mathbf{v})|_{\gamma_{m(i)}} - (\pi\mathbf{v})|_{\delta_{m(j)}} \right\|_{0,\gamma_m}^2 \\ &\leq Ch^{-1} \left( \left\| (\pi\mathbf{v})|_{\gamma_{m(i)}} - \mathbf{v}|_{\delta_{m(j)}} \right\|_{0,\gamma_m}^2 + \left\| \mathbf{v}|_{\delta_{m(j)}} - (\pi\mathbf{v})|_{\delta_{m(j)}} \right\|_{0,\gamma_m}^2 \right) \\ &:= Ch^{-1} (K_1 + K_2). \end{aligned} \quad (3.14)$$

From trace theorem and (3.13), it follows that

$$K_2 \leq Ch \|\mathbf{v}\|_{h,j}^2. \quad (3.15)$$

So we only need to estimate  $K_1$ . Owing to  $\mathbf{v} \in (H_0^1(\Omega))^2$ , we then obtain

$$\left\| (\pi\mathbf{v})|_{\gamma_{m(i)}} - \mathbf{v}|_{\delta_{m(j)}} \right\|_{0,\gamma_m}^2 = \left\| (\pi\mathbf{v})|_{\gamma_{m(i)}} - \mathbf{v}|_{\gamma_{m(i)}} \right\|_{0,\gamma_m}^2 \leq Ch \|\mathbf{v}\|_{h,i}^2. \quad (3.16)$$

The bounds in (3.15) and (3.16) lead to

$$\left\| \Xi_{h,\delta_{m(j)}} \pi\mathbf{v} \right\|_h^2 \leq C \left( \|\mathbf{v}\|_{h,i}^2 + \|\mathbf{v}\|_{h,j}^2 \right), \quad (3.17)$$

which together with (3.13) and (3.17) give

$$\|\mathbf{v}^*\|_h \leq C\|\mathbf{v}\|_{(H^1(\Omega))^2}. \quad (3.18)$$

Since  $\{(H_0^1(\Omega))^2, L_0^2(\Omega)\}$  is div-stable, following (3.8) and (3.18), by Fortin rules, we have completed the proof of Lemma 3.1  $\square$

Now we recall the following Brezzi theory about the existence, uniqueness, and error estimate for the discrete solution.

**Theorem 3.2.** *The bilinear forms  $a_h(\cdot, \cdot)$  and  $b_h(\cdot, \cdot)$  have the following properties:*

- (i)  $a_h(\cdot, \cdot)$  is continuous and uniformly elliptic on the mortar-type  $P_1$  nonconforming space  $X_h(\Omega)$ , that is,

$$\begin{aligned} a_h(\mathbf{u}_h, \mathbf{v}_h) &\leq \|\mathbf{u}_h\|_h \|\mathbf{v}_h\|_h, \quad \forall \mathbf{u}_h, \mathbf{v}_h \in X_h(\Omega), \\ a_h(\mathbf{v}_h, \mathbf{v}_h) &\geq C\|\mathbf{v}_h\|_h^2, \quad \forall \mathbf{v}_h \in X_h(\Omega); \end{aligned} \quad (3.19)$$

- (ii)  $b_h(\cdot, \cdot)$  is also continuous on the space family  $X_h(\Omega) \times Q_h(\Omega)$ , that is,

$$b_h(\mathbf{v}_h, q) \leq \|\mathbf{v}_h\|_h \|q\|_{L^2(\Omega)}, \quad \forall \mathbf{v}_h \in X_h(\Omega), q \in Q_h(\Omega); \quad (3.20)$$

- (iii) the family  $\{X_h(\Omega), Q_h(\Omega)\}$  satisfies the inf-sup condition, that is, there exists a constant  $\beta$  that does not depend on  $h$  of triangulation such that

$$\sup_{\mathbf{v} \in X_h(\Omega)} \frac{b_h(\mathbf{v}, q)}{\|\mathbf{v}\|_h} \geq \beta \|q\|_{L^2(\Omega)}, \quad \forall q \in Q_h(\Omega), \quad (3.21)$$

so the problem (2.16) has a unique solution, and if one lets  $(\mathbf{u}, p), (\mathbf{u}_h, p_h)$  be the solution of (2.3) and (2.16), respectively, where  $(\mathbf{u}, p) \in (H_0^1(\Omega))^2 \times L_0^2(\Omega)$ ,  $\mathbf{u}|_{\Omega_k} \in (H^2(\Omega_k))^2$ ,  $p|_{\Omega_k} \in H^1(\Omega_k)$ , then

$$\|\mathbf{u} - \mathbf{u}_h\|_h + \|p - p_h\|_{L^2(\Omega)} \leq C \sum_{k=1}^N h_k \left( \|\mathbf{u}\|_{(H^2(\Omega_k))^2} + \|p\|_{H^1(\Omega_k)} \right). \quad (3.22)$$

*Proof.* The statements of Brezzi theory are that the properties (3.19)–(3.21) lead to the existence, uniqueness, and error estimate of the discrete solution. In [16], it is proven that  $a_h(\cdot, \cdot)$  is continuous on  $X_h(\Omega)$  and is elliptic with a constant uniformly bounded. Furthermore, it is straightforward that  $b_h(\cdot, \cdot)$  is continuous on  $X_h(\Omega) \times Q_h(\Omega)$ . The point that needs verification is a uniform inf-sup condition (3.21), or equivalently that the family  $\{X_h(\Omega) \times Q_h(\Omega)\}$  is div-stable.

Using local inf-sup condition (3.2) and the above lemma, arguing as the proof in Proposition 5.1 of [12], we have the global inf-sup condition (3.21).  $\square$

## 4. Numerical Algorithm

In this section, we present a numerical algorithm, that is, the  $\mathcal{W}$ -cycle multigrid method for the discrete system (2.16), and we prove the optional convergence of the multigrid method. We use a simpler and more convenient analysis method than that in [5].

In order to set the multigrid algorithm, we need only to change the index  $h$  of the partition  $\mathcal{T}_h$  in Section 2 to be  $k$ , and let  $\mathcal{T}_1$  be the coarsest partition. By connecting the opposite midpoints of the edges of the triangle, we split each triangle of  $\mathcal{T}_1$  into four triangles and we refine the partition  $\mathcal{T}_1$  into  $T_2$ . The partition  $\mathcal{T}_2$  is quasi-uniform of size  $h_2 = h_1/2$ . Repeating this process, we get a sequence of the partition  $\mathcal{T}_k (k = 1, 2, \dots, L)$ , each quasi-uniform of size  $h_k = h_1/2^{k-1}$ .

As in Section 2, with the partition  $\mathcal{T}_k$ , we define the mortar  $P_1$  nonconforming element velocity space and  $P_0$  element pressure space as  $X_k$  and  $Q_k$ , respectively. We can see that  $X_k (k = 1, 2, \dots, L)$  are nonnested, and  $Q_k (k = 1, 2, \dots, L)$  are nested. Furthermore, we denote the  $P_1$  nonconforming element product space on  $\Omega$  by  $\tilde{X}_k$ .

Let  $\{\varphi_k^i\}$  be the basis of  $X_k$ , and let  $\{\psi_k^i\}$  be the basis of  $Q_k$ . For any  $\mathbf{v}_k \in X_k$ ,  $q_k \in Q_k$ , we have the corresponding vector  $\underline{\mathbf{v}}_k = (\underline{\mathbf{v}}_{k,i})$  and  $\underline{q}_k = (\underline{q}_{k,i})$ . We introduce the matrices  $A_k$ ,  $B_k$ , and  $f_k$  having the entries  $a_{k,ij} = a(\varphi_k^i, \varphi_k^j)$ ,  $b_{k,ij} = b(\varphi_k^i, \psi_k^j)$ , and  $f_{k,i} = (\mathbf{f}, \varphi_k^i)$ , respectively. Then at level  $k$ , the problem (2.16) is equivalent to

$$\begin{pmatrix} A_k & B_k^T \\ B_k & 0 \end{pmatrix} \begin{pmatrix} \underline{\mathbf{u}}_k \\ \underline{p}_k \end{pmatrix} = \begin{pmatrix} f_k \\ 0 \end{pmatrix}. \quad (4.1)$$

In the following of this section, we introduce our multigrid method; the key of this method is the intergrid transfer operator.

We first define the intergrid transfer operator on the product space,  $L_{k-1}^k : \tilde{X}_{k-1} \rightarrow \tilde{X}_k$

$$L_{k-1}^k \mathbf{v}(m_i) = \begin{cases} \mathbf{v}(m_i), & m_i \in \kappa, \kappa \in T_{k-1}, \\ \frac{1}{2}(\mathbf{v}|_{\kappa_1}(m_i) + \mathbf{v}|_{\kappa_2}(m_i)), & m_i \in \partial\kappa_1 \cap \partial\kappa_2, \kappa_1, \kappa_2 \in T_k, \\ 0, & m_i \in \partial\Omega, \end{cases} \quad (4.2)$$

where  $\kappa, \kappa_i (i = 1, 2)$  is the partition of  $\mathcal{T}_{k-1}, \mathcal{T}_k$  respectively,  $m_i \in \Omega_{k,i}^{\text{CR}} (1 \leq i \leq N)$ .

Then we define the intergrid operator on the mortar  $P_1$  nonconforming element velocity space,  $R_{k-1}^k : X_{k-1} \rightarrow X_k$

$$R_{k-1}^k \mathbf{v} = L_{k-1}^k \mathbf{v} + \sum_{m=1}^M \Xi_{k,\delta_m(j)} L_{k-1}^k \mathbf{v}, \quad (4.3)$$

where  $\Xi_{k,\delta_m(j)}$  is defined as (3.7).

On the  $P_0$  element pressure space, we apply the natural injection operator  $J_{k-1}^k : Q_{k-1} \rightarrow Q_k$ , that is,

$$J_{k-1}^k = I. \quad (4.4)$$

Therefore, our prolongation operator on velocity space and pressure space can be written as

$$I_{k-1}^k = [R_{k-1}^k, J_{k-1}^k]. \quad (4.5)$$

### Multigrid Algorithm

If  $k = 1$ , compute the  $(\mathbf{u}_1, p_1)$  directly. If  $k \geq 2$ , do the following three steps.

*Step 1.* Presmoothing: for  $j = 0, 1, \dots, m_1 - 1$ , solving the following problem:

$$\begin{aligned} \begin{pmatrix} \underline{u}_k^{j+1} \\ \underline{p}_k^{j+1} \end{pmatrix} &= \begin{pmatrix} \underline{u}_k^j \\ \underline{p}_k^j \end{pmatrix} - \begin{pmatrix} \alpha_k I_k & B_k^T \\ B_k & 0 \end{pmatrix}^{-1} \\ &\quad \times \left\{ \begin{pmatrix} A_k & B_k^T \\ B_k & 0 \end{pmatrix} \begin{pmatrix} \underline{u}_k^j \\ \underline{p}_k^j \end{pmatrix} - \begin{pmatrix} f_k \\ 0 \end{pmatrix} \right\}, \end{aligned} \quad (4.6)$$

where  $\alpha_k$  is a real number which is not smaller than the maximal eigenvalue of  $A_k$ .

*Step 2.* Coarse grid correction: find  $(\tilde{\mathbf{u}}_{k-1}, \tilde{p}_{k-1}) \in X_{k-1} \times Q_{k-1}$ , such that

$$\begin{aligned} &a_{k-1}(\tilde{\mathbf{u}}_{k-1}, \mathbf{v}_{k-1}) + b_{k-1}(\mathbf{v}_{k-1}, \tilde{p}_{k-1}) \\ &= \langle \mathbf{f}, R_{k-1}^k \mathbf{v}_{k-1} \rangle - a_k(\mathbf{u}_k^{m_1}, R_{k-1}^k \mathbf{v}_{k-1}) - b_k(R_{k-1}^k \mathbf{v}_{k-1}, p_k^{m_1}), \quad \forall \mathbf{v}_{k-1} \in X_{k-1}, \\ &b_{k-1}(\tilde{\mathbf{u}}_{k-1}, q_{k-1}) = 0, \quad \forall q_{k-1} \in Q_{k-1}. \end{aligned} \quad (4.7)$$

Compute the approximation  $(\mathbf{u}_{k-1}^*, p_{k-1}^*)$  by applying  $\mu \geq 2$  iteration steps of the multigrid algorithm applied to the above equations on level  $k-1$  with zero starting value. Set

$$\mathbf{u}_k^{m_1+1} = \mathbf{u}_k^{m_1} + R_{k-1}^k \mathbf{u}_{k-1}^*, \quad p_k^{m_1+1} = p_k^{m_1} + p_{k-1}^*. \quad (4.8)$$

*Step 3.* PostsMOOTHing: for  $j = 0, 1, \dots, m_2 - 1$  solving following problem:

$$\begin{aligned} \begin{pmatrix} \underline{u}_k^{m_1+j+2} \\ \underline{p}_k^{m_1+j+2} \end{pmatrix} &= \begin{pmatrix} \underline{u}_k^{m_1+j+1} \\ \underline{p}_k^{m_1+j+1} \end{pmatrix} - \begin{pmatrix} \alpha_k I_k & B_k^T \\ B_k & 0 \end{pmatrix}^{-1} \\ &\quad \times \left\{ \begin{pmatrix} A_k & B_k^T \\ B_k & 0 \end{pmatrix} \begin{pmatrix} \underline{u}_k^{m_1+j+1} \\ \underline{p}_k^{m_1+j+1} \end{pmatrix} - \begin{pmatrix} f_k \\ 0 \end{pmatrix} \right\}, \end{aligned} \quad (4.9)$$

then,  $(\mathbf{u}_k^{m_1+m_2+1}, p_k^{m_1+m_2+1})$  is the result of one iteration step.

For convenience, at level  $k$  the problem (2.16) can be written as follows: find  $(\mathbf{u}_k, p_k) \in X_k \times Q_k$  such that

$$L_{h,k}((\mathbf{u}_k, p_k); (\mathbf{v}_k, q_k)) = F_k((\mathbf{v}_k, q_k)), \quad \forall (\mathbf{v}_k, q_k) \in X_k \times Q_k. \quad (4.10)$$

Since  $L_{h,k}((\mathbf{u}_k, p_k); (\mathbf{v}_k, q_k))$  is a symmetric bilinear form on  $X_k \times Q_k$ , there is a complete set of eigenfunctions  $(\phi_k^j, \psi_k^j)$ , which satisfy

$$\begin{aligned} L_{h,k}((\mathbf{u}_k, p_k); (\mathbf{v}_k, q_k)) &= \lambda_j \left[ (\phi_k^j, \mathbf{v}_k)_0 + h^2 (\psi_k^j, q_k)_0 \right], \quad \forall (\mathbf{v}_k, q_k) \in X_k \times Q_k, \\ (\mathbf{v}_k, q_k) &= \sum_j c_j (\phi_k^j, \psi_k^j). \end{aligned} \quad (4.11)$$

In order to verify that our multigrid algorithm is optimal, we need to define a set of mesh-dependent norms. For each  $k \geq 0$  we equip  $X_k \times Q_k$  with the norm

$$\| |(\mathbf{v}, q)| \|_{0,k} = \| (\mathbf{v}, q) \|_{0,k} = \left( \|\mathbf{v}\|_{L^2(\Omega)}^2 + h_k^2 \|q\|_{L^2(\Omega)}^2 \right)^{1/2} = \left( (\mathbf{v}, \mathbf{v})_k + h_k^2 (q, q)_k \right)^{1/2}, \quad (4.12)$$

and define

$$\| |(\mathbf{v}_k, q_k)| \|_{s,k} = \left\{ \sum_j |\lambda_j|^s |c_j|^2 \right\}^{1/2}, \quad \|\mathbf{v}\|_k^2 = \sum_\tau (\nabla \mathbf{v}, \nabla \mathbf{v})_k. \quad (4.13)$$

For our multigrid algorithm, we have the following optional convergence conclusion.

**Theorem 4.1.** *If  $(\mathbf{u}, p)$  and  $(\mathbf{u}_h^i, p_h^i)$  ( $0 \leq i \leq m+1$ ) are the solutions of problems (2.16) and (4.10), respectively, then there exists a constant  $0 < \gamma < 1$  and positive integer  $m$ , all are independent of the level number  $k$ , such that*

$$\| |(\mathbf{u}, p) - (\mathbf{u}_k^{m+1}, p_k^{m+1})| \|_{0,k} \leq \gamma \| |(\mathbf{u}, p) - (\mathbf{u}_k^0, p_k^0)| \|_{0,k}. \quad (4.14)$$

To prove this theorem, we give in the next section two basic properties for convergence analysis of the multigrid, that is, the smoothing property and approximation property.

## 5. Proof of Theorem 4.1

From the standard multigrid theory, the  $\mathcal{W}$ -cycle yields a  $h$ -independent convergence rate based on the following two basic properties.

We first show the smoothing property. By [[12] Theorem 5.1], we have the following.

**Lemma 5.1** (smoothing property). *Assume that  $\lambda_{\max}(A_k) \leq \alpha_k \leq C \lambda_{\max}(A_k)$ , if the number of smoothing steps is  $m$ , then*

$$\| |(\mathbf{u}_h^m - \mathbf{u}_h, p_h^m - p_h)| \|_{2,k} \leq \frac{Ch^{-2}}{m} \left\| \mathbf{u}_h^0 - \mathbf{u}_h \right\|_{L^2(\Omega)}. \quad (5.1)$$

*The property has been proved in [11].*

Next, we prove the approximation property. We just apply the following conclusion in [14], which can simplify the complexity of theoretical analysis.

**Lemma 5.2.** *If the prolongation operator  $I_{k-1}^k$  defined in (4.5) satisfies the following criterion, Then, the approximation property in multigrid method holds and the multigrid algorithm converges optimally.*

$$(A.1) \| \mathbf{v} - R_{k-1}^k \mathbf{v} \|_{L^2(\Omega)} \leq C h_k \| \mathbf{v} \|_{k-1}, \quad \forall \mathbf{v} \in X_{k-1},$$

$$(A.2) \| J_{k-1}^k q \|_{L^2(\Omega)} \leq C \| q \|_{L^2(\Omega)}, \quad \forall q \in Q_{k-1},$$

$$(A.3) \| |(\mathbf{u}_k, p_k) - I_{k-1}^k (\mathbf{u}_{k-1}, p_{k-1})| \|_{0,k} \leq C h_k^2 (\| \mathbf{u} \|_{H^2(\Omega)} + \| p \|_{H^1(\Omega)}).$$

where  $(\mathbf{u}, p) \in (H_0^1(\Omega) \cap H^2(\Omega))^2 \times (L_0^2(\Omega) \cap H^1(\Omega))$  is the solution of (2.3) with the force term  $\mathbf{f} \in (L^2(\Omega))^2$  and  $(\mathbf{u}_{k-1}, p_{k-1})$ ,  $(\mathbf{u}_k, p_k)$  are the mixed finite element approximation of  $(\mathbf{u}, p)$  at levels  $k-1$  and  $k$ , respectively.

This lemma has been proved in [14].

**Lemma 5.3** (approximation property). *Let  $(I_{k-1}^k)^* : X_k \times Q_k \rightarrow X_{k-1} \times Q_{k-1}$  ( $k \geq 1$ ) be defined as follows:*

$$\begin{aligned} & L_{k-1} \left( \left( I_{k-1}^k \right)^* (\mathbf{v}_k, q_k), (\mathbf{v}_{k-1}, q_{k-1}) \right), \\ & = L_k \left( (\mathbf{v}_k, q_k), I_{k-1}^k (\mathbf{v}_{k-1}, q_{k-1}) \right), \quad \forall (\mathbf{v}_{k-1}, q_{k-1}) \in X_{k-1} \times Q_{k-1}, (\mathbf{v}_k, q_k) \in X_k \times Q_k. \end{aligned} \quad (5.2)$$

Then one has

$$| | | (\mathbf{v}, q) - I_{k-1}^k \left( I_{k-1}^k \right)^* (\mathbf{v}, q) | | |_{0,k} \leq C h_k^2 | | (\mathbf{v}, q) | |_{2,k}, \quad \forall (\mathbf{v}, q) \in X_k \times Q_k. \quad (5.3)$$

*Proof.* By Lemma 5.2, we only need to prove our prolongation operator  $I_{k-1}^k$  that satisfies (A.1), (A.2), and (A.3).

For any  $\mathbf{v} \in X_{k-1}$ , the inequality (A.1) holds. In fact

$$| | | \mathbf{v} - R_{k-1}^k \mathbf{v} | | |_{L^2(\Omega)} \leq | | | \mathbf{v} - L_{k-1}^k \mathbf{v} | | |_{L^2(\Omega)} + \left| \left| \sum_{m=1}^M \Xi_{k,\delta_m(j)} L_{k-1}^k \mathbf{v} \right| \right|_{L^2(\Omega)}, \quad (5.4)$$

by Lemma 5.2 in [14], we can get

$$| | | \mathbf{v} - L_{k-1}^k \mathbf{v} | | |_{L^2(\Omega)} \leq C h_k \| \mathbf{v} \|_{k-1}, \quad (5.5)$$

by norm equivalence, we deduce

$$\begin{aligned}
\left\| \Xi_{k,\delta_{m(j)}} L_{k-1}^k \mathbf{v} \right\|_{L^2(\Omega)}^2 &\leq h_k^2 \sum_{m_i^k \in \delta_{k,m(j)}^{\text{CR}}} \left( \Xi_{k,\delta_{m(j)}} L_{k-1}^k \mathbf{v} \right)^2(m_i^k) \\
&= h_k^2 \sum_{m_i^k \in \delta_{k,m(j)}^{\text{CR}}} Q_{k,\delta_{m(j)}} \left( (L_{k-1}^k \mathbf{v})|_{\gamma_{m(i)}} - (L_{k-1}^k \mathbf{v})|_{\delta_{m(j)}} \right)^2(m_i^k) \\
&\leq Ch_k \left\| Q_{k,\delta_{m(j)}} \left( (L_{k-1}^k \mathbf{v})|_{\gamma_{m(i)}} - (L_{k-1}^k \mathbf{v})|_{\delta_{m(j)}} \right) \right\|_{0,\gamma_m}^2 \\
&\leq Ch_k \left\| (L_{k-1}^k \mathbf{v})|_{\gamma_{m(i)}} - (L_{k-1}^k \mathbf{v})|_{\delta_{m(j)}} \right\|_{0,\gamma_m}^2 \\
&\leq Ch_k \left( \left\| (L_{k-1}^k \mathbf{v})|_{\gamma_{m(i)}} - \mathbf{v}|_{\delta_{m(j)}} \right\|_{0,\gamma_m}^2 + \left\| \mathbf{v}|_{\delta_{m(j)}} - (L_{k-1}^k \mathbf{v})|_{\delta_{m(j)}} \right\|_{0,\gamma_m}^{2.0} \right) \\
&= Ch_k(K_1 + K_2).
\end{aligned} \tag{5.6}$$

Using trace theorem and (5.5), we have

$$K_2 \leq Ch_k \|\mathbf{v}\|_{k-1,j}^2. \tag{5.7}$$

Owing to  $\mathbf{v} \in X_{k-1}$ , then

$$\begin{aligned}
\left\| (L_{k-1}^k \mathbf{v})|_{\gamma_{m(i)}} - \mathbf{v}|_{\delta_{m(j)}} \right\|_{0,\delta_{m(j)}}^2 &\leq 2 \left\| (L_{k-1}^k \mathbf{v})|_{\gamma_{m(i)}} - Q_{k-1,\delta_{m(j)}}(\mathbf{v}|_{\gamma_{m(i)}}) \right\|_{0,\gamma_{m(i)}}^2 \\
&\quad + 2 \left\| Q_{k-1,\delta_{m(j)}}(\mathbf{v}|_{\delta_{m(j)}}) - \mathbf{v}|_{\delta_{m(j)}} \right\|_{0,\delta_{m(j)}}^2.
\end{aligned} \tag{5.8}$$

The second term of the above inequality can be estimated as follows:

$$\left\| Q_{k-1,\delta_{m(j)}}(\mathbf{v}|_{\delta_{m(j)}}) - \mathbf{v}|_{\delta_{m(j)}} \right\|_{0,\gamma_m}^2 = \sum_{e \in T_{k-1}(\delta_{m(j)})} \int_e (\mathbf{v} - Q_e \mathbf{v})^2 ds, \tag{5.9}$$

where  $Q_e$  is the  $L^2$  orthogonal projection onto one-dimensional space which consists of constant functions on an element  $e$ , and  $e$  is an edge of  $E$  which is in the triangulation  $T_{k-1}$ . Using the scaling argument in [17], for any constant  $c$  we have

$$\begin{aligned}
\int_e (\mathbf{v} - Q_e \mathbf{v})^2 ds &\leq \int_e (\mathbf{v} - c)^2 ds \leq Ch_k \int_{\hat{e}} (\hat{\mathbf{v}} - c)^2 d\hat{s} \leq Ch_k \|\hat{\mathbf{v}} - c\|_{1,\hat{E}}^2 \\
&\leq Ch_k |\hat{\mathbf{v}}|_{1,\hat{E}}^2 \leq Ch_k |\mathbf{v}|_{1,E}^2,
\end{aligned} \tag{5.10}$$

which combining with (5.9) gives

$$\left\| Q_{k-1,\delta_{m(j)}}(\mathbf{v}|_{\delta_{m(j)}}) - \mathbf{v}|_{\delta_{m(j)}} \right\|_{0,\gamma_m} \leq Ch_k^{1/2} \|\mathbf{v}\|_{k,j}. \quad (5.11)$$

For the first term of the right side of (5.8), we have

$$\begin{aligned} & \left\| (L_{k-1}^k \mathbf{v})|_{\gamma_{m(i)}} - Q_{k-1,\delta_{m(j)}}(\mathbf{v}|_{\gamma_{m(i)}}) \right\|_{0,\gamma_m}^2 \\ &= \left\| (L_{k-1}^k \mathbf{v})|_{\gamma_{m(i)}} - \mathbf{v}|_{\gamma_{m(i)}} + \mathbf{v}|_{\gamma_{m(i)}} + Q_{k-1,\delta_{m(j)}}(\mathbf{v}|_{\gamma_{m(i)}}) \right\|_{0,\gamma_m}^2 \\ &\leq 2 \left\| (L_{k-1}^k \mathbf{v} - \mathbf{v})|_{\gamma_{m(i)}} \right\|_{0,\gamma_{m(i)}}^2 + 2 \left\| \mathbf{v}|_{\gamma_{m(i)}} - Q_{k-1,\delta_{m(j)}}(\mathbf{v}|_{\gamma_{m(i)}}) \right\|_{0,\gamma_{m(i)}}^2 \\ &= F_1 + F_2. \end{aligned} \quad (5.12)$$

Trace theorem and (5.5) give

$$F_1 \leq Ch_k \left\| L_{k-1}^k \mathbf{v} \right\|_{k,i}^2 \leq Ch_k \|\mathbf{v}\|_{k-1,i}^2. \quad (5.13)$$

For  $F_2$ , by trace theorem and the approximation of the operator  $Q_{k-1,\delta_{m(j)}}$ , we have

$$F_2 \leq Ch_k \|\mathbf{v}\|_{k-1,i}^2, \quad (5.14)$$

which together with (5.4)–(5.13), gives (A.1).

Obviously, (A.2) naturally holds so we only need to prove (A.3).

By proof of Lemma 5.2 in [14], we can see that

$$\begin{aligned} & \left\| (\mathbf{u}_k, p_k) - I_{k-1}^k(\mathbf{u}_{k-1}, p_{k-1}) \right\|_{0,k} \\ & \leq \left\| \mathbf{u}_k - L_{k-1}^k \mathbf{u}_{k-1} \right\|_{0,k} + \left\| \sum_{m=1}^M \Xi_{k,\delta_{m(j)}} L_{k-1}^k \mathbf{u}_{k-1} \right\|_{0,k} + h_k^2 \left\| p_k - J_{k-1}^k p_{k-1} \right\|_{0,k} \\ & \leq Ch_k^2 \left( \|\mathbf{u}\|_{H^2(\Omega)} + \|p\|_{H^1(\Omega)} \right) + \left\| \sum_{m=1}^M \Xi_{k,\delta_{m(j)}} L_{k-1}^k \mathbf{u}_{k-1} \right\|_{0,k}. \end{aligned} \quad (5.15)$$

Arguing as (5.6), we obtain

$$\begin{aligned}
& \left\| \Xi_{k,\delta_{m(j)}} L_{k-1}^k \mathbf{u}_{k-1} \right\|_{0,k}^2 \\
& \leq h_k^2 \sum_{m_i^k \in \delta_{k,m(j)}^{\text{CR}}} \Xi_{k,\delta_{m(j)}} \left( L_{k-1}^k \mathbf{u}_{k-1} \right)^2 \left( m_i^k \right) \\
& = h_k^2 \sum_{m_i^k \in \delta_{k,m(j)}^{\text{CR}}} \left( Q_{k,\delta_{m(j)}} \left( \left( L_{k-1}^k \mathbf{u}_{k-1} \right) |_{\gamma_{m(i)}} - \left( L_{k-1}^k \mathbf{u}_{k-1} \right) |_{\delta_{m(j)}} \right) \right)^2 \left( m_i^k \right) \\
& \leq Ch_k \left\| Q_{k,\delta_{m(j)}} \left( \left( L_{k-1}^k \mathbf{u}_{k-1} \right) |_{\gamma_{m(i)}} - \mathbf{u}_k |_{\gamma_{m(i)}} + \mathbf{u}_k |_{\delta_{m(j)}} - \left( L_{k-1}^k \mathbf{u}_{k-1} \right) |_{\delta_{m(j)}} \right) \right\|_{0,k}^2 \\
& \leq Ch_k \left( \left\| \left( L_{k-1}^k \mathbf{u}_{k-1} \right) |_{\gamma_{m(i)}} - \mathbf{u}_k |_{\gamma_{m(i)}} \right\|_{0,\gamma_m}^2 + \left\| \left( L_{k-1}^k \mathbf{u}_{k-1} \right) |_{\delta_{m(j)}} - \mathbf{u}_k |_{\delta_{m(j)}} \right\|_{0,\gamma_m}^2 \right) \\
& = Ch_k (K_1 + K_2).
\end{aligned} \tag{5.16}$$

By (5.15) and trace theorem, we get that

$$K_1 \leq Ch_k^3 \|\mathbf{u}\|_{H^2(\Omega_i)}^2, \quad K_2 \leq Ch_k^3 \|\mathbf{u}\|_{H^2(\Omega_j)}^2, \tag{5.17}$$

together with (5.15), (A.3) has been proved, and we have completed the proof of Lemma 5.3.  $\square$

## 6. Numerical Results

In this section, we present some numerical results to illustrate the theory developed in the earlier sections. The examples are as same as those in [5], so that we can compare the conclusion with the mortar rotated  $Q_1$  element method.

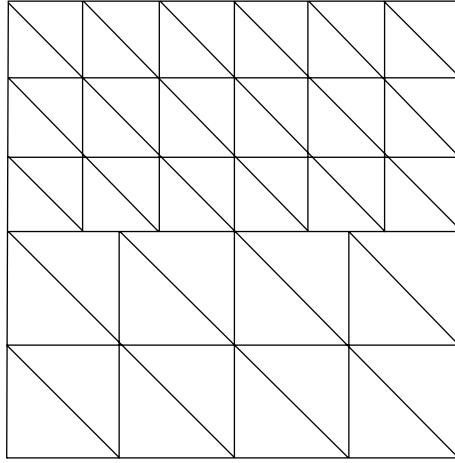
Here we deal with  $\Omega = (0, 1)^2$ . We choose the exact solution of (2.1) as

$$u_1 = 2x^2(1-x)^2y(1-y)(1-2y), \quad u_2 = -2x(1-x)(1-2x)y^2(1-y)^2, \tag{6.1}$$

for the velocity and  $p = x^2 - y^2$  for the pressure.

For simplicity, we decompose  $\Omega$  into two subdomains:  $\Omega_1 = (0, 1) \times (0, 1/2)$  as nonmortar domain and  $\Omega_2 = (0, 1) \times (1/2, 1)$  as mortar domain. The sizes of the coarsest grid are denoted by  $h_{1,1}$  and  $h_{1,2}$ , respectively (see Figure 1). The test concerns the convergence of the  $\mathcal{W}$ -cycle multigrid algorithm. In what follows,  $k$  denotes the level,  $N_u$  and  $N_p$  are the number of the unknowns of the velocity and pressure, the norm  $\|\cdot\|_{0,d}$  is the usual Euclidean norm of a vector which is equivalent to  $\|\cdot\|_h$ .  $\text{iter}_{(m_1, m_2)}$  denotes the number of iterations to achieve the relative error of residue less than  $10^{-3}$ , where  $m_1$  and  $m_2$  are the presmoothing steps, and the postsmothing steps respectively, and the initial approximative solution for the iteration is zero. The numerical results are presented in Tables 1 and 2.

From Table 1, we can see that the errors of the mortar element method for the velocity and the pressure are small, which demonstrates Theorem 3.2.



**Figure 1:** The coarsest mesh with  $h_{1,1} = 1/4$  and  $h_{1,2} = 1/6$ .

**Table 1:** Error estimate for the mortar element method with  $h_{1,1} = 1/4$  and  $h_{1,2} = 1/6$ .

$k$	$N_u$	$N_p$	$\ \underline{u} - \underline{u}_k\ _{0,d}$	$\ p - p_k\ _{L^2(\Omega)}$
1	178	51	0.0772974	0.164013
2	668	207	0.0455288	0.133103
3	2584	831	0.0242334	0.0733174
4	10160	3327	0.0123836	0.037679
5	40288	13311	0.00619081	0.0195834

**Table 2:** Iterative numbers for the  $\mathcal{W}$ -cycle with  $h_{1,1} = 1/4$  and  $h_{1,2} = 1/6$ .

$k$	2	3	4	5
iter <sub>(4,4)</sub>	9	8	8	9
iter <sub>(5,5)</sub>	8	8	7	7

From Table 2, we can see that the convergence for the  $\mathcal{W}$ -cycle multigrid algorithm is optimal; that is, the number of iterations is independent of the level number  $k$ . Meanwhile, we note that the number of iterations is less than the rotated  $Q_1$  element method in [5] when achieving the same relative error.

## Acknowledgment

The authors would like to express their sincere thanks to the referees for many detailed and constructive comments. The work is supported by the National Natural Science Foundation of China under Grant 11071124.

## References

- [1] V. Girault and P.-A. Raviart, *Finite Element Methods for Navier-Stokes Equations*, vol. 5 of *Springer Series in Computational Mathematics*, Springer, Berlin, Germany, 1986.

- [2] S. Zhang, "On the  $P_1$  Powell-Sabin divergence-free finite element for the Stokes equations," *Journal of Computational Mathematics*, vol. 26, no. 3, pp. 456–470, 2008.
- [3] T. Neckel, M. Mehl, and C. Zenger, "Enhanced divergence-free elements for efficient incompressible flow simulations in the PDE framework peano," in *Proceedings of the 5th European Conference on Computational Fluid Dynamics (ECCOMAS CFD '10)*, pp. 14–17, Lisbon, Portugal, June 2010.
- [4] D. Braess and R. Sarazin, "An efficient smoother for the Stokes problem," *Applied Numerical Mathematics*, vol. 23, no. 1, pp. 3–20, 1997.
- [5] J. Chen and P. Huang, "Analysis of mortar-type  $Q_1^{\text{rot}}/Q_0$  element and multigrid methods for the incompressible Stokes problem," *Applied Numerical Mathematics*, vol. 57, no. 5-7, pp. 562–576, 2007.
- [6] V. John and L. Tobiska, "A coupled multigrid method for nonconforming finite element discretizations of the 2D-Stokes equation," *Computing*, vol. 64, no. 4, pp. 307–321, 2000.
- [7] J. Li, "A dual-primal FETI method for incompressible Stokes equations," *Numerische Mathematik*, vol. 102, no. 2, pp. 257–275, 2005.
- [8] H. H. Kim, C.-O. Lee, and E.-H. Park, "A FETI-DP formulation for the Stokes problem without primal pressure components," *SIAM Journal on Numerical Analysis*, vol. 47, no. 6, pp. 4142–4162, 2010.
- [9] J. Li and O. Widlund, "BDDC algorithms for incompressible Stokes equations," *SIAM Journal on Numerical Analysis*, vol. 44, no. 6, pp. 2432–2455, 2006.
- [10] C. Bernardi, Y. Maday, and A. T. Patera, "A new nonconforming approach to domain decomposition: the mortar element method," in *Nonlinear Partial Differential Equations and Their Applications*, H. Brezis and J. Lions, Eds., vol. 11 of *Pitman Research Notes in Mathematics* 299, pp. 13–51, Longman Scientific Technical, Harlow, UK, 1994.
- [11] D. Braess, W. Dahmen, and C. Wieners, "A multigrid algorithm for the mortar finite element method," *SIAM Journal on Numerical Analysis*, vol. 37, no. 1, pp. 48–69, 1999.
- [12] F. Ben Belgacem, "The mixed mortar finite element method for the incompressible Stokes problem: convergence analysis," *SIAM Journal on Numerical Analysis*, vol. 37, no. 4, pp. 1085–1100, 2000.
- [13] F. Wang, J. Chen, W. Xu, and Z. Li, "An additive Schwarz preconditioner for the mortar-type rotated  $Q_1$  FEM for elliptic problems with discontinuous coefficients," *Applied Numerical Mathematics*, vol. 59, no. 7, pp. 1657–1667, 2009.
- [14] Q.-p. Deng and X.-p. Feng, "Multigrid methods for the generalized Stokes equations based on mixed finite element methods," *Journal of Computational Mathematics*, vol. 20, no. 2, pp. 129–152, 2002.
- [15] F. Brezzi, "On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers," *RAIRO Analyse Numérique*, vol. 8, pp. 129–151, 1974.
- [16] L. Marcinkowski, "The mortar element method with locally nonconforming elements," *BIT*, vol. 39, no. 4, pp. 716–739, 1999.
- [17] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, vol. 4, North-Holland, Amsterdam, The Netherlands, 1978.

*Research Article*

## **Applying Neural Networks to Prices Prediction of Crude Oil Futures**

**John Wei-Shan Hu,<sup>1,2</sup> Yi-Chung Hu,<sup>2</sup> and Ricky Ray-Wen Lin<sup>3</sup>**

<sup>1</sup> Department of Finance, Chung Yuan Christian University, Chung-Li 32023, Taiwan

<sup>2</sup> Department of Business Administration, Chung Yuan Christian University, Chung-Li 32023, Taiwan

<sup>3</sup> Tsann Kuen Trans-Nation Group, Taiwan

Correspondence should be addressed to Yi-Chung Hu, ychu@cycu.edu.tw

Received 2 March 2012; Accepted 12 June 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 John Wei-Shan Hu 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 global economy experienced turbulent uneasiness for the past five years owing to large increases in oil prices and terrorist's attacks. While accurate prediction of oil price is important but extremely difficult, this study attempts to accurately forecast prices of crude oil futures by adopting three popular neural networks methods including the multilayer perceptron, the Elman recurrent neural network (ERNN), and recurrent fuzzy neural network (RFNN). Experimental results indicate that the use of neural networks to forecast the crude oil futures prices is appropriate and consistent learning is achieved by employing different training times. Our results further demonstrate that, in most situations, learning performance can be improved by increasing the training time. Moreover, the RFNN has the best predictive power and the MLP has the worst one among the three underlying neural networks. This finding shows that, under ERNNs and RFNNs, the predictive power improves when increasing the training time. The exceptional case involved BPNs, suggesting that the predictive power improves when reducing the training time. To sum up, we conclude that the RFNN outperformed the other two neural networks in forecasting crude oil futures prices.

### **1. Introduction**

During the past three years, the global economy has experienced dramatic turbulence owing to uneasiness because of terrorists' attacks and rapidly rising oil prices. For example, the US light crude oil futures price rapidly climbed to the all-time peak about US\$80 recently. Simultaneously, the US Federal Reserve continuously increased its benchmark short-term interest rates by seventeen times to prevent inflation till August 2006. Consequently, many governments and corporate managers attempted to seek a method of accurately forecasting the crude oil prices.

Accurate prediction of crude oil price is important yet extremely complicated and difficult. For example, Kumar [1] found that the traditional model-based forecasts had larger errors compared to forecast crude oil prices using futures price. However, Pham and Liu [2] and Refenes et al. [3] showed that neural networks had significant performance of forecasting. According to Chen and Pham [4], numerous real-world application problems cannot be fully described and handled via classical set theory. Meanwhile, fuzzy set theory can deal with partial membership. Although Omlin et al. [5] argued that fuzzy neural networks (FNNs) combine the advantages of both fuzzy systems and neural networks, whereas Omlin et al. [5] proposed that most of the FNNs could only process state input-output relationships, FNNs were unable to process temporal input sequences with arbitrary length. Since recurrent neural networks (RNNs) are dynamic systems involving temporal state representation, RNNs are computationally powerful. Although the Elman recurrent neural network (ERNN) is a special case of RNN and is less efficient than standard RNN, Pham and Liu [2] posited that ERNN can model a very large class of linear and nonlinear dynamic systems. Additionally, Lee and Teng [6] argued that the RFNNs have the same advantages as RNNs and extended the application domain of the FNNs to temporal problems. These findings motivate us to apply three neural network models (namely, traditional backpropagation neural networks (BPNs), Elman recurrent neural networks (ERNNs), and recurrent fuzzy neural networks (RFNNs)) to forecast crude oil futures prices.

The focus of this paper is to apply neural networks for predicting crude oil futures prices. This work has the following objectives: forecast crude oil futures prices using BPNs, ERNNs, and RFNNs; compare the learning and predictive performance among BPNs, ERNNs, and RFNNs, and explore how training time affects prediction accuracy.

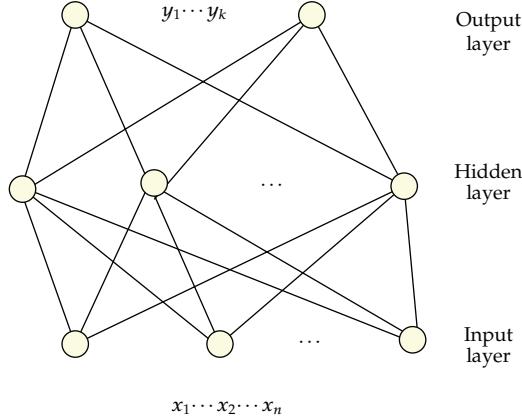
This study classifies the previous literature into three main groups: (1) the studies that compared artificial neural networks (ANNs) with other methods to forecast futures prices, (2) the works that combined fuzzy systems with recurrent neural networks, and (3) the researches that examined the evolution or forecasting accuracy of energy futures prices.

The following studies have applied various ANNs to predict futures prices. Refenes et al. [3], Castillo and Melin [7], Giles et al. [8], Donaldson and Kamstra [9], and Sharma et al. [10] all demonstrated that neural networks outperformed classical statistical techniques in forecasting ability. Although Kamstra and Boyd [11] also found that ANNs outperformed the naive model for most commodities in forecasting ability, yet Kamstra and Boyd found that ANNs have less predictive power than linear model for barley and rye.

The following works combined fuzzy system with recurrent neural networks. Omlin et al. [5], Juang and Lin [12], Nürnberger et al. [13, 14], Zhang and Morris [15], Giles et al. [8], Lee and Teng [6], Mastorocostas and Theocharis [16, 17], Juang [18], Yu and Ferreyra [19], Hong and Lee [20], and Lin and Chen [21] all designed to combine recurrent neural networks (RNNs) with fuzzy system for identification and prediction.

The following researches examined the evolution or forecasting accuracy of energy prices. Hirshfeld [22], Ma [23], Serletis [24], Kumar [1], Pindyck [25], Adrangi et al. [26], and Krehbiel and Adkins [27] reviewed or examined the energy futures prices and the price risk. Among these literatures, Adrangi et al. [26] found strong evidence of nonlinear dependencies in crude oil futures prices, but the evidence is not consistent with chaos.

The paper is organized as follows. Three kinds of artificial neural networks are described in Section 2. The performance of the proposed learning algorithms is examined by the computer simulations are described in Section 4. Conclusion is presented in Section 5. The performance valuation method is presented in Section 3.



**Figure 1:** Structure of multilayer perceptron.

## 2. Neural Networks for Crude Oil Forecasting

### 2.1. Multilayer Perceptron (MLP)

As a promising generation of information processing system that expresses the ability to learn, recall, and generalize based on training patterns or data, artificial neural networks (ANNs) are interconnected assembly of simple processing nodes, whose functionality is similar to human neurons. ANNs have become popular during the last two decades for diverse applications, ranging from financial prediction to machine vision. According to Refenes et al. [3], the main potentials with respect to ANNs include the following: (1) handling complex nonlinear function; (2) learning from training data; (3) adapting to changing environments; (4) handling incomplete, noisy, and fuzzy information; (5) performing high-speed information processing. The most popular and widespread method used to train the multilayer perceptron (MLP) is the back propagation algorithm. MLP can be interpreted as a universal approximators and is used to estimate the parameter values via a gradient descent algorithm in problems involving nonlinear regression. The popularity of MLP is based on the simplicity and power of the underlying algorithm. Figure 1 shows the structure of MLP.

BPN involves two steps. The first step generates a forward flow of activation from the input layer to the output layer via the hidden layer.

The sigmoid function is usually served as

$$y = \frac{1}{(1 + e^{-\alpha \sum x_i})}, \quad (2.1)$$

where  $\alpha \in R$ . An activation function can be differentiated since the steepest descent method is employed to derive the weight updating rule. The response of the hidden layer is the input of the output layer. In the second step, an overall error,  $E_T$ , which is the difference between the actual and the desired output, is minimized employing a supervised learning task performed by MLP:

$$E_T = \sum_p E_p = \frac{1}{2} \sum_p \sum_k (d_{pk} - y_{pk})^2, \quad (2.2)$$

where  $E_T$  denotes the total error for a neural network across the entire training set,  $E_p$  represents the network error for the  $p$ th pattern,  $d_{pk}$  denotes the desired output of the  $k$ th unit in the output layer for pattern  $p$ , and  $y_{pk}$  is the actual output of the  $k$ th unit in the output layer for the  $p$ th pattern.

Then the gradient method is applied to optimize the weight vector of  $E_T$  to minimize the summed square error between the actual and the desired network outputs throughout the training period. The network weight is adjusted whenever a training data is inputted. The size of the adjustment is positively related to the sensitivity of the error function to weight connections. The general weight updating rule for the connection weight between the  $i$ th input node and the  $j$ th output node is as follows:

$$\Delta w_{ji} = -\eta \frac{\partial E_T}{\partial w_{ji}}, \quad (2.3)$$

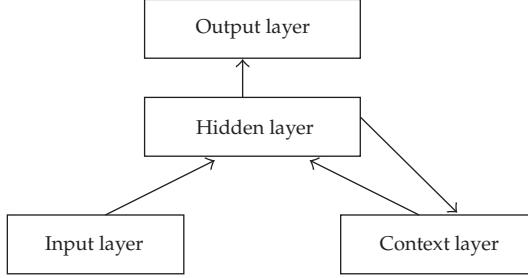
where  $\eta$  is the learning rate.

BPNs can be widely applied to sample identification, pattern matching, compression, classification, diagnosis, credit rating, stock price trend forecasting, adaptive control, functional link, optimization, and data clustering. They can also be trained via a supervised learning task to reduce the difference between the desired and the actual outputs and have high learning accuracy. Yet BPNs have the following weaknesses: (1) slow learning speed, (2) long executing time, (3) very slow convergence; (4) falling into a local minimum of error functions, (5) lack of systematic methods in the network dynamics, (6) inability to use past experience to forecast its future behavior. This study further uses two dynamic neural networks to predict the crude oil futures prices.

## 2.2. Recurrent Neural Networks

Recurrent neural networks (RNNs) were first developed by Hopfield [28] in a single-layer form and later were developed using multilayer perceptrons comprising concatenated input-output, processing, and output layers. An RNN is a dynamic neural network that permits self-loops and backward connections so that the neurons have recurrent actions and provide local memory functions. The feedback within RNN can be achieved either locally or globally. The ability of feedback with delay provides memory to the network and is appropriate for prediction. According to Haykin [29], RNN can summarize all required information on past system behavior to forecast future behavior. The ability of RNN to dynamically incorporate past experience based on internal recurrence makes it more powerful than BPN. The structure of RNN generally comprises the following: (a) recurring information from the output or hidden layer to the input layer, and (b) mutual connection of neurons within the same layer. The advantages of RNNs are as follows: (1) fast learning speed, (2) short executing time, and (3) fast converging to a stable state.

Refenes et al. [3] argued that RNNs exhibited full connection between each node and all other nodes in the network, whereas partial recurrent networks contain a number of specific feedback loops. They quoted Hertz et al. [30] who assigned the name Elman architecture when the feedback to the network input is from one or more hidden layers. This name originated from Elman [31] who designed a neural network with recurrent links for providing networks with dynamic memory. The Elman recurrent neural network (ERNN) is a temporal and simple recurrent network with a hidden layer, assuming that the neural network operates



**Figure 2:** Structure of the simple recurrent neural network.

using discrete time steps. The activations of the hidden units at time  $t$  are fed backwards and serve as inputs to “context layer” at time  $t+1$  and thus represent a form of short-term memory that enables limited recurrence. Moreover, the feedback links run from the hidden layer to the context layer and produce both temporal and spatial patterns. As depicted in Figure 2, this network is a two-layer network involving feedback in the first layer [32].

### 2.3. Elmann Recurrent Neural Network

According to Hammer and Nørskov [33], ERNN is a special case of RNN, differing mainly in that the learning algorithm is simply a truncated gradient descent method and training is less efficient than the standard method employed for RNN. However, Pham and Liu [2] argued that ERNN can be used to model a very large class of linear and nonlinear dynamic systems. Li et al. [34] designed a new nonlinear neuron-activation function into the framework of an Elman network, thus enabling the neural network to generate chaos. Li et al. [34] described a context layer  $x^c(\kappa)$  in the Elman network as follows:

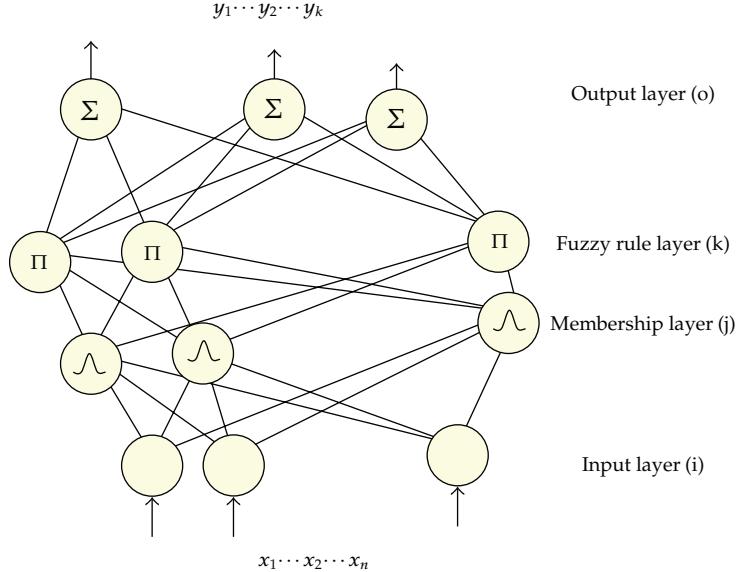
$$x^c(\kappa) = x(\kappa - 1). \quad (2.4)$$

The descriptive equations of ERNN can be considered as a nonlinear state-space model in which all weight values are constant following initialization:

$$x_i(\kappa) = f \left( \sum_{j=1}^N w_{ij}^x x_j(\kappa - 1) + w_i^u u(\kappa - 1) \right), \quad (2.5)$$

$$y(\kappa) = \sum_{i=1}^n w_i^y x_i(\kappa), \quad (2.6)$$

where  $w_{ij}^x$  demonstrates weight linking the  $i$ th hidden-layer neuron and the  $j$ th context-layer neuron,  $w_i^u$  indicates weight linking the input neuron  $u(\kappa - 1)$  and the  $i$ th hidden-layer neuron,  $w_i^y$  refers to weight linking the output neuron  $y(\kappa)$  and the  $i$ th hidden-layer neuron,  $f(\cdot)$  expresses nonlinear activation function in the hidden-layer node, and  $n$  is the number of hidden-layer nodes. Since it is difficult to interpret the network functions of RNNs, this study further incorporates the fuzzy logic into RNNs.



**Figure 3:** A structure of the recurrent fuzzy neural networks.

## 2.4. Recurrent Fuzzy Neural Networks

Fuzzy sets theory has first been introduced by Zadeh [35, 36]. Zimmermann [37] argued that fuzzy set theory can be adapted to different circumstances and contexts. Buckley and Hayashi [38] stated that fuzzy neural network (FNN) is a layered, feedforward, neural net that has fuzzy set signals and weights. Neural networks may utilize the data bank to train and learn, while the solution obtained by fuzzy logic may be verified by empirical study and optimization. Omlin et al. [5] noted that FNN comprises both clear physical meanings and good training ability. However, FNN only applies to static problems.

Besides the fact that RNN's underlying theory is complicated and RNN is difficult to interpret, Hu and Chang [39] also found that there are limitations to forecast the accurate valuation for the long-term period by both BPN and RNN. This study thus uses a recurrent fuzzy neuron network (RFNN) model in addition to BPN and ERNN. According to Lee and Teng [6], RFNN has several key aspects: dynamic mapping capability, temporal information storage, universal approximation, and the fuzzy inference system. Li et al. [34] also argued that RFNN has the same dynamic and robust advantage as RNN. In addition, the network function can be interpreted using fuzzy inference mechanism. Therefore, long-term prediction fuzzy models can be easily implemented using RFNNs. The network output in RFNN is fed back to the network input using one or more time delay units. As depicted in Figure 3, the general microstructure of RFNN consists of four layers in general: an input layer, a membership layer, a fuzzy rule layer, and an output layer.

The information transmission process and basic functions of each layer are as follows.

- (1) Input layer: the input nodes in this layer represent input variables. The input layer only transmits the input value to the next layer directly and no computation is

conducted in this layer. From (2.5), the connection weight at the input layer ( $w_i^1$ ) is unity:

$$g_i^{(1)} = x_i^{(1)}. \quad (2.7)$$

- (2) Membership layer: the membership layer is also known as a fuzzification layer and contains several different types of neurons, each neuron performs membership function. The membership nodes in this layer correspond to the linguistic label of the input variables in the input layer and serve as a unit of memory. Each of these variables is transformed into several fuzzy sets in the membership layer where each neuron corresponds to a particular fuzzy set, with the actual membership function being provided by the neuron output. Each neuron in this layer represents characteristics of each membership function, and Gaussian function serves as the membership function. The  $j$ th neuron in this layer has the following input and output:

$$g_{i(t)}^{(2)} = \exp \left[ -\frac{[x_{ij}^{(2)} - m_{ij}]^2}{\sigma_{ij}^2} \right], \quad (2.8)$$

where

$$x_{ij}^{(2)}(t) = g_i^{(1)}(t) + g_{ij}^{(2)}(t-1)\theta_{ij}, \quad (2.9)$$

$m_{ij}$  denotes the mean value of a Gaussian membership function of the  $j$ th term with respect to the  $i$ th input variable,  $\sigma_{ij}$  represents the standard derivation of the Gaussian type membership function of the  $j$ th term with respect to the  $i$ th input,  $x_{ij}^{(2)}(t)$  is the input of this layer at the discrete time  $t$ ,  $g_{ij}^{(2)}(t-1)$  denotes the feedback unit of memory which stores the past network information and represents the main difference between FNN and RFNN, and  $\theta_{ij}$  indicates the connection weight of the feedback unit.

Each node in the membership layer possesses three adjustable parameters:  $m_{ij}$ ,  $\sigma_{ij}$ , and  $\theta_{ij}$ .

- (3) Fuzzy rule layer: The fuzzy rule layer comprises numerous nodes, each node corresponds to a fuzzy operating region of the process being modeled. This layer constructs the entire fuzzy rule data set. The nodes in this layer equal the number of fuzzy sets corresponding to each external linguistic input variable and receive the one-dimensional membership degree of the associated rule from the nodes of a set in the membership layer. The output of each neuron in the fuzzy rule layer is obtained by using a multiplication operation. The input and output for the  $k$ th neuron in the fuzzy rule layer are as follows:

$$g_i^{(3)} = \Pi_i x_i^{(3)} = \exp \left\{ -D_i (x_i^{(2)} - m_i)^T [D_i (x_i^{(2)} - m_i)] \right\}, \quad (2.10)$$

where  $D_i = \text{diag}[1/\sigma_{1i}, 1/\sigma_{2i}, \dots, 1/\sigma_{ni}]$ ,  $X_i = [x_{1i}, x_{2i}, \dots, x_{ni}]^T$ ,  $m_i = [m_{1i}, m_{2i}, \dots, m_{ni}]$ ,  $x_j^{(3)}$  is the  $i$ th input value that inputs to the neuron of the fuzzy rule layer, and  $g_i^{(3)}$  denotes the output of a fuzzy rule node representing the “firing strength” of its corresponding rule. Links before and fuzzy rule layer indicate the preconditions of the rules, and links after and fuzzy rule layer demonstrate the consequences of the fuzzy rule nodes.

- (4) Output layer: the output layer performs the defuzzification operation. Nodes in this layer are called output linguistic nodes, where each node is for an individual output of the system. The links between the fuzzy rule layer and the output layer are connected by the weighting values  $w_{jk}$ .

For the  $k$ th neuron in the output layer,

$$y_k = g_k^{(4)} = \sum_i^m w_{jk} g_i^{(3)}, \quad (2.11)$$

where  $w_{jk}$  is the output action strength of the  $k$ th output associated with the  $j$ th fuzzy rule and serves as the tuning factor of this layer,  $g_k^{(4)}$  is the final inferred result, and  $y_k$  represents the  $k$ th output of the FRNN.

### 3. Valuation Performance

This work uses the mean square error (MSE) method to assess the performance of three neural networks. The MSE is calculated as the average of the sum of the square of the error, which is given by the difference between the actual and the designed output. MSE thus is computed as

$$\text{MSE} = \frac{1}{T - (T_1 - 1)} \sum_{t=T_1}^T (r_t - \hat{r}_t)^2, \quad (3.1)$$

where  $T$  indicates the total number of samples,  $T_1$  refers to the number of estimated samples,  $r_t$  represents the actual output, and  $\hat{r}_t$  denotes the desired output.

### 4. Computer Simulations

#### 4.1. Data Description

This study focuses on energy futures for the near-month. Daily oil prices for Brent, WTI, DUBAI, and IPE are used in this investigation. The data sources are obtained from the Energy Bureau of USA and International Petroleum Exchange (IPT) of the Great Britain. This work explores the influence of training times on prediction performance so that it classifies the training period from January 1, 1990 to April 30, 2005 into three five-year sections. Table 1 shows the different training periods.

**Table 1:** The periods of the training patterns.

Data set	Training period	Training times (sec.)
I	2000.01.01–2004.12.31.	1244
II	1995.01.01–2004.12.31.	2499
III	1990.01.01–2004.12.31.	3758

**Table 2:** The training times of three parts of various ANNs.

Data set	Backpropagation	Elman RNN	RNN with fuzzy	Times for each ANN
I	BPN1	ERNN1	RFNN1	1244
II	BPN2	ERNN2	RFNN2	2499
III	BPN3	ERNN3	RFNN3	3758

#### 4.2. Comparison among Various Neural Networks

This work divides the training period into three parts and uses Matlab software to perform training and testing. In order to compare the predictive power of these three artificial neural network (ANNs), the training function, namely, Levenberg Marquardt method, is employed, and the number of iterations over the data set is arbitrarily set to 1000 in order to train individual neural networks.

Table 2 shows the symbolization and training times of multi-layer perceptron (MLP), Elman recurrent neural networks (ERNNs) and recurrent fuzzy neural networks (RFNNs).

##### (1) The Comparison of Learning Performance

Following 1000 training times, Table 3 illustrates the following ranking for the learning performance of the three ANNs: RFNN ranks first, followed by ERNN, and finally BPN. Table 3 shows that the learning performance of the ANNs improves with increasing training time of ANNs. One exceptional case is the MSE at part 2 under RFNN, which is less than that obtained from part 3.

##### (2) The Comparison of Predictive Power

The empirical results indicate that the predictive power of the three ANNs is ranked as follows: RFNN ranks first, followed by ERNN, and finally MLP. Table 4 shows that, under ERNNs and RFNNs, the predictive power of the ANNs improves with increasing training time. However, the predictive power of MLP differs from those of ERNNs and RFNNs as the predictive power of the MLP retrogresses with increasing training time. One possible explanation for this phenomenon is that a large difference exists between the forecasting value and the actual value from March 20, 2005 to March 28, 2005. The MLP is not a dynamic network and it cannot be applied by the past experience to the behavioral forecasting.

### 5. Conclusion

This study uses multi-layer perception (MLP), Elman recurrent neural networks (ERNNs) and recurrent fuzzy neural networks (RFNNs) to forecast the crude oil prices and compare

**Table 3:** The comparison of the learning ability (MSE) for the three neural networks.

ANNs	Data set			MSE
	I	II	III	
BPN	$6.84589E - 05$	$4.13072E - 05$	$3.75245E - 05$	$4.90969E - 05$
ERNN	$6.53719E - 05$	$3.71397E - 05$	$3.42490E - 05$	$4.355769E - 05$
RFNN	$2.43727E - 05$	$1.66990E - 05$	$1.78329E - 05$	$1.96349E - 05$

**Table 4:** The comparison of the predictive power of the three neural networks.

ANNs	Part 1	Part 2	Part 3	MSE average value
MLP	$2.6652E - 03$	$2.6653E - 03$	$2.6767E - 03$	$2.6691E - 03$
ERNN	$2.6730E - 03$	$2.6558E - 03$	$2.6449E - 03$	$2.6579E - 03$
RFNN	$2.6085E - 03$	$2.6074E - 03$	$2.6045E - 03$	$2.6068E - 03$

the predictive power of the above three neural network models. Results of this work are summarized as follows.

All of the MSE values obtained under different training times through MLP, ERNNs, and RFNNs are below 0.0026768, suggesting that the use of the neural networks to forecast the crude oil futures prices is appropriate, and consistent learning ability can be obtained by using different training times. This investigation confirms that, under most circumstances, the more training times the neural networks take, the more the learning performance of the neural networks improves. The only exceptional case occurs at part 2 under the RFNN model, where MSE is slightly less than that obtained from part 3.

Regarding the predictive power of the three neural networks, this study finds that RFNN has the best predictive power and MLP has the least predictive power among the three neural networks. This work also finds that, under ERNNs and RFNNs, the predictive power improves when increasing the training time. However, the results are different from those obtained under MLP, indicating that the predictive power improves when decreasing the training time. Possible explanation for this phenomenon is the existence of a large difference between the predictive value and the actual value during a 9-day period. To summarize, this study concludes that the recurrent fuzzy neural network is the best among the three neural networks.

## Acknowledgments

The authors would like to thank Dr. Oleg Smirnov for his insightful comments at the 81st WEA Annual Conference on July 1, 2006. Also, the authors would like to thank the anonymous referees for their valuable comments. This research is partially supported by the National Science Council of Taiwan under grant NSC 99-2410-H-033-026-MY3.

## References

- [1] M. S. Kumar, "The forecasting accuracy of crude oil futures prices," *IMF Staff Papers*, vol. 39, no. 2, pp. 432–461, 1992.
- [2] D. T. Pham and X. Liu, *Neural Networks for Identification Prediction and Control*, Springer, London, UK, 1995.

- [3] A. Nicholas Refenes, A. Zapranis, and G. Francis, "Stock performance modeling using neural networks: a comparative study with regression models," *Neural Networks*, vol. 7, no. 2, pp. 375–388, 1994.
- [4] Chen and Pham, "Time series prediction combining dynamic system, theory and statistical methods," in *Proceedings of the IEEE/IAFE 1995 Computational Intelligence for Financial Engineering*, pp. 151–155, 2001.
- [5] C. W. Omlin, K. K. Thornber, and C. L. Giles, "Fuzzy finite-state automata can be deterministically encoded into recurrent neural networks," *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 1, pp. 76–89, 1998.
- [6] C.-H. Lee and C.-C. Teng, "Identification and control of dynamic systems using recurrent fuzzy neural networks," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 4, pp. 349–366, 2000.
- [7] O. Castillo and P. Melin, "Intelligent system for financial time series prediction combining dynamical systems theory, fractal theory, and statistical methods," in *Proceedings of the IEEE/IAFE Computational Intelligence for Financial Engineering (CIFEr '95)*, pp. 151–155, April 1995.
- [8] C. L. Giles, D. Chen, G. Z. Sun, H. H. Chen, Y. C. Lee, and M. W. Goudreau, "Constructive learning of recurrent neural networks: limitations of recurrent cascade correlation and a simple solution," *IEEE Transactions on Neural Networks*, vol. 6, no. 4, pp. 829–836, 1995.
- [9] R. G. Donaldson and M. Kamstra, "Forecast combining with neural networks," *Journal of Forecasting*, vol. 15, no. 1, pp. 49–61, 1996.
- [10] J. Sharma, R. Kamath, and S. Tuluca, "Determinants of corporate borrowing: an application of a neural network approach," *American Business Review*, vol. 21, no. 2, pp. 63–73, 2003.
- [11] I. Kamstra and M. S. Boyd, "Forecasting futures trading volume using neural networks," *The Journal of Futures Markets*, vol. 15, no. 8, pp. 935–970, 1995.
- [12] C. F. Juang and C. T. Lin, "A recurrent self-organizing neural fuzzy inference network," *IEEE Transactions on Neural Networks*, vol. 10, no. 4, pp. 828–845, 1999.
- [13] A. Nurnberger, A. Radetzky, and R. Kruse, "Determination of elastodynamic model parameters using a recurrent neuro-fuzzy system," in *Proceedings of the 7th European Congress on Intelligent Techniques and Soft Computing*. Verlag Mainz, Aachen, Germany, 1999.
- [14] A. Nürnberger, A. Radetzky, and R. Kruse, "Using recurrent neuro-fuzzy techniques for the identification and simulation of dynamic systems," *Neurocomputing*, vol. 36, pp. 123–147, 2001.
- [15] J. Zhang and A. J Morris, "Recurrent neuro-fuzzy networks for nonlinear process modeling," *IEEE Transactions on Neural Networks*, vol. 10, no. 2, pp. 313–326, 1999.
- [16] P. Mastorocostas and J. Theocharis, "FUNCOM: a constrained learning algorithm for fuzzy neural networks," *Fuzzy Sets and Systems*, vol. 112, no. 1, pp. 1–26, 2000.
- [17] P. A. Mastorocostas and J. B. Theocharis, "A recurrent fuzzy-neural model for dynamic system identification," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 32, no. 2, pp. 176–190, 2002.
- [18] C. F. Juang, "A TSK-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms," *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 2, pp. 155–170, 2002.
- [19] W. Yu and A. Ferreyra, "System identification with state-space recurrent fuzzy neural networks," in *Proceedings of the 43rd IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, December 2004.
- [20] Y.-Y. Hong and C.-F. Lee, "A neuro-fuzzy price forecasting approach in deregulated electricity markets," *Electric Power Systems Research*, vol. 73, no. 2, pp. 151–157, 2005.
- [21] C. J. Lin and C. H. Chen, "Identification and prediction using recurrent compensatory neuro-fuzzy systems," *Fuzzy Sets and Systems*, vol. 150, no. 2, pp. 307–330, 2005.
- [22] D. J. Hirshfeld, "A fundamental overview of the energy futures market," *The Journal of Futures Market*, vol. 3, no. 1, pp. 75–100, 1983.
- [23] C. W. Ma, "Forecasting efficiency of energy futures prices," *The Journal of Futures Markets*, vol. 9, pp. 373–419, 1989.
- [24] A. Serletis, "Rational expectations, risk and efficiency in energy futures markets," *Energy Economics*, vol. 13, no. 2, pp. 111–115, 1991.
- [25] R. S. Pindyck, "Long-run evolution of energy prices," *Energy Journal*, vol. 20, no. 2, pp. 1–24, 1999.
- [26] B. Adrangi, A. Chatrath, K. K. Dhanda, and K. Raffiee, "Chaos in oil prices? Evidence from futures markets," *Energy Economics*, vol. 23, no. 4, pp. 405–425, 2001.
- [27] T. Krehbiel and L. C. Adkins, "Price risk in the NYMEX energy complex: an extreme value approach," *Journal of Futures Markets*, vol. 25, no. 4, pp. 309–337, 2005.
- [28] J. J. Hopefield, "Networks and physical systems with emergent collective," *Proceedings of National Academy Science*, vol. 79, pp. 2554–2558, 1982.

- [29] S. Haykin, *Neural Networks: A Comprehensive Fundation*, Prentice Hall, Upper Saddle River, NJ, USA, 2nd edition, 1999.
- [30] J. A. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation*, Addison-Wesley Longman, Boston, Mass, USA, 1991.
- [31] J. L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, no. 2, pp. 179–211, 1990.
- [32] C.-T. Lin and C. S. G. Lee, *Neural Fuzzy System: A Neuro-Fuzzy Synergism to Intelligent Systems*, Prentice Hall, Upper Saddle River, NJ, USA, 1996.
- [33] B. Hammer and J. K. Nørskov, "Theoretical surface science and catalysis-calculations and concepts," *Advances in Catalysis*, vol. 45, pp. 71–129, 2000.
- [34] X. Li, Z. Chen, Z. Yuan, and G. Chen, "Generating chaos by an Elman network," *IEEE Transactions on Circuits and Systems I*, vol. 48, no. 9, pp. 1126–1131, 2001.
- [35] L. Zadeh, "Fuzzy Sets Yulia," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.
- [36] L. Zadeh, "Fuzzy sets and systems," in *Proceedings of the Symposium on Systems Theory*, Polytechnic Institute of Brooklyn, April 1965.
- [37] H.-J. Zimmermann, *Fuzzy Set Theory and Its Applications*, Kluwer Academic Publishers, Boston, Mass, USA, 1985.
- [38] J. J. Buckley and Y. Hayashi, "Fuzzy neural networks: a survey," *Fuzzy Sets and Systems*, vol. 66, no. 1, pp. 1–13, 1994.
- [39] J. W.-S. Hu and C.-H. Chang, "Euro/USD currency option valuation combining two artificial neural network models and various volatilities," in *Proceedings of the 80th Western Economic Association Annual Conference (WEA '05)*, San Francisco, Calif, USA, July 2005.

## Research Article

# A Novel Method for Technology Forecasting and Developing R&D Strategy of Building Integrated Photovoltaic Technology Industry

**Yu-Jing Chiu<sup>1</sup> and Tao-Ming Ying<sup>2</sup>**

<sup>1</sup> Department of Business Administration, Chung Yuan Christian University, 200 Chung Pei Road, Chung Li 32023, Taiwan

<sup>2</sup> Department of Construction Engineering, National Taiwan University of Science and Technology, No. 43, Section 4, Keelung Road, Da'an District, Taipei City 106, Taiwan

Correspondence should be addressed to Yu-Jing Chiu, [yujing@cycu.edu.tw](mailto:yujing@cycu.edu.tw)

Received 24 February 2012; Revised 17 May 2012; Accepted 22 May 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Y.-J. Chiu and T.-M. Ying. 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.

Because of global warming, renewable energy technologies have become more essential currently, with solar energy technology advancing worldwide. Therefore, interdisciplinary integration is an important trend, and building-integrated photovoltaic (BIPV) is an emerging technology involving the photovoltaic and building fields. The purpose of this study is to understand the technology evolution of BIPV and to determine the R&D planning direction. This paper proposes a hybrid approach to explore the life cycle of BIPV technology and develop the R&D strategy of related industries. The proposed approach comprises the following submodules. First, patent analysis is employed to transform patent documents into structured data. Second, the logistic growth model is used to explore the life cycle of BIPV technology. Third, a patent matrix map analysis is used to develop the R&D strategy of the BIPV industry. Through the analysis by the logistic model, the BIPV technology is transformed from the emerging stage to the growth stage of a long-term life cycle. The other important result is created by the three-dimensional matrix for R&D strategies in this paper.

## 1. Introduction

Issues of energy conservation and prevention of global warming have been of great concern in recent years. For energy shortages, the development of renewable energy technologies is an inevitable trend and includes solar, wind, hydro- and bio-mass energy. There are advantages to solar energy, including that it is natural, nonpolluting, inexhaustible, noiseless, and not subject to geographical constraints. Solar energy contains few moving parts, and therefore, needs almost no maintenance. The cost of solar energy is fixed at the initial investment stage.

It is the parity technology in the long-term operation. Solar energy technology can be divided into the ground-based system, rooftop system, and building-integrated photovoltaic (BIPV) system. Since the ground-based system needs considerable land and the rooftop system is restricted to a limited number of building roofs, this study focuses on the BIPV system, which is used to construct three-dimensional space to create the largest possible photovoltaic power generation with less area. In addition, solar energy technology can be embedded in various ways of designed buildings, which not only has the ability to improve the use of renewable energy but is also by far the oldest and the most cleanly green energy worldwide. BIPV will gain increasing attention of the building energy efficiency market in the twenty-first century.

Consequently, the ability to manage intellectual property plays a critical role in the R&D process or new product development. The report of the World Intellectual Property Organization (WIPO) shows that 90%–95% of the world's inventions can be found in patent documents, and 80% of these technologies do not appear in other resources, such as journals, magazines, and encyclopedias [1]. Moreover, a WIPO investigation showed that a company can save up to 60% of time in R&D and 40% of research budget if it can use patent documents efficiently. Patents contain substantial information including practical and specific technological content, technical reporting, technological trends of firms, and technological development trends [2]. Patent information is a sufficient source of technical knowledge, and the patent contains characteristics of the technology and market because it fulfills a clear standard and originality, technical feasibility, and commercial value [3]. It can also be the basis of technical competitive analysis and technology trends analysis [4], and it can be regarded as an important indicator of technological changes [5]. Therefore, to master the technology development trend and competitive advantage of enterprises, we must rely on detailed patent intelligence to fully achieve maximum value from patent analysis [6].

The analysis of patent databases can provide the latest information technology and product. The growth of the number of patents may represent a national or industrial development of technology. It is a leading indicator of competitiveness, allowing one to grasp the message of patent data and analysis of industry trends to trace an evasive technology or technological innovation development strategy.

Many experts and scholars used patent analysis to investigate different levels as follows:

- (1) discussion of national economic growth or innovation activities: Barroso et al. [7] studied 21 Latin American countries, including analysis of patents as a source of technical information to improve scientific and technological research and development and innovation in their own countries; Abraham and Moitra [8] used patent analysis to accurately assess the technical progress and technical innovation;
- (2) discussion of enterprise technology ability: Liu and Shyu [1] used patent analysis of Taiwan LED and TFT industry technology development, strategy, and technology planning and forecasting. Ernst [9] analyzed the German mechanical engineering industry and four patent strategies of performance associated with the company;
- (3) discussion of distribution of industries or enterprises: Guo [10] analyzed Taiwan national patent database (TWPAT) for the number of applications and development trend. Chen and Cheng [11] analyzed the US patent database (USPTO) in patent classification and UPC solar patent analysis research;
- (4) an analysis of patent citation data: Hu [12] studied the USPTO citation data seeking knowledge of the East Asian regional traffic. Tijssen [13] related to patent

citation analysis of global and domestic industry of interaction between science and technology and knowledge flows.

Technology forecasting is the theory and practice of technology development strategy. Many researchers have combined patent analysis with different technology prediction methods to identify future development trends and strategies. Trappey et al. [14] constructed the multitechnology cluster to analyze the RFID (radio frequency identification) technology forecasting model. They discussed future development strategies of industry by analyzing RFID patent clustering, which has reached the saturation stage of the technology life cycle. This model provided systematic quantitative analysis of technology development and supported expert intuition and judgment [15]. Chen et al. [16] considered the time dimension of technology forecasting. He stated that the *S*-curve was an effective quantitative technology forecasting model for cumulatively analyzing patents. However, researcher ability limited patent data analysis, and the accuracy and reliability of the data also affected analysis accuracy. Therefore, Cheng et al. [17] used the FAHP (fuzzy analytic hierarchy process) method as a technology forecasting method to evaluate and select novel material fields in a rapidly changing environment.

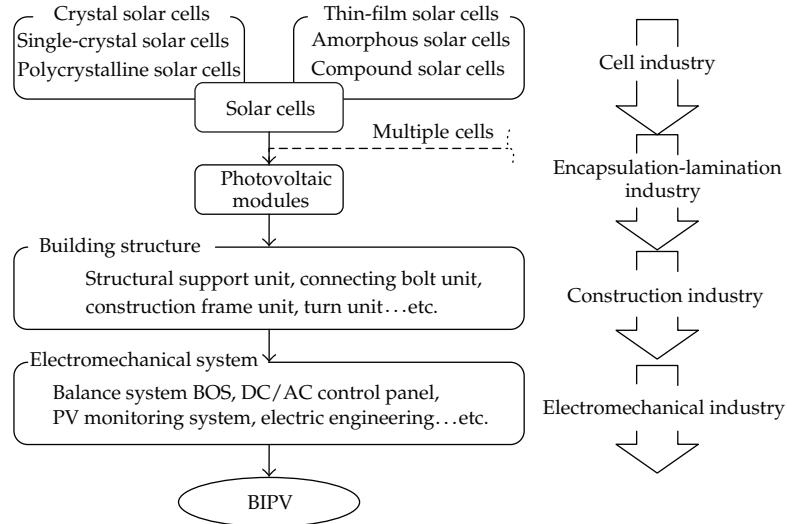
However, it is more important for an R&D manager to know how to master the market of technology imports and the exit market of time. If an enterprise wants to keep a competitive advantage, it must establish a technology forecasting system [18]. According to technology life cycle displayed by *S*-curve, we can effectively understand the technology development track.

Technology forecast is through text or digital form to describe future machinery, entity program, and the application of program and development [19]. Technology forecasters must focus on changes of technology function, the significance of innovation, and the implementation of time points. Technology forecasting of content contains technology ability of growth, replacement rate of old and new technology, situation of technology diffusion, market penetration, and future possibilities [20]. There are three types of technological prediction methods. They are direct forecasting, correlative forecasting, and structural forecasting. However, as the technology industry competition intensifies and technology development continues in the globalized world, technology forecasting must master the principles of accurate, fast, and simple to achieve the desired results. Therefore, we use the direct forecasting method by reliable patent data in this study.

We are interested in the BIPV's development and aim to explore its evolution. This study examines the viewpoint of patent analysis and hopes to offer some useful recommendations for BIPV's technology development. Thereby, we use a logistic model to explore the life cycle of BIPV technology development. The patent database of the United States Patent and Trademark Office (USPTO) is by patent number. We combine logistic growth model analysis of embryonic development of BIPV technology trends and forecasts to allow industry R&D managers to develop technology strategies for future planning.

## 2. BIPV Technology and Industry

All energy consumption in a building includes raw-material production, construction process, decoration, and the final usage stage. Through the combination of solar power technology and building itself, the building can function as a power generator. Here, photovoltaic modules are fabricated as building material and facilitate the building to convert



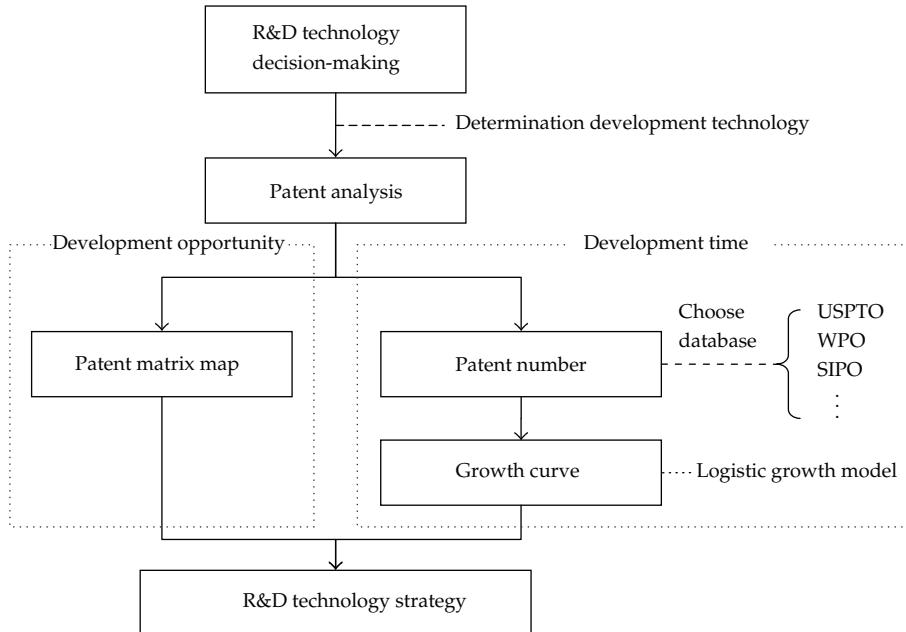
**Figure 1:** BIPV technology industry chain.

solar energy into electricity. Therefore, BIPV can be called a static generator as well as the new generation of photovoltaic building [21].

BIPV is a new concept of converting solar energy into electrical energy. In addition to developing self-efficient zero-energy buildings (ZEB) with the use of BIPV, heat load within the building can be reduced via the thermal resistance of the photovoltaic module and then achieve the dual effect of energy saving and carbon reduction. At the same time, the excess of clean energy can be sold back to the commercial electricity providers to sell to other customers, thus creating a multiple effect of energy saving and carbon reduction. In the future, the integrated distribution of BIPV electricity network will form the solar power grid of a metropolitan area that we might call a photovoltaic city.

The methods of integrating photovoltaic and building can be distinguished into two kinds of technology: installed PV module above a building and PV module integrated with the building. BIPV's technology industry chain can be divided into four flows, including solar cells, photovoltaic modules, building structure, and electromechanical system (Figure 1). The industry chain goes from the most upstream raw material of the silicon, a seed crystal puller rod for growing silicon boule, and then cuts into a silicon wafer, which is the main source of solar cells. After being made into solar cells that can be converted into energy, the solar cells are linked together in modules. Later, these solar modules are combined with a balance system (balance system BOS) into a form such as the converter, current breaker, building support structures, power cables, and monitoring instrumentation and then finally installed.

Figure 1 shows technology and market chain relationships for the BIPV industry. It is possible to clearly understand technology and market relationships in the BIPV industry. For example, PV manufacturers must consider and consistently comply with national building codes when developing products; the construction and electromechanical industries in the wiring and installation of PV modules; electrical and mechanical piping interface design, integration, and coordination when building a high-performance, high-security system; visual appeal. Finally, BIPV building systems are handed to market users after completion and achieve effective energy savings and carbon reduction, furthering the realization of zero-energy buildings.



**Figure 2:** R&D technology strategy model.

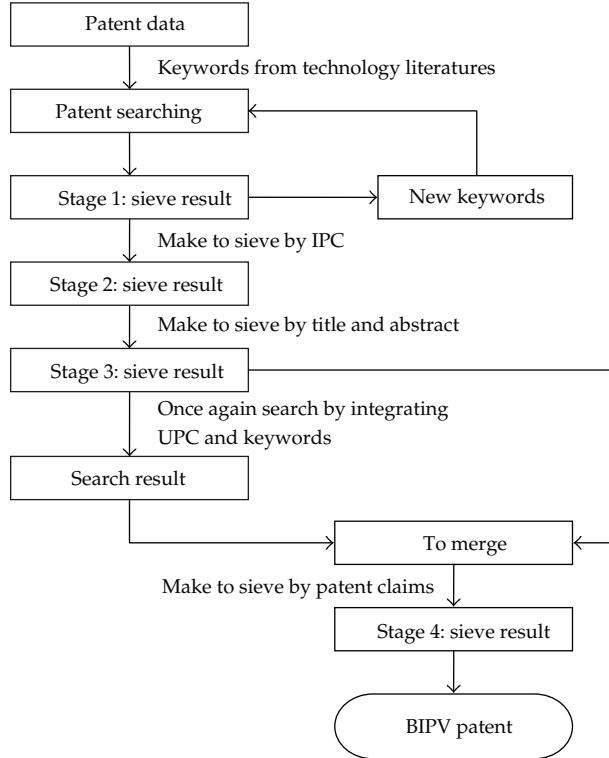
Since BIPV is integrated into the three-dimensional space with the building structure and does not take up any additional ground space, BIPV is the best model of applying renewable energy in the highly urbanized area. However, its installation cost is high, which affects the development of market demand. Therefore, when developed countries are promoting renewable energy, they rely on the government's four policy factors, including incentive policy, subsidies, low installation cost, quick approval process, and lending policies for financial institutions, which sustain the promotion of the BIPV industry and market development. In the future, the key factor in reducing the impact on market development is the continuity of technology development. The installation cost, meaning the economies of scale of the BIPV industry, could flourish unimaginably when the cost of solar photovoltaic electricity and cost of commercial electricity are balanced.

### 3. Methodology

For this study, we intend to explore completed lifecycle evolution of BIPV technology. The quantity and quality of data acquisition are priorities and explicitly defining the industry technology is necessary to avoid forecasting bias. For the above reasons, we adapt the expert opinions and growth curve to develop the forecasting models. Therefore, we combined patent analysis, growth curve, and expert opinions to analyze the development of BIPV technology and construct the R&D strategy. The framework of this study is presented in Figure 2. These are summarized below.

#### 3.1. Patent Searching

Patent searching is based on a specific field of study, through the relevant patent documents issued by relevance of patent data search, application for the analysis, and followup of



**Figure 3:** Process of BIPV patent searching.

research. We must understand its retrieval application purpose, scope, search method, and search tools before patent searching. This research through the patent analysis is to predict technology trends and use the BIPV technologies as scope, and then to group Boolean searches through the USPTO patent database of keywords (Boolean Search), its use of the retrieval tools patented analysis software developed for Lian Ying Technology Patent and Patent Guider 2008 Tech tools for analysis. The patent searching strategies for this study are the main access point to keywords and to supplement, patent classification (international patent classification (IPC)), to achieve the accuracy and integrity of the retrieval processes in Figure 3.

### 3.2. Keywords Extraction

To construct our patent map, keywords are required first. It forms important patent search methods by technical or patent documents for keywords, and keywords in groups to a logical combination of relevance. Therefore, the processes of keywords extraction are including literature review, compile keywords, re-searching and find out new keywords, and establish the keywords pool (Figure 4). After literature review, the three parts keywords of BIPV industry chain can be extracted. The new keywords found in the retrieval process to establish more complete keywords pool of BIPV (Figure 5).

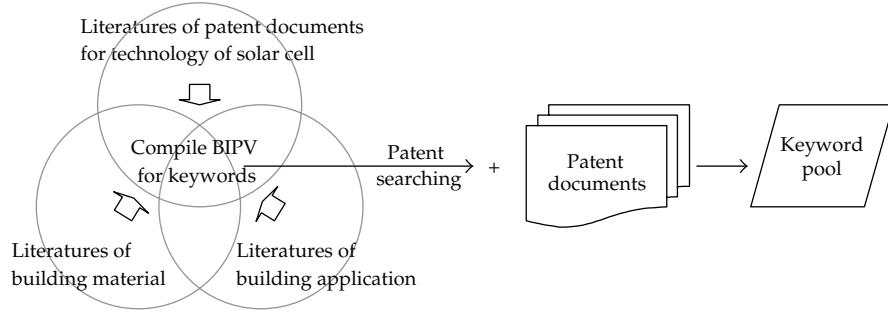


Figure 4: The process of keywords extraction.

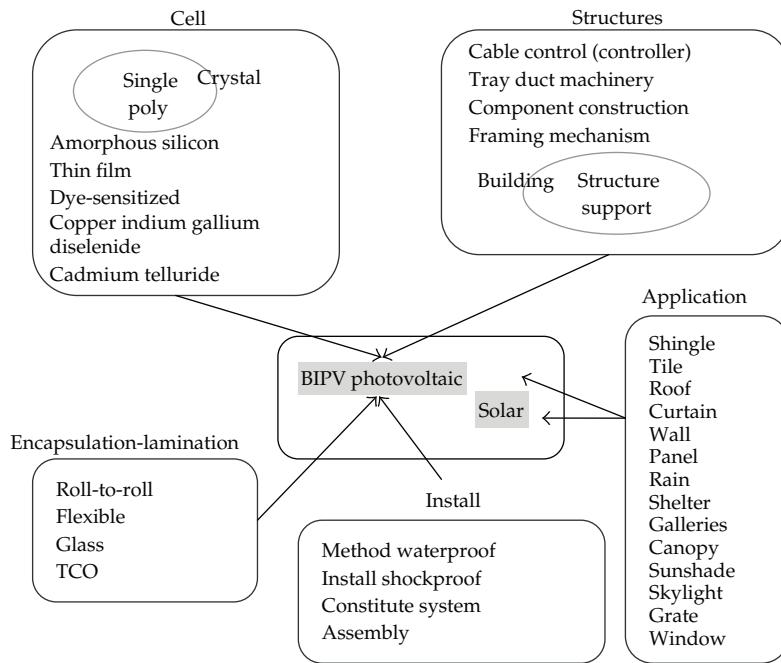


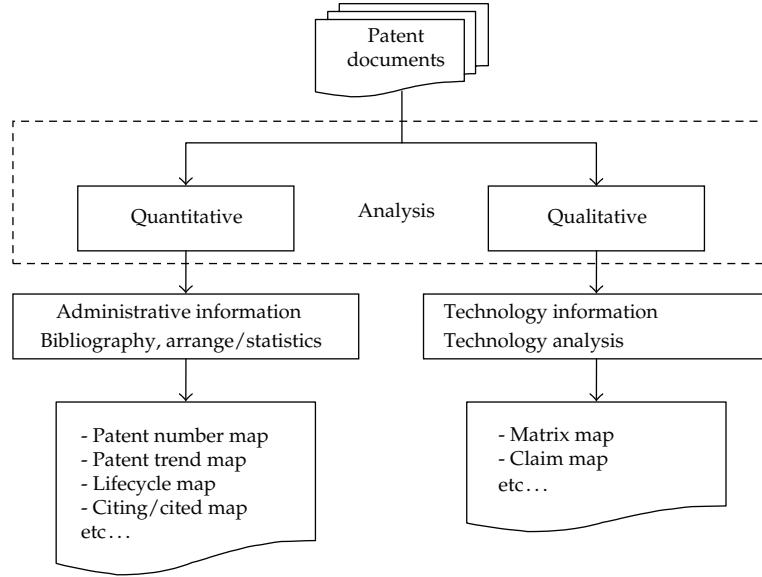
Figure 5: Keywords pool of BIPV.

### 3.3. Patent Map

The main purpose of a patent map is easy and effective understanding of patent information and analysis on a wide range of complex data and to represent them through measures such as tables or schemata. Figure 6 explains the process and related information of the patent map.

### 3.4. Matrix Map

A matrix map is two-dimensional array, and the array matrix is a simple data structure. Two-dimensional array is by  $x, y$  two-axial plane, and we can construct a spatial distribution of the state that through the display of the spatial distribution can clearly see the correlation



**Figure 6:** The process of patent map.

between the  $x$ -axis items with the  $y$ -axis items and the trends. Therefore, we can understand the relationship of its elements and functions in a two-dimensional array matrix notation as shown in Table 1. This study uses this two-dimensional array matrix to represent the technology function matrix, as shown in Table 2. A matrix map shows patent intelligence with the correlation between technical information, including technology sort and purpose sort. The matrix can be used to teach important technology and development trends. Therefore, R&D managers can use patent matrix maps to develop related technologies and preventive strategies.

### 3.5. Logistic Growth Model

The study supposes technology growth with the property of S-curve. We adopt the Loglet Lab 2, which is a kind of analysis software developed by Rockefeller University in 1994. There is a great advantage in setting all the parameters related to time series into analysis and the result presents as an S-curve [22]. S-curve can be classified into two types of symmetry and unsymmetry. The symmetry type represents the growth curve with inflection point, which can be presumed by the following equation:

$$P(t) = \frac{k}{1 + e^{-\alpha(t-\beta)}}, \quad (3.1)$$

where  $P(t)$  means the patent numbers change with the time  $t$ ;  $\alpha$  means the slope (growth rate) of S-curve;  $\beta$  means the inflection point of growth, which is the turning point of time spent into technique;  $k$  means the saturation level of growth;  $e$  means nature log. Based on the practice consideration,  $[k * 10\%, k * 90\%]$  is defined as the growth interval. The period time from 10% of technology with maximum utility value to 90% presents by  $\Delta t$ . To proper

**Table 1:** Arrays of two dimensions.

		Row-Major					
		<i>n</i>					
<i>m</i>		<i>R</i> <sub>1</sub>	<i>R</i> <sub>2</sub>	<i>R</i> <sub>3</sub>	...	...	<i>R</i> <sub><i>n</i></sub>
Column-Major	<i>C</i> <sub>1</sub>	<i>Q</i> [1][1]	<i>Q</i> [1][2]	<i>Q</i> [1][3]	...	...	<i>Q</i> [1][ <i>n</i> ]
	<i>C</i> <sub>2</sub>	<i>Q</i> [2][1]	<i>Q</i> [2][2]	<i>Q</i> [2][3]	...	...	<i>Q</i> [2][ <i>n</i> ]
	<i>C</i> <sub>3</sub>	<i>Q</i> [3][1]	<i>Q</i> [3][2]	<i>Q</i> [3][3]	...	...	<i>Q</i> [3][ <i>n</i> ]
	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	<i>C</i> <sub><i>m</i></sub>	<i>Q</i> [ <i>m</i> ][1]	<i>Q</i> [ <i>m</i> ][2]	<i>Q</i> [ <i>m</i> ][3]	...	...	<i>Q</i> [ <i>m</i> ][ <i>n</i> ]

**Table 2:** Matrix map of patent.

		Purpose sort					
		<i>P</i> <sub>1</sub>	<i>P</i> <sub>2</sub>	<i>P</i> <sub>3</sub>	...	...	<i>P</i> <sub><i>n</i></sub>
Technology sort	<i>T</i> <sub>1</sub>	PDQ [1][1]	PDQ [1][2]	PDQ [1][3]	...	...	PDQ [1][ <i>n</i> ]
	<i>T</i> <sub>2</sub>	PDQ [2][1]	PDQ [2][2]	PDQ [2][3]	...	...	PDQ [2][ <i>n</i> ]
	<i>T</i> <sub>3</sub>	PDQ [3][1]	PDQ [3][2]	PDQ [3][3]	...	...	PDQ [3][ <i>n</i> ]
	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	<i>T</i> <sub><i>m</i></sub>	PDQ [ <i>m</i> ][1]	PDQ [ <i>m</i> ][2]	PDQ [ <i>m</i> ][3]	...	...	PDQ [ <i>m</i> ][ <i>n</i> ]

Notes: PDQ (patent document quantity).

represent the growth process and make it with more fitness,  $\alpha$  is replaced by (3.2) in practice [23]. Therefore, the  $P(t)$  is proposed as (3.3) as follows:

$$\Delta t = \frac{\ln 81}{\alpha}, \quad (3.2)$$

$$P(t) = \frac{k}{1 + e^{-(\ln 81/\Delta t) - (t-t_m)}}. \quad (3.3)$$

This research is through the BIPV technology growth model calculation and describe technical development trend.

## 4. BIPV Technology Development

This model of research in technology forecasting and strategy for research and development is under construction by the world's major advanced countries such as the United States, Japan, and Germany, plus positive development of a PV industry in China. The BIPV technology and applications are as the empirical techniques in the study.

### 4.1. Patent Map Analysis

Using the keyword pool to begin the patent searching based on USPTO database, the date from the BIPV technology patents in the field of data begins on July 26, 1977 and runs to October 31, 2011. Through the patent searching process, results of 263 valid patents in this study are obtained. The results of patent numbers in every year and accumulative patent

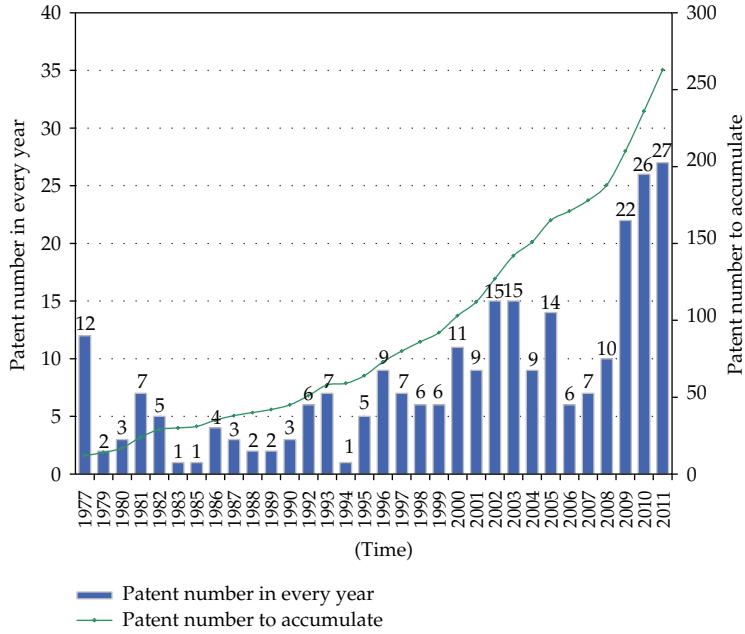


Figure 7: Patent number and accumulative numbers of BIPV.

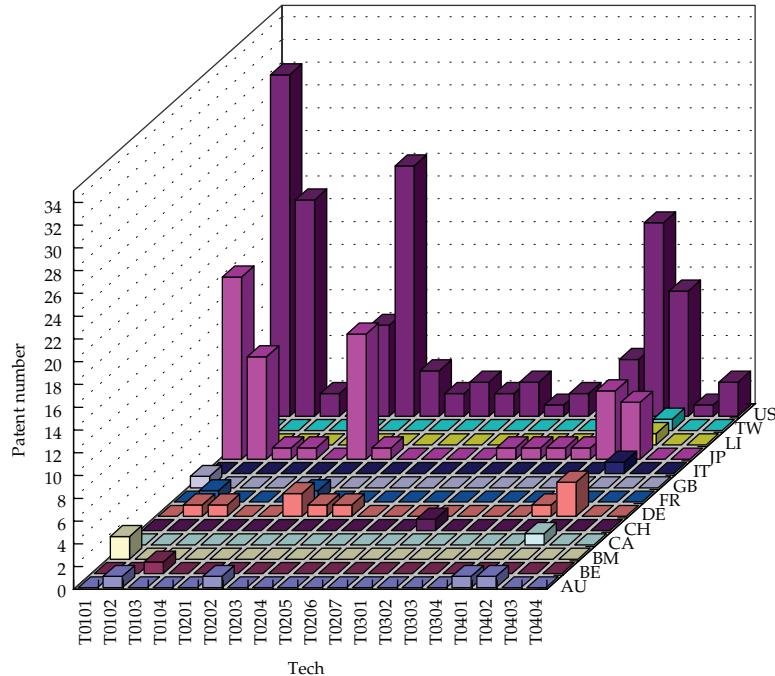
numbers are shown in Figure 7. Global warming and near-future energy crisis results show an increase in related patents since early 2000.

Analyzing the number of patents for technology, purpose, and application (Figure 8) can determine the BIPV technology development level of a country. The patent output of United States (US) is the first; Japan (JP) is the second. Germany (DE) accounted for approximately 5% of BIPV technology in Europe. Results show that countries most focused on R&D of BIPV technology are the United States, Japan, and Germany.

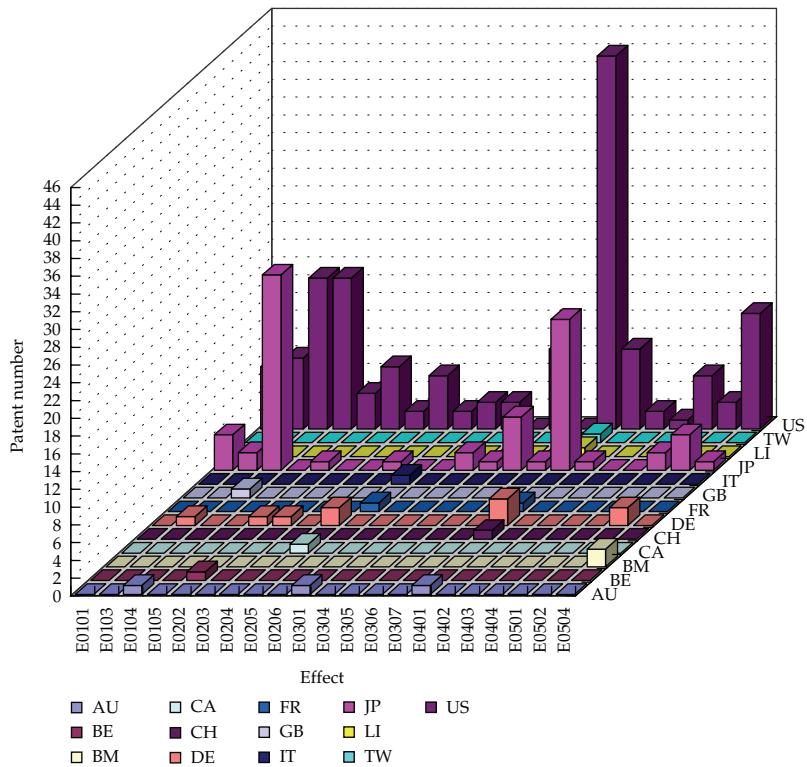
Results of the firm-level analysis show major manufacturers and product applications. Table 3 shows PV module technology directions and strategies in the BIPV industry. The table shows that most manufacturers continue developing roof technology and its applications. However, some manufacturers have changed R&D strategies to focus on glass curtain walls and external walls.

#### 4.2. Matrix Map

The results of technology effect analysis, the matrix map is shown in Figure 9. According to the complex BIPV technology, we can classify the functions into two groups: main function and subfunction. The main functions of BIPV are generating electric, building materials to diversify quality and reliability, install and maintain, and keeping cost down. The main function map presents the development of BIPV from 1977 to 2011. A larger pie chart means a large number of patents and represents the technology development center. On the contrary, a smaller pie chart means fewer patents, which is worthy of note by research and development personnel. BIPV derives technologies corresponding to the early functions in the field of

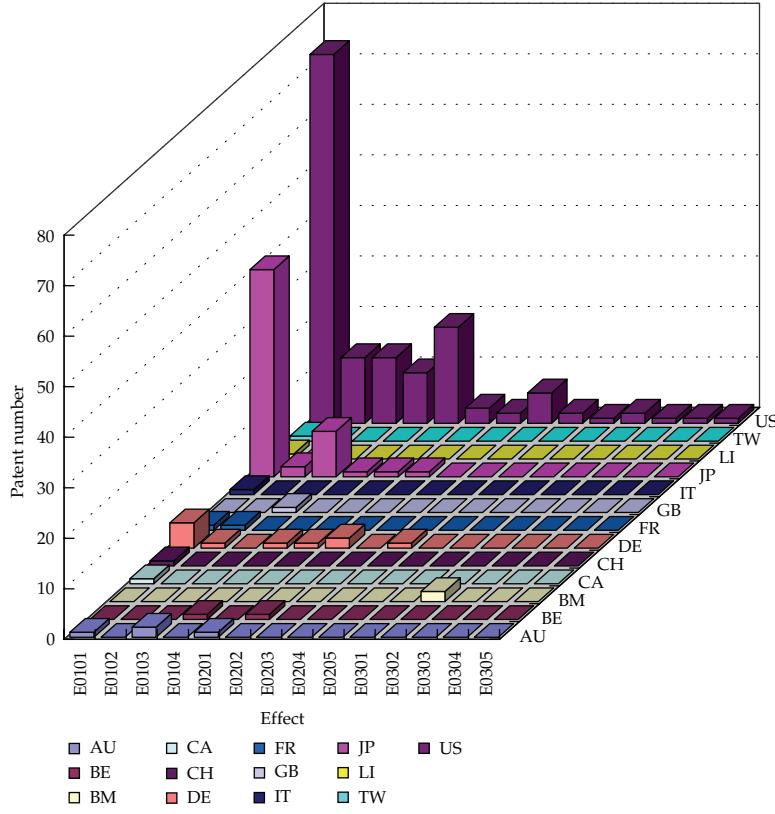


(a) The number of patents in the focus on technology



(b) The number of patents in the focus on the purpose

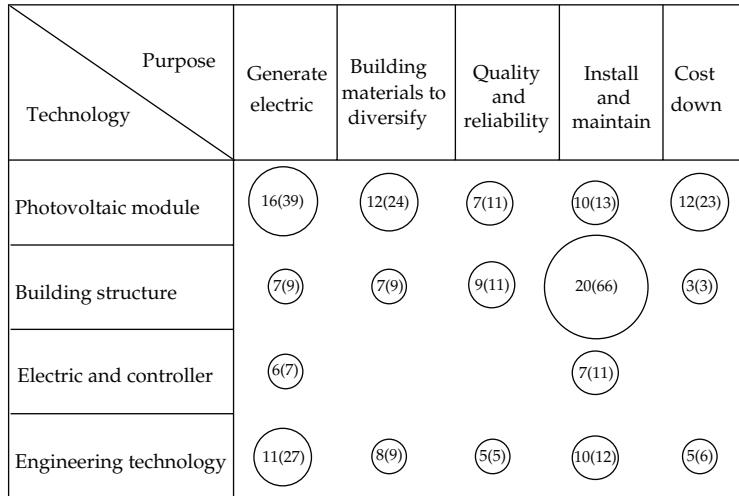
**Figure 8:** Continued.



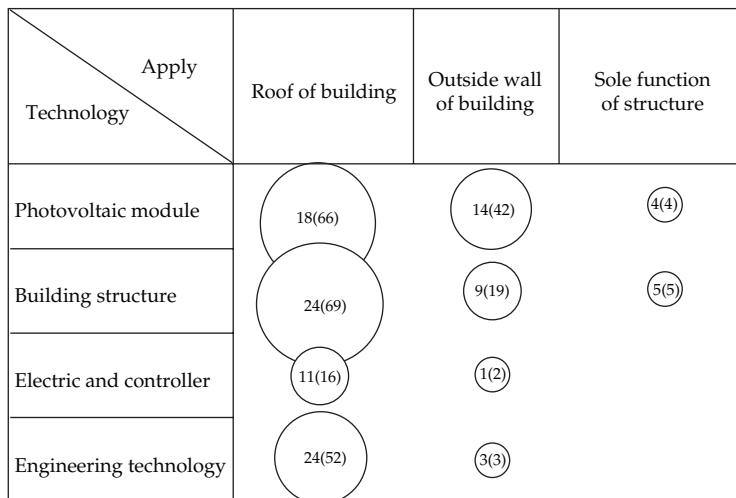
(c) The number of patents in the focus on the application

**Figure 8:** BIPV technology three-dimensional (technology, purpose, and application) map in country level.**Table 3:** BIPV manufacturers and product applications.

Manufacturer	Country	Product name	Application
Uni-solar	US	Photovoltaic Laminate	Various shapes of roofs in commercial buildings
		Power Tilt	The flat roofs in commercial buildings
		EnerGen	Residential roofs
		PowerShingle	Residential roofs
Power Film Solar	US	N/A	Commercial building roof
First Solar	US	N/A	Commercial building roof
Ascent Solar	US	WaveSol	Roof
Global Solar Energy	US	Power FLEX	Roof
Solyndra	US	100/200 series	Commercial building roof
Konarka	US	Power Plastic	Roofs and external walls
Sharp	JP	ND/NU series	Roof
		NA series	Roofs and external walls
Kaneka Solartech	JP	Hybrid silicon	Roof
		SOLTILEX, VISOLA, GRANSOLA	Tile style roof in a residential
		See-Through	Glass curtain wall
		HEM series	Roof
Würth Solar	DE	GeneCIS series	Roofs and external walls
Dyesol	AU	Sun2	Roofs and external walls



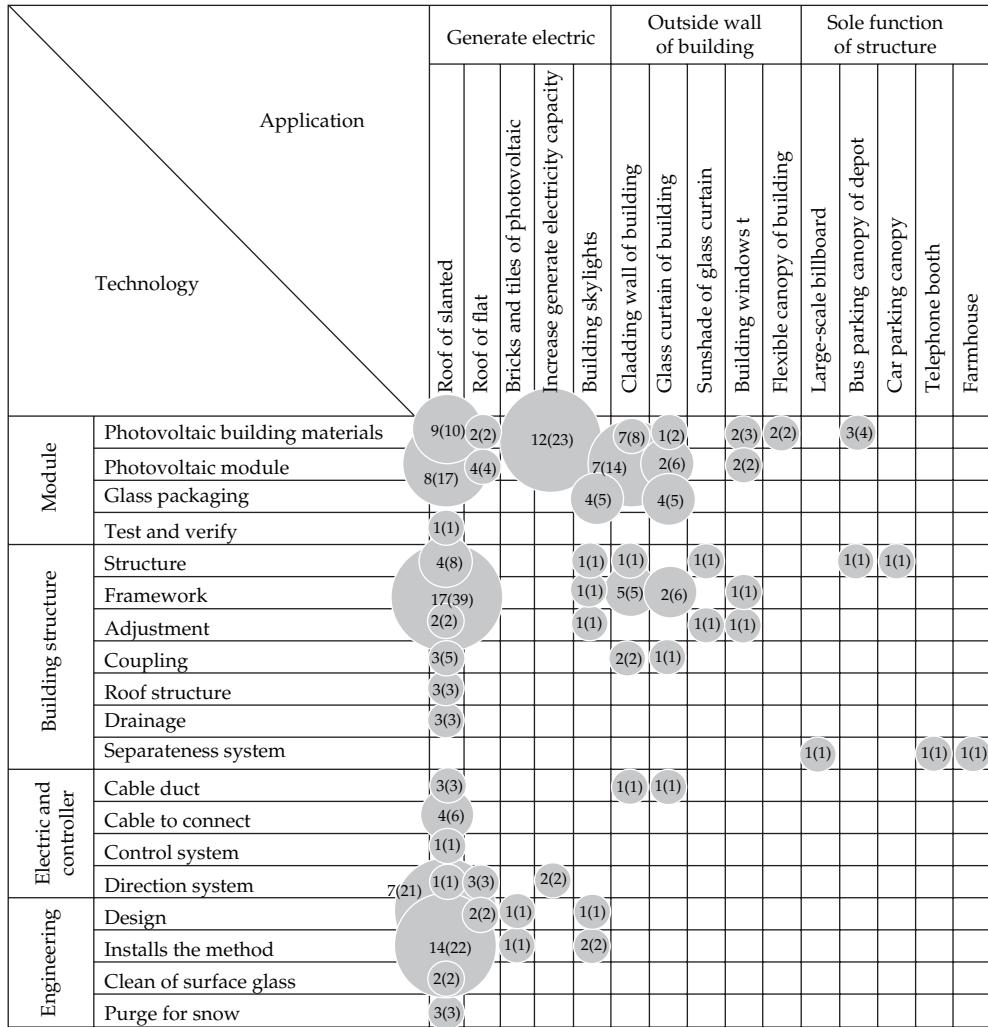
**Figure 9:** Main purpose matrix map of BIPV.  $Y(Q)$ : patent number of years (in accordance with this technology and the effectiveness of patent). The pie chart represents the number of patents. Patent number of the more means larger the pie chart, and vice versa smaller.



**Figure 10:** Main apply matrix map of BIPV.  $Y(Q)$ : patent number of years (in accordance with this technology and the effectiveness of patent). The pie chart represents the number of patents. Patent number of the more means larger the pie chart, and vice versa smaller.

technology development from 1977 to 2011. The technical function matrix gives a more precise formulation of the future direction of development policy.

We can understand the technology development trends by analyzing the effect of the technology and the application of BIPV. After further study through a combination of two oriented analyses, we will be better able to judge technology trends and liquidity phenomena. According to technology mapping from 1977 to 2011, the main applications of BIPV are in the roof of the building (Figure 10), especially in USPTO patent pieces in the United States, Japan, and Germany.

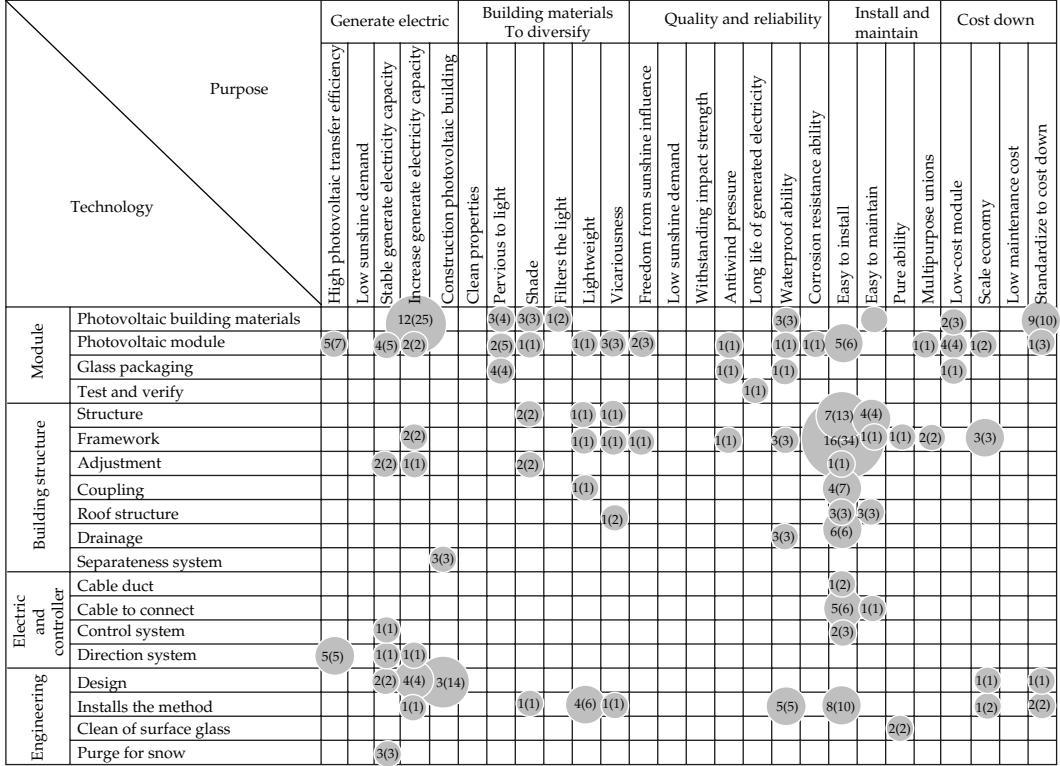


**Figure 11:** Subapplication matrix map of BIPV. Y(Q): patent number of years (in accordance with this technology and the effectiveness of patent). The pie chart represents the number of patents. Patent number of the more means larger the pie chart, and vice versa smaller.

Due to high latitudes, the technology development is mainly to oblique roof. We also can find out technology application and purpose in the field of BIPV technology development by Figures 11 and 12. The subtechnology matrix can provide more accurate positioning and development of the strategic direction of future research and development.

We can use dynamic trend analysis, technology research, and development activities to watch the technology development of the whole picture through patent maps. The cross-analysis by purpose and applications of technology can help us to grasp the accurate R&D direction.

After analyzing the technology purpose and application matrix map, it was combined with time to show the evolution of BIPV purposes and applications over time. This study proposes a patent number weight over time concept. In Figure 13, there are five major purposes of BIPV technology on the  $x$ -axis and three major applications of BIPV technology



**Figure 12:** Subpurpose matrix map of BIPV.  $Y(Q)$ : patent number of years (in accordance with this technology and the effectiveness of patent). The pie chart represents the number of patents. Patent number of the more means larger the pie chart, and vice versa smaller.

on the  $y$ -axis. Over three periods ( $s_1 - s_3$ ), the growth in patent number weights of outside wall building materials and the installation and maintenance of solar panels can be identified. The patent number weights include technical application and purpose over three periods. Technical purposes include electricity generation, diversified building materials, quality and reliability, installation and maintenance, and cost reduction. Technical applications are structure function, outside building wall, and building roof. The weights for the technical purpose and application of the patent number are calculated using (4.1):

$$w_{x_i,s_j} = \frac{p_{x_i,s_j}}{\sum_{i=1}^m p_{x_i,s_j}} \times 100\% \quad (4.1)$$

$$w_{y_i,s_j} = \frac{p_{y_i,s_j}}{\sum_{i=1}^n p_{y_i,s_j}} \times 100\%,$$

where  $w_{x_i,s_j}$  is the patent number weight for different technical purposes ( $x_i$ ) in a different stage ( $s_j$ ).  $p_{x_i,s_j}$  is the patent number for different technical purposes ( $x_i$ ) in a different stage ( $s_j$ ).  $w_{y_i,s_j}$  is the patent number weight for different technical applications ( $y_i$ ) in a different stage ( $s_j$ ).  $p_{y_i,s_j}$  is the patent number for different technical purposes ( $y_i$ ) in a different stage ( $s_j$ ).

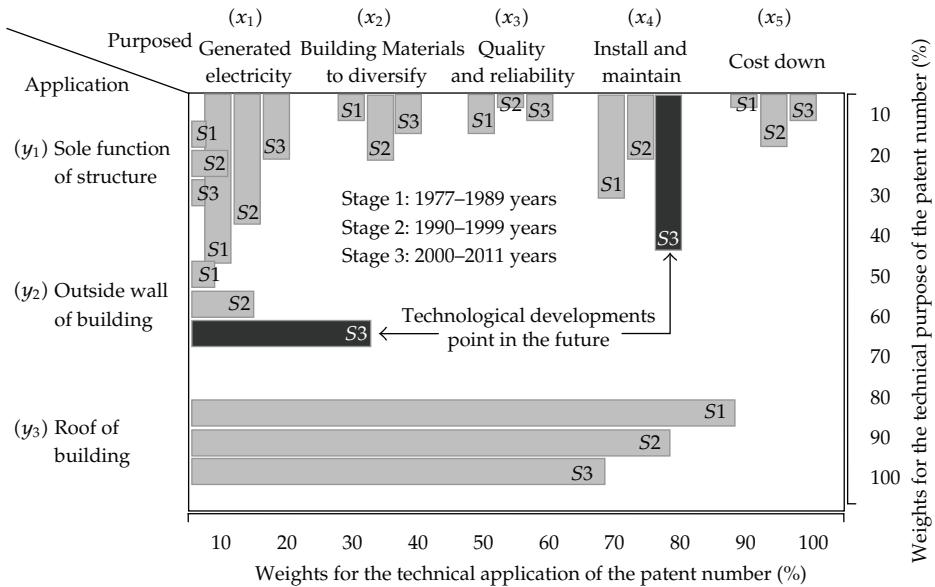
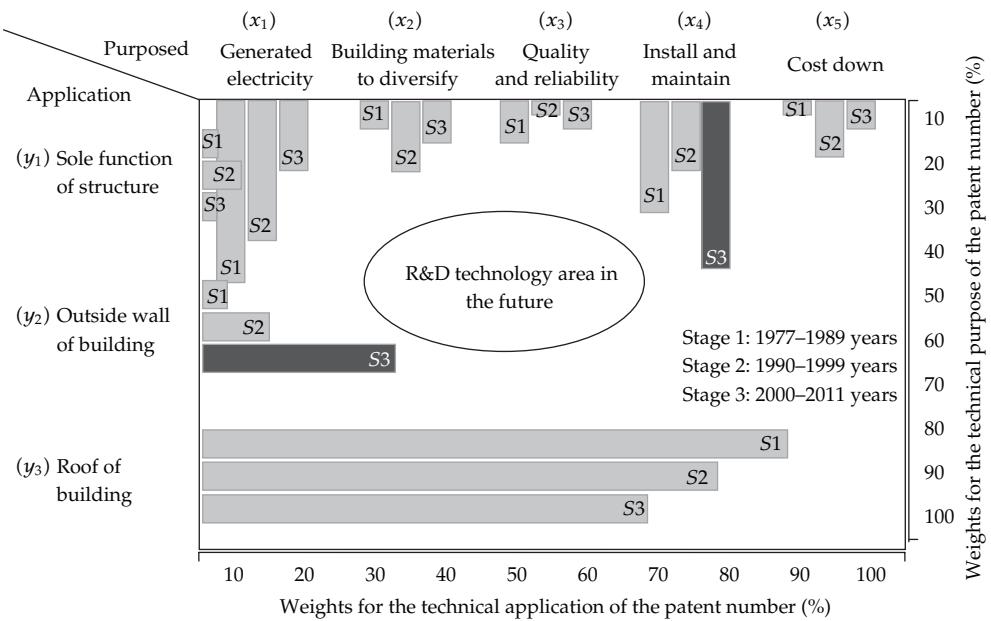


Figure 13: BIPV technology development trends.

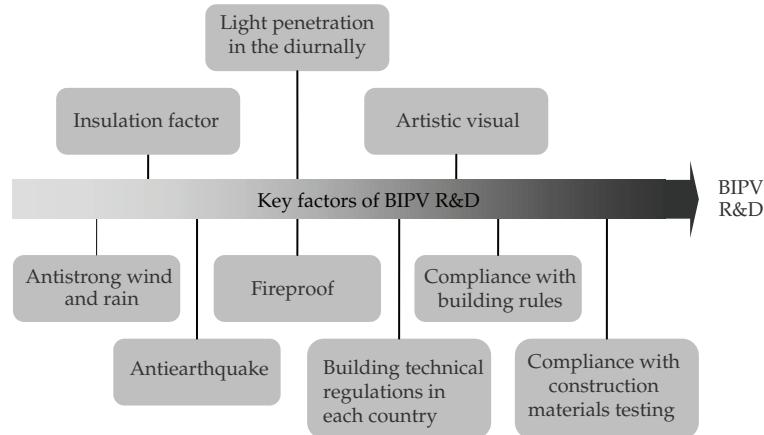
The objective of installation and maintenance is to ensure that diverse building materials in the outside walls of high-rise buildings are of high quality and reliable. The technical purpose weights of the number of diverse building material patents are low (Figure 14). Other than R&D into photovoltaic panels, they emphasize future methods of engineering and installing photovoltaic panels on the outside walls of high-rise buildings. However, safety and durability must be considered when developing diverse building materials. Indoor comfort must be considered when installing and maintaining buildings because people are central to BIPV. Key BIPV R&D factors are insulation, daylight penetration, visual appeal, wind and rain proof, earthquake proof, fireproof, national technical building regulations, compliance with building rules, and compliance with construction material tests (Figure 15 and Table 4).

#### 4.3. Technology Forecasting: Logistic Growth

This study uses Loglet Lab developed by Jason Yung, Perrin S. Meyer, and Jesse Ausubel as a tool to predict future development of BIPV technology trends. Technology start time, transition time, and saturation time are shown through the S-curve to depict the message (Figure 16). Up to October 31, 2011, a total of 263 pen announcement patents were approved by the USPTO patent database since July 26, 1977. The logistic growth curve diagram shows the technical saturation point for the 1881 patents. Defined according to the value of  $k$  [ $k \times 10\%$ ,  $k \times 90\%$ ], it corresponds to the growth point in time in February 2008. The turning point from growth to maturity is in January 2032. To close to 90% saturation point in time through the system to calculate the growth time of 49.4 years in June 2057, its BIPV technology patent saturation point is about 1881 cumulative patents. From the phenomenon of technology development at the application level, many patents focus on roof of building. However, the number of patents in the outside wall of building applications has grown well above the roof of building. This means that focus has moved to outside wall of building



**Figure 14:** BIPV technology R&D areas in the future.

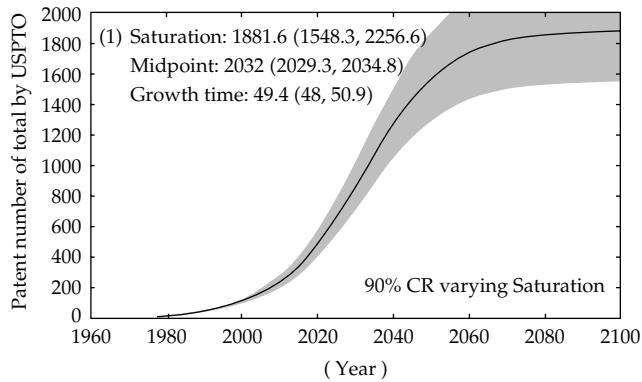
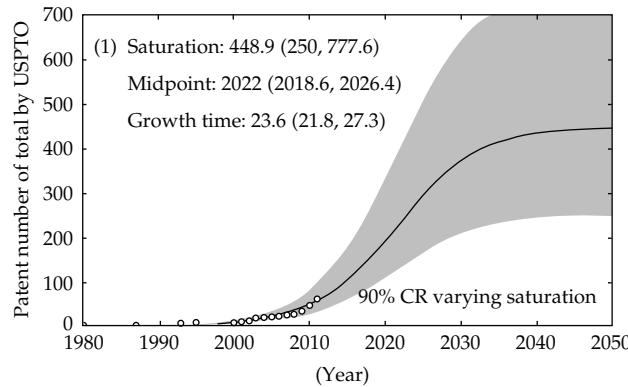


**Figure 15:** Key factors of BIPV R&D.

technology development. Therefore, this field of research for BIPV technology is in line with the number of patents in the outside wall of building development trend forecast in this study (Figure 17). Through the logistic growth, the curve graph shows the technical saturation point for the 449 patents. Defined according to the value of  $k$ ,  $[k \times 10\%, k \times 90\%]$  corresponds to the growth point in time, in February 2010. The turning point from growth to maturity is in January 2022. To close to 90% saturation point in time through the system to calculate the growth time of 23.6 years in August 2033, its BIPV technology patent saturation point about cumulative to 449 patents.

**Table 4:** The definition of the key factors of BIPV development.

Key factor	Definition
Insulation	Block thermal radiation heat from the sun and reduce indoor heat loads
Daylight penetration	Increase PV module light penetration and reduce interior lighting power demands
Artistic visual	PV fusion of the exterior of a building improves architectural aesthetics
Antistrong wind and rain	Resistant to strong wind and rain, thereby increasing building security
Antiearthquake	Resistant to earthquakes, thereby increasing building security
Fireproof	Resistant to fire, avoiding fire spreading from inside a building to adjacent buildings and avoiding external fires spreading inside
National technical building regulations	Must comply with national standards
Compliance with building rules	Must comply with construction technique rules
Compliance with construction material tests	Must comply with specifications for inspection and testing

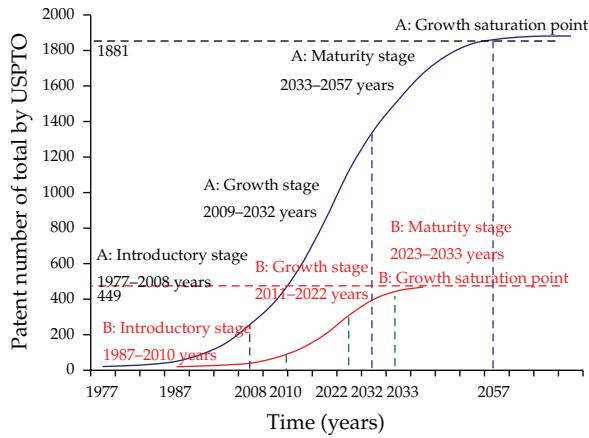
**Figure 16:** Technology growth curve of BIPV.**Figure 17:** Technology growth curve of BIPV for outside wall of building.

**Table 5:** Life cycle phases of BIPV.

Life cycle	Introductory stage	Growth stage	Maturity stage	Saturation stage
Beginning time	1977.07	2008.02	2032.01	2057.06

**Table 6:** Life cycle phases of BIPV on outside wall of building.

Life cycle	Introductory stage	Growth stage	Maturity stage	Saturation stage
Beginning time	1987.05	2010.02	2022.01	2033.08

**Figure 18:** S-curve of BIPV. S-curve A: BIPV technology develop forecasting. S-curve B: BIPV on outside wall of building technology develop forecasting.

By Loglet Lab statistics describing logistic growth curve diagram out BIPV technology life cycle in Table 5 and BIPV technology application in the outside wall of building development of the technology life cycle in Table 6.

Figure 18 shows the BIPV technology S-curve. S-curve A refers to BIPV technology developing forecasting, and S-curve B refers to BIPV outside wall building technology developing forecasting. S-curve A shows that the introductory BIPV stage was from 1977 to 2008; the growth stage is from 2009 to 2032; the maturity stage is from 2033 to 2057; the growth saturation point is near 2057. S-curve B shows that the introductory stage of BIPV outside wall building technology was from 1987 to 2010; the growth stage is from 2011 to 2022; the maturity stage is from 2023 to 2033; the growth saturation point is near 2033.

## 5. R&D Strategy

BIPV technology functions through the research and technical application of matrix state space dimensions, combined with the logistic growth curve that takes on a timeline surface. It shows BIPV technology patents in the field focused on the building roof assembly and installation techniques and applications. In the vertical plane of buildings, the technology and application of patents clearly show the phenomenon of loose or blank. This one-block technology for three-dimensional space of the buildings shows the use of maximum power generation area in the world's major countries to actively develop the future generation of

**Table 7:** BIPV technology key point in the future.

Technology node	Description
Durability	When PV modules are used in construction, the difference between the PV module and the building lifespan must be considered. The life of an average building is more than several decades or even centuries, but the life of a PV module is only 10–20 years. Therefore, in addition to enhancing photoelectric conversion efficiency, PV module durability should be a major focus in the future
Ease of maintenance	When faced with replacing PV modules of BIPV modules in high-rise buildings, consider (1) speed and ease of replacing the PV module (2) when PV modules become more popular, original specification sizes may be discontinued and replacement components may be difficult to find. New specifications must be developed to be compatible with previous specifications. BIPV installation and maintenance will become more important in the future
Security	Global warming has recently caused intense thermal radiation from the sun and frequent earthquakes and hurricanes. Therefore, BIPV glass in high-rise buildings must be antiradiative and resistant to strong wind, earthquakes, and fire. Research should also focus on developing building materials with these qualities
Customization	PV modules must be able to meet a wide range of building installation requirements. R&D should focus on methods of changing the production process to produce novel PV modules with different architectural designs. This will allow for a new industry in the future. It will be able to generate a new industry, that is, in the future producing a PV module will require semifinished products. A PV template will be cut according to a building shape and returned to the plant for postproduction to manage different construction types. This is customized BIPV building material development technology. It will be a new industry in the future and a model accepted by the construction industry

**Table 8:** Matrix of R&D strategy.

Patent	R&D	R&D strategy		Description
Degree of protection	R&D capability	Benefit creation	R&D strategy	Description
H	H	H	Leader	The patent with high-strength and high-product benefits usually adopt a cross-licensing model and other competitors to obtain the balance of terror
H	H	L	Attacker	Actively to the goal of Sheep to obtain royalty patent portfolio through intensive construction
H	L	H	Stalker	Through the acquisition of key patents, in addition to protection products and the other can sue the other party infringement strategy
H	L	L	Patent Troll	As a means through to obtain and hold patents and patent infringement litigation as a threat to force payment of royalties
L	H	H	The target	Revenue up to the scale of the degree will become the main subject matter of the patent litigation with the attacks to pay high royalties
L	H	L	Problem	Uncertainty region
L	L	H	Responder	Technologies and strategies for competitive advantage, and only passively follow the changes in the environment
L	L	L	Give up	

low-energy buildings. With technological development of a new generation of low-energy buildings, researchers also can evade patents in the patent radar district and also can compare the similarities and differences through analysis and interpretation of the patent information for reverse engineering in order to find a niche cut into the conducting dig technology policy. In addition, vertical walls of BIPV technology for space technology to divide policies are found in the lower reaches of this technology on the chain of a key, and that node is full of patent as long as the technology vendor in this field must go through this technology node (Table 7) and must pay patent royalties for different strategies.

Technology research and development policy is often associated with the market environment and technical ability having a close relationship. Technical developments, in particular in the early development of BIPV, through a technology growth curve that still has a long development time, and BIPV technology are the future of new generations of low buildings. Annual energy output of high technology and reduced costs to obtain construction and low construction investment for technology development and market factors continue. Therefore, enterprise technology products developed simultaneously must consider technology research and development capabilities, patent protection, and creating patent interest to locate the future development strategy of the three factors.

However, enterprises in competitive markets must improve manpower, time, and resources to maintain the technological lead. Technological backwardness of a company or country must be to narrow the distance with the technology leader in technology R&D and innovation. If companies want to enhance overall competitiveness, they must first assess their own abilities (including R&D personnel, capital, and the ability to create benefits and protect patents) and then, according to the size of the business and its abilities, select suitable R&D strategies. This study provides a three-dimensional matrix (Table 8) of degree of protection, R&D capability, and benefit creation to select R&D strategies. The first assessment indicator is the intensity of innovative R&D and the degree of patent protection. The second assessment indicator is the R&D capability of an enterprise with R&D manpower and resources. The third assessment indicator is the ability of an enterprise to create benefits using technology. This assessment method proposes eight strategies to assist an enterprise in locating their future R&D strategy when conducting BIPV technology R&D.

The results (Figures 13, 14, and 15) show that BIPV development moves gradually from roofs to outside walls or glass curtain walls. If companies want to produce a BIPV technology patent portfolio, they can use the three-dimensional matrix to assess and select a suitable R&D strategy according to individual enterprise circumstances. They can also use the patent map from this study to understand key high-intensity factor technologies for strategic actions to build important technology paths and patent intensity to increase profitability.

## 6. The Future Work of BIPV Industry

The results show that the BIPV industry is experiencing rapid growth and technology R&D. However, the BIPV manufacturing industry faces global competitive pressures, and the following characteristics can assist future development of the BIPV industry.

- (1) Low installation costs: because of different factors such as global economic conditions, the built environment, and landscape, balancing the regional economic environment and technology is important for the BIPV industry;
- (2) diversification of building materials: materials must be lightweight and fit building exteriors. PV modules must meet the requirements of different buildings, high

security and reliability, unique color, and pattern change while maintaining high photoelectric conversion efficiency. This enhances market appeal;

- (3) amorphous silicon solar-cell efficiency: amorphous silicon thin-film solar-cell modules can be bent and absorb diffuse light to produce electrical energy characteristics. These advantages are not matched by silicon solar-cell modules. Amorphous silicon thin-film solar-cell module use in the BIPV industry is small. However, power generation efficiency of amorphous silicon thin-film solar-cell modules is less than that of silicon solar-cell modules. In the future solar-cell market, silicon solar-cell modules will remain the main market, but amorphous silicon thin-film solar-cell modules will continue to grow. Therefore, enhancing the photoelectric conversion efficiency of amorphous silicon thin-film solar-cell modules should be a BIPV industry priority;
- (4) strategic integration of industry groups: to continually improve BIPV market attractiveness and reduce manufacturing costs, the BIPV industry must integrate the PV, construction, and electromechanical industries to form an industrial group. Active industrial and technical cooperation can promote and enhance the BIPV industry for future development;
- (5) import green building materials: as government subsidy policies gradually decrease, subsidies will no longer affect BIPV market development. Future BIPV technology development must be combined with the concept of green buildings and comply with building material specifications. Making BIPV products a part of green building materials is necessary for the future.

## 7. Conclusions and Recommendations

Due to the issue of global warming, the renewable energy technologies become more essential nowadays. The solar energy technology is concerned in the worldwide. Therefore, the interdisciplinary integration is an important trend. The building-integrated photovoltaic (BIPV) is an emerging technology integrating the photovoltaic and building fields. We are interested in the BIPV's development and aim to explore its evolution. From the perspective of patent analysis, this study hopes to offer some useful recommendations for BIPV's technology development. To explore life cycle of BIPV technology, we use logistic model analyzing patent numbers from the database of United States Patent and Trademark Office (USPTO).

We propose a hybrid approach in exploring the life cycle of BIPV technology and develop the R&D strategies of related industry. The proposed approach comprises the following submodules. First, patent analysis is employed to transform patent documents into structured data. Second, the logistic growth model is used to explore the life cycle of BIPV technology. Third, patent matrix map analysis is used to develop the R&D strategies of BIPV industry. Through the analysis by logistic model, the BIPV technology is expected a long-term life cycle from the emerging stage to the growth stage. The logistic results show the development of BIPV technology is growing fast since 2000 and still have development potential up to 2057.

The other important result is discovered in the three-dimensional matrix for R&D strategies in this paper. The three dimensions are R&D capability, degree of protection, and benefit creation of patent. There are eight patent-portfolio strategies for BIPV industry. We

think that the proposed model is useful for forecasting the trend of technology, especially in the field of BIPV. In the future, we hope it will also be applied in other technologies and industries.

## References

- [1] S. J. Liu and J. Shyu, "Strategic planning for technology development with patent analysis," *International Journal of Technology Management*, vol. 13, no. 5-6, pp. 661–680, 1997.
- [2] J. L. Chen, S. J. Liu, and C. H. Tseng, "Technological innovation and strategy adaptation in the product life cycle," *Technology Management: Strategies and Application*, vol. 5, no. 3, pp. 183–202, 2000.
- [3] S. Kuznets, "Innovative activity: problems of definition and measurement," in *The Rate and Direction of Inventive Activity*, R. Nelson, Ed., Princeton University Press, New Jersey, NY, USA, 1962.
- [4] B. H. Hall, Z. Griliches, and J. A. Hausman, "Patents and R&D: is there a lag?" *International Economic Review*, vol. 27, no. 2, pp. 265–283, 1986.
- [5] H. Ernst, "The use of patent data for technological forecasting: the diffusion of CNC-technology in the machine tool industry," *Small Business Economics*, vol. 9, no. 4, pp. 361–381, 1997.
- [6] S. B. Chang, *Establishing two stages technological positions model: derived from the patent analysis of business method* [Doctoral dissertation], National Yunlin University of Science and Technology, Yunlin, Taiwan, 2004.
- [7] W. Barroso, L. Quoniam, and E. Pacheco, "Patents as technological information in Latin America," *World Patent Information*, vol. 31, no. 3, pp. 207–215, 2009.
- [8] B. P. Abraham and S. D. Moitra, "Innovation assessment through patent analysis," *Technovation*, vol. 21, no. 4, pp. 245–252, 2001.
- [9] H. Ernst, "Patenting strategies in the German mechanical engineering industry and their relationship to company performance," *Technovation*, vol. 15, no. 4, pp. 225–240, 1995.
- [10] C. H. Guo, "Taiwan solar cell patent application for the current situation," *Guang Lian: Guang Dian Chan Ye Ji Shu Xue Bao*, vol. 76, pp. 50–56, 2008 (Chinese).
- [11] Y. B. Chen and X. Y. Cheng, "Solar cell patent analysis and development trends," in *Proceedings of the ITRI Innovation and Technology Management Conference*, pp. 1–8, 2006.
- [12] A. G. Hu, "The regionalization of knowledge flows in East Asia: evidence from patent citations data," *World Development*, vol. 37, no. 9, pp. 1465–1477, 2009.
- [13] R. J. W. Tijssen, "Global and domestic utilization of industrial relevant science: patent citation analysis of science—technology interactions and knowledge flows," *Research Policy*, vol. 30, no. 1, pp. 35–54, 2001.
- [14] C. V. Trappey, H. Y. Wu, F. Taghaboni-Dutta, and A. J. C. Trappey, "Using patent data for technology forecasting: China RFID patent analysis," *Advanced Engineering Informatics*, vol. 25, no. 1, pp. 53–64, 2011.
- [15] C. Choi and Y. Park, "Monitoring the organic structure of technology based on the patent development paths," *Technological Forecasting and Social Change*, vol. 76, no. 6, pp. 754–768, 2009.
- [16] Y. H. Chen, C. Y. Chen, and S. C. Lee, "Technology forecasting and patent strategy of hydrogen energy and fuel cell technologies," *International Journal of Hydrogen Energy*, vol. 36, no. 12, pp. 6957–6969, 2011 (Chinese).
- [17] A. C. Cheng, C. J. Chen, and C. Y. Chen, "A fuzzy multiple criteria comparison of technology forecasting methods for predicting the new materials development," *Technological Forecasting and Social Change*, vol. 75, no. 1, pp. 131–141, 2008.
- [18] W. B. Ashton and R. K. Sen, "Using patent information in technology business planning-II," *Research Technology Management*, vol. 32, no. 1, pp. 36–42, 1989.
- [19] S. M. Millett and E. J. Honton, *A Manager's Guide to Technology Forecasting and Strategy Analysis Methods*, Battelle Press, Ohio, USA, 1991.
- [20] A. L. Porter, *Forecasting and Management of Technology*, John Wiley & Sons, New York, NY, USA, 1991.
- [21] S. T. Yeh, "Design application and case study of BIPV in Taiwan," in *Proceedings of the 4th Architectural Forum in Taiwan—The Wisdom of Building and Eco-City*, National Federation of the Architects Association of the Republic of China, 2007.

- [22] T. Liour, C. W. Chen, C. B. Chen, and C. C. Chen, "White-light LED lighting technology life cycle forecasting and its National and company-wide competitiveness," in *Proceedings of the International Conference on Asia Pacific Business Innovation & Technology Management*, pp. 1–14, 2011.
- [23] W. T. Chang, *Technology development and corporation competence in magneto-resistive random access memory: a study base on USPTO patent* [Master's dissertation], Frng Chia University, Taiwan, 2007.

*Research Article*

## **Solving the Tractor and Semi-Trailer Routing Problem Based on a Heuristic Approach**

**Hongqi Li,<sup>1</sup> Yue Lu,<sup>1</sup> Jun Zhang,<sup>2</sup> and Tianyi Wang<sup>1</sup>**

<sup>1</sup> School of Transportation Science and Engineering, BeiHang University, No. 37 Xueyuan Road, Haidian District, Beijing 100191, China

<sup>2</sup> School of Economics and Management, BeiHang University, No. 37 Xueyuan Road, Haidian District, Beijing 100191, China

Correspondence should be addressed to Hongqi Li, lihongqi@buaa.edu.cn

Received 27 February 2012; Accepted 27 May 2012

Academic Editor: Jianming Shi

Copyright © 2012 Hongqi Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We study the tractor and semi-trailer routing problem (TSRP), a variant of the vehicle routing problem (VRP). In the TSRP model for this paper, vehicles are dispatched on a trailer-flow network where there is only one main depot, and all tractors originate and terminate in the main depot. Two types of decisions are involved: the number of tractors and the route of each tractor. Heuristic algorithms have seen widespread application to various extensions of the VRP. However, this approach has not been applied to the TSRP. We propose a heuristic algorithm to solve the TSRP. The proposed heuristic algorithm first constructs the initial route set by the limitation of a driver's on-duty time. The candidate routes in the initial set are then filtered by a two-phase approach. The computational study shows that our algorithm is feasible for the TSRP. Moreover, the algorithm takes relatively little time to obtain satisfactory solutions. The results suggest that our heuristic algorithm is competitive in solving the TSRP.

### **1. Introduction**

In this paper, we consider the tractor and semi-trailer routing problem (TSRP), a variant of the vehicle routing problem (VRP). The VRP is one of the most significant problems in the fields of transportation, distribution, and logistics. The basic VRP consists of some geographically dispersed customers, each requiring a certain weight of goods to be delivered (or picked up). A fleet of identical vehicles dispatched from a depot is used to deliver the goods, and the vehicles must terminate at the depot. Each vehicle can carry a limited weight and only one vehicle is allowed to visit each customer. It is assumed that some parameters (e.g., customer demands and travel times) are known with certainty. The solution of the problem consists

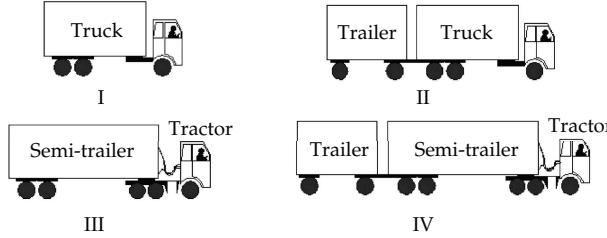
of finding a set of routes that satisfy the freight demand at minimal total cost. In practice, additional operational requirements and restrictions, as in the case of the truck and trailer routing problem (TTRP), may be imposed on the VRP [1]. The TTRP was first studied by Semet and Taillard et al. [2] and Gerdessen [3] in the 1990s, and it was subsequently studied by Chao [4], Scheuerer [5], and others. In the TTRP, the use of trailers (a commonly neglected feature in the VRP) is considered. Some customers can be served by a combination vehicle (i.e., a truck pulling a trailer, as in type II in Figure 1), while other customers can only be served by a truck (as type I in Figure 1) due to some limitations such as government regulations, limited maneuvering space at customer sites, road conditions, and so forth. These constraints exist in many practical situations [1].

The VRP and its various extensions have long been one of the most studied combinatorial optimization problems due to the problem's complexity and extensive applications in practice [6–10]. The truck and trailer combination is employed widely by enterprises around the world, but there are additional features introduced by trailers that have attracted some research. A number of studies have concentrated on applications of the TTRP. For instance, Semet and Taillard et al. [2] and Caramia and Guerriero et al. [11] gave some real-world TTRP applications in collection and delivery operations in rural areas or crowded cities with accessibility constraints. Theoretically, being an extension of the VRP, the TTRP is NP-Hard. The TTRP is computationally more difficult to solve than the VRP [1]. Because the VRP is usually tackled by heuristic methods [6–9, 12–15], it is feasible to develop heuristic approaches for the TTRP.

Gerdessen [3] extended the VRP to the vehicle routing problem with trailers and investigated the optimal deployment of a fleet of truck-trailer combinations by a construction and improvement heuristic. Scheuerer [5] proposed construction heuristics (called T-Cluster and T-Sweep) along with a tabu search algorithm for the TTRP. Tan et al. [16] proposed a hybrid multiobjective evolutionary algorithm featuring specialized genetic operators, variable-length representation; and local search heuristics to solve the TTRP. Lin et al. [1] proposed a simulated annealing (SA) heuristic for the TTRP and suggested that SA is competitive with tabu search (TS) for solving the TTRP. Villegas et al. [17] solved the TTRP by using a hybrid metaheuristic based on a greedy randomized adaptive search procedure (GRASP), variable neighborhood search (VNS), and path relinking (PR).

Villegas et al. [18] proposed two metaheuristics based on GRASP, VND, and evolutionary local search (ELS) to solve the single truck and trailer routing problem with satellite depots (STTRPSD). Considering the number of available trucks and trailers to be limited in the TTRP, Lin et al. [19] relaxed the fleet size constraint and developed a SA heuristic for solving the relaxed truck and trailer routing problem (RTTRP). Lin et al. [20] proposed a SA heuristic for solving the truck and trailer routing problem with time windows (TTRPTW).

Research to date has considered most types of road vehicles, especially trucks and truck and trailer combinations. However, there has been little research on the types of tractor and semi-trailer combinations. Hall and Sabnani et al. [21] studied routes that consisted of two or more segments and two or more stops in the tour for a tractor. At each stop, the tractor could drop off one or more trailers and pick up one or more trailers. Control rules based on predicted route productivity were developed to determine when to release a tractor. Derigs et al. [22] presented two approaches to solve the vehicle routing problem with multiple uses of tractors and trailers. The primary objective was to minimize the number of required tractors. Cheng et al. [23] proposed a model for a steel plant to find the tractor and semi-trailer equipment and running routes for the purpose of minimizing transport distance. Liang [24]



**Figure 1:** The basic types of vehicles. Note: In practice, many vehicle types are used in road freight transportation. This figure only lists four basic types. A large number of other types can be derived from the four basic types by the number of axles, tires and the combination style. Enterprises in most of the countries in the world employ various types.

established a dispatching model of tractors and semi-trailers in a large steel plant and used a tabu search algorithm to find the optimal driving path and the cycle program.

We aim to propose a heuristic for the TSRP. This aim is based on the practical knowledge that tractor and semi-trailer combinations are popular in some countries, particularly China. The remainder of this paper is organized as follows. Section 2 compares the TTRP and the TSRP and defines the TSRP. Section 3 proposes a heuristic algorithm to solve the TSRP. Section 4 employs the heuristic algorithm to solve some experimental networks of the TSR. Section 5 draws conclusions and gives future research directions.

## 2. Problem Definition

### 2.1. The TTRP and the TSRP

Although there is little literature devoted to the definition and solution of the TSRP in the fields of transportation or logistics, plenty of research has been done on the TTRP, providing important references for the TSRP. In the TTRP, a heterogeneous fleet composed of  $m_{tu}$  trucks and  $m_{tr}$  trailers ( $m_{tu} > m_{tr}$ ) serves a set of customers from a main depot. Each customer has a certain demand, and the distances between any two points (including customers and depots) are known. The capacities of the trucks and trailers are determinate. Some customers must be served only by a truck, while other customers can be served either by a truck or by a combination vehicle. The objective of the TTRP is to find a set of routes with minimum total distance or cost so that each customer is visited in a route performed by a compatible vehicle, the total demand of the customers visited on a route does not exceed the capacity of the allocated vehicle, and the numbers of required trucks and trailers are not greater than  $m_{tu}$  and  $m_{tr}$ , respectively [1, 17]. There are three types of routes in a TTRP solution, as illustrated in Figure 2: (1) a pure truck route traveled by a single truck; (2) a pure vehicle route without any subtour traveled by a combination vehicle; (3) a combination vehicle route consisting of a main tour traveled by a combination vehicle and at least one subtour traveled by the truck alone.

The vehicle types in the TSRP are different from those in the TTRP. The TTRP focuses on trucks and trailers, both of which can carry cargo. The TSRP involves  $m_{ta}$  tractors and  $m_{st}$  semi-trailers ( $m_{ta} < m_{st}$ ). Although a tractor cannot carry cargo, it has more flexible dispatching options, and it can pull different semi-trailers on various segments of its route by the pick-up and drop-off operation at depots.

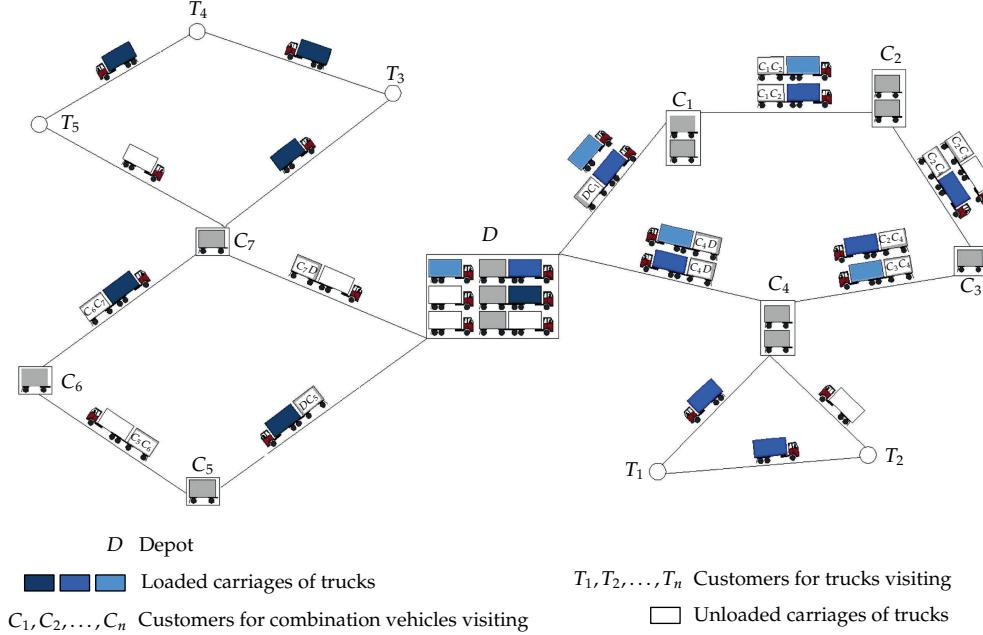
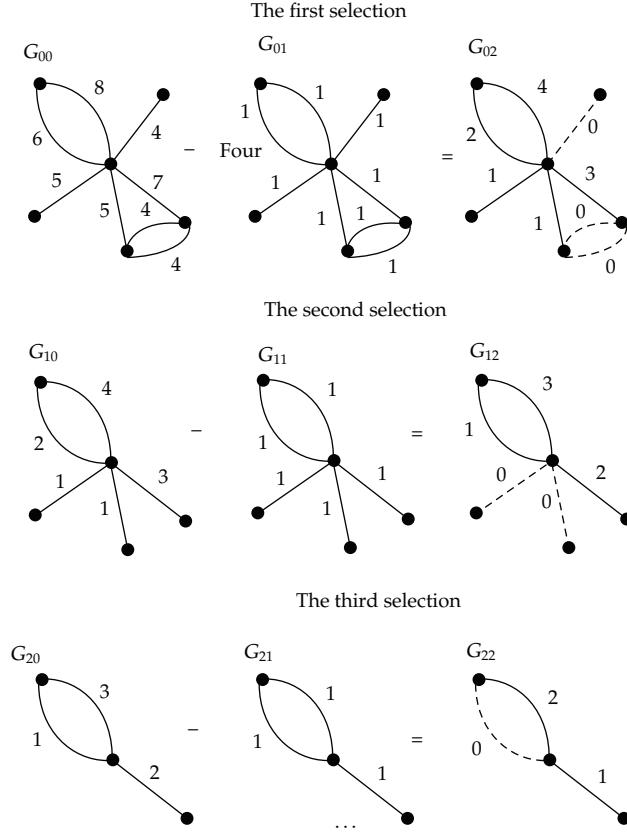


Figure 2: Different types of vehicle routes in the TTRP.

The TSRP can be formally defined on a directed graph  $G = (V, A)$ , where  $V = \{0, 1, 2, \dots, n\}$  is the set of vertices and  $A = \{(i, j) : i, j \in V\}$  is the set of arcs. Each arc  $(i, j)$  is generally associated with a transportation distance decided by road infrastructure. The freight flow from  $i$  to  $j$  is regarded as certain weight of arc  $(i, j)$ . Vertex  $0, \dots, v$  ( $v < n$ ) represents the main depots, in which many tractors and semi-trailers park. Some loaded semi-trailers wait for visiting customers, and other unloaded semi-trailers wait for visiting or maintenance. The remaining vertices ( $s_i$ ) in  $V$  (i.e.,  $V \setminus \{0, \dots, v\}$ ) correspond to customers who have  $m$  ( $m \geq 1$ ) loaded semi-trailers waiting to visit  $l$  ( $1 \leq l \leq n$  and  $l \leq m$ ) orientations at the beginning of the simulation. Customers may have other unloaded semi-trailers waiting for loading.

There are various tractor-driving modes on graph  $G$  during one daily period. For example, (1) tractor  $\text{Tr}_j$ , pulling one loaded semi-trailer, goes from its main depot to a customer in one-day period, and the customer has tractor-parking available; (2) tractor  $\text{Tr}_j$ , pulling one loaded semi-trailer, goes from its main depot to  $c_1$ . After the dropping and pulling operations at  $c_1$ , the tractor goes on to another customer,  $c_2$ . The tractor  $\text{Tr}_j$  terminates at a customer who has tractor-parking available. (3) It is similar to the running course listed in (2), but Tractor  $\text{Tr}_j$  terminates at its main depot. The most basic elements of tractor-driving modes include the following: how many semi-trailers can be pulled synchronously by a tractor, how many vertexes are passed by the tractor, how many times per day the tractor can drop off one or more trailers and pick up one or more trailers, whether a tractor terminates at its original main depot, whether the semi-trailer pulled by a tractor loads cargo, and if a tractor runs alone. In addition, a time window constraint is probably required.



**Figure 3:** An example of selecting unit-flow networks. Note: Black points denote depots. Lines denote the distribution of freight flows. Numbers near lines denote freight flow volume (unit: one semi-trailer).

## 2.2. The TSRP Model

In practice, there are many depots on the freight transportation network of an enterprise. Depots have different functions and sizes. In the TSRP, we classify these depots into two types: main depots and customer depots. The flow of freight between any two depots is usually uneven. In our method, we abstract the freight transportation network onto a graph (denoted by  $G_{00}$ ). Graph  $G_{00}$  has one main depot and a number of customer depots where semi-trailers can park. Initially, all tractors are parked in the main depot, and semi-trailers that carry cargos are waiting for transport.

Because the freight flows among various depots are unequal, we select a freight flow network denoted by  $G_{01}$ .  $G_{01}$  is a subset of  $G_{00}$  on which the freight flows among various depots are equal. Graph  $G_{01}$  is probably a combination of some unit-flow network. On a unit-flow network, the freight flow on every arc is one semi-trailer. We denote the subset  $G_{00} - G_{01}$  by  $G_{02}$ . We go on to select another equal flow network (denoted by  $G_{11}$ ) from  $G_{02}$ . After repeating this process several times, the original freight flow network is split into several unit-flow networks (e.g.,  $G_{01}$ ,  $G_{11}$ , and  $G_{21}$  in Figure 3). The study of unit-flow networks is meaningful and important in solving the TSRP.

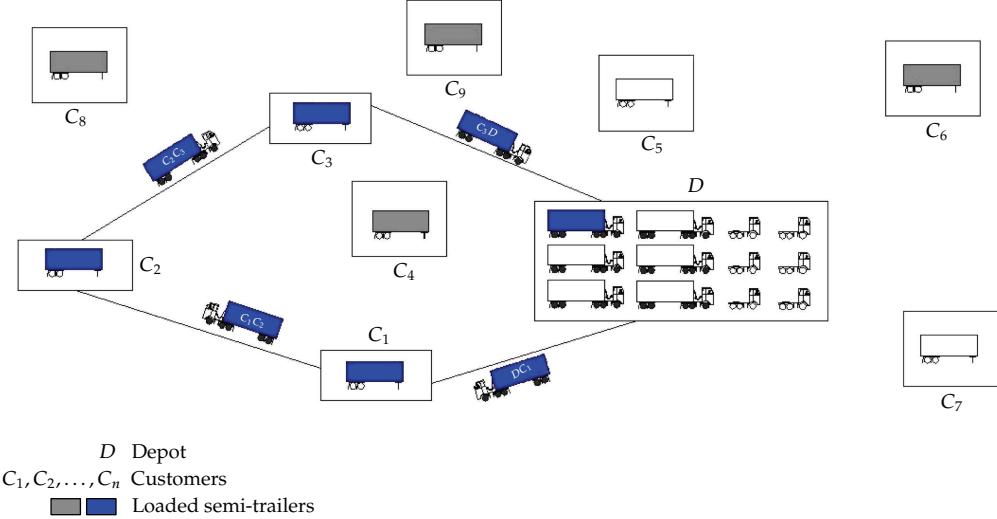


Figure 4: Vehicle routes in the TSRP model.

The TSRP model in this paper uses the unit-flow network. Vertex 0 represents the main depot where some loaded semi-trailers are waiting to be delivered to customers. The vertices in  $V \setminus \{0\}$  correspond to customers who have  $m$  ( $1 \leq m \leq n$ ) loaded semi-trailers waiting for going to  $m$  customers. The  $n$  customers have no parked tractors. The tractor-driving modes must satisfy some constraints, including the following: tractors terminate at the main depot, a tractor can pull one loaded semi-trailer and can also run alone, and the working time of a tractor is decided by its driver team. All tractors or vehicles (a vehicle is one tractor pulling one semi-trailer) originate and terminate at the main depot. Whenever a tractor passes by a customer, the tractor picks up a semi-trailer. Whenever a vehicle passes by a customer, the vehicle drops off its semi-trailer and picks up another one. After one-day period, the number of semi-trailers parked in every customer point is not less than a minimum (Figure 4).

The TSRP model consists of determining the number of tractors to be used and the route of each tractor so that the variable costs and service level are balanced, while each route starts and ends at the main depot. Variable costs are reduced by decreasing the overlap distance of tractors running alone. The service level is based on the percentage of customer demand that is satisfied.

### 3. A Heuristic Algorithm for the TSRP

#### 3.1. Construct the Initial Solution Set

Drivers are assigned to ensure flexible running of tractors. A driver's on-duty hours per day are determined by the legal on-duty time and driver dispatching mode. On-duty hours consist of driving hours plus temporary rest time or residence time in depots. The residence time at the main depot (denoted by  $H$ ) is for pick-up, drop-off, and some essential maintenance work on vehicles. The temporary rest time at other depots (denoted by  $s_i$  ( $i = 1, 2, \dots$ )) is for pick-up and/or drop-off semi-trailers. The driving hours restrict tractor running time.

The elements of the initial solution set are tractor routes. To give the form of a route, we suggest the following procedure. The number of drivers assigned to each tractor is  $k$ . The on-duty time of each driver is  $T$  (hours per person · day). The distance between depots  $i$  and  $j$  is  $d_{ij}$ . The depot sequence on each route is denoted by  $H - s_1 - \dots - s_f - H$ , which is the form of a route. The same customer is visited only once on a certain tractor's route, and each  $s_i$  ( $i = 1, 2, \dots, f$ ) is unique. The on-duty time  $T$  is a constraint on the route. That is,

$$\rho_1 \cdot kT \leq \sum_i \sum_j \left( \frac{d_{ij}}{v} \right) + f \cdot t_s + 2 \cdot t_H \leq \rho_2 \cdot kT, \quad (3.1)$$

where  $t_s$  and  $t_H$  are the temporary rest time and the residence time, respectively.  $\rho_1$  ( $0 < \rho_1 < 1$ ) and  $\rho_2$  ( $1 \leq \rho_2 \leq \tau$ ,  $\tau$  is a limited number) are the lower and upper limits of the utilization ratio of the on-duty time, respectively.  $v$  is the average velocity of the tractor.

We suggest the steps below to construct elements of the initial solution set.

*Step 1.* Transform the distance matrix into a running time matrix. Use  $v$  as the parameter in the transformation. Factors affecting  $v$  in practice include the tractor condition, the driver's skill, and traffic conditions. We estimate  $v$  by enterprise experience.

*Step 2.* Search the running time matrix. Let  $f$  be the sum of customers on a route. If  $f$  is very large, there are too many customers on the route to allow too much temporary rest time. Therefore,  $f$  has a maximum, and the maximum is certainly less than  $(kT - 2 \cdot t_H)/t_s$ . Once  $f$  is found, we implement an entire search on the running time matrix to find all routes that satisfy the constraint (3.1).

*Step 3.* Compare the routes with freight flow demand. Every route in Step 2, which contains many segments, is constructed by the segment running time of all customer pairs. In fact, not all pairs of customers require freight exchange. There are segments on which no freight flows, and tractors run alone on such segments. To save variable costs, the time of tractors running alone is limited. Therefore, we obtain the initial solution set after the elimination of routes based on the freight flow demand and the cost-saving requirement.

### 3.2. A 2-Phase Approach to Improve the Initial Solution Set

#### 3.2.1. The First Phase

The more elements (i.e., tractor routes) in the initial solution set, the more choices for freight enterprises. We classify tractor routes into certain types, according to the number of customers on a route. There is one customer passed by in the 1st type, two customers passed by in the 2nd type, and so on. When there are many customers on a route, the tractor can serve more freight demand. When more temporary rest time is consumed at customer points, the effective running hours of the tractor are reduced. Routes of the same type generally have some "overlap arcs" in which only one tractor pulls a semi-trailer, and others run alone. We suggest reducing the total distance of "overlap arcs."

- (1) The first step is for the same type. An overlap arc  $(i, j)$  where the tractor running time is less than a maximum  $(t_{ij})_{\max}$  can be accepted. If the tractor needs more time

than  $(t_{ij})_{\max}$  on arc  $(i, j)$ , only one of the routes containing arc  $(i, j)$  is permitted to be chosen.

- (2) The second step is for different types. A “tractor route—overlap arc” matrix ( $A_0$ , with elements  $a_{ik}$ ) is constructed. The rows of matrix  $A_0$  are serial numbers ( $sn_i$ ) of routes and the columns are various overlap arcs. The elements of matrix  $A_0$  are 1 or 0. If the element on row  $i$  and column  $k$  is “1”, then the  $sn_i$  route has an overlap arc. When  $a_{ik} = 1$ , any elements of the matrix that satisfy  $a_{ij} = 1$  are considered. The column which contains  $a_{ij}$  has a sum  $\sum_i a_{ij}$ . If the route with serial number  $sn_i$  is chosen,  $\sum_j \sum_i a_{ij}$  should be at a minimum. Once the  $sn_i$  route is chosen, other routes that have the same overlap arcs with the  $sn_i$  route are eliminated from  $A_0$ . Consequently, a row in  $A_0$  changes, and a new matrix  $A_1$  appears. The operation is repeated until there is no row available in the last “tractor route—overlap arc” matrix.

In the first phase, a transitional solution set that contains such elements as the  $sn_i$  routes is constructed by improving the initial solution set.

### 3.2.2. The Second Phase

In the second phase, we propose the “fill-and-cut” approach to attain a satisfactory solution to the TSRP.

*Step 1.* Construct a zero matrix  $O$  (its elements are  $o_{ij}$ ) whose rows and columns are depots (i.e., the main depot and customer depots). Because of transportation demand, there are freight flows between two particular depots. Because the segments of tractor routes in the transitional solution set are defined by depots, we add 1 to  $o_{ij}$  when there is a route containing arc  $(i, j)$ . We call such an operation a “fill”. By a “fill” operation, we mark all segments of routes in the transitional solution set on matrix  $O$ . A new matrix  $B_0$  is thus formed.

*Step 2.* All route segments in the transitional solution set actually have corresponding elements in matrix  $B_0$ . If a certain percentage (e.g., 80~100%) of all corresponding elements of the  $sn_i$  route are greater than 1, the  $sn_i$  route is eliminated. When the  $sn_i$  route is eliminated, all of the corresponding elements subtract 1. We call such an operation a “cut”. Repeat the “cut” operation, and a new matrix  $B_1$  finally forms. The routes corresponding to  $B_1$  make up the satisfactory solution set.

In some cases, the number of nonzero elements of  $B_1$  is less than that of the freight flow demands. Therefore, routes corresponding to  $B_1$  cannot satisfy all transportation demand. In order to satisfy more customers’ demands, we can add some routes that contain overlap arcs to increase the market adaptability of the satisfactory solution. However, too many overlap arcs can exist because of uneven freight flows. Therefore, to balance the service level and costs, meeting a certain percentage (e.g., 80%) of all transportation demand can be the objective.

## 4. Computational Study

We abstract the transportation network on an  $N \times N$  grid, where the nodes denote the main depot and customer depots. In our computational study, the “RANDOM” function in Matlab,

**Table 1:** The tractor running time between two depots (hours).

From	H	To												
		1	2	3	4	5	6	7	8	9	10	11		
H	0	5.5	4.0	4.5	1.5	1.0	1.0	4.0	3.0	4.5	1.0	3.5	3.5	3.0
1	5.5	0	1.5	1.0	4.0	4.5	4.5	2.5	2.5	3.0	5.5	4.0	5.0	6.5
2	4.0	1.5	0	2.5	2.5	3.0	3.0	4.0	3.0	4.5	4.0	3.5	3.5	5.0
3	4.5	1.0	2.5	0	3.0	3.5	3.5	1.5	1.5	2.0	4.5	3.0	4.0	5.5
4	1.5	4.0	2.5	3.0	0	0.5	0.5	4.5	3.5	5.0	1.5	4.0	4.0	3.5
5	1.0	4.5	3.0	3.5	0.5	0	1.0	5.0	4.0	5.5	2.0	4.5	4.5	4.0
6	1.0	4.5	3.0	3.5	0.5	1.0	0	4.0	3.0	4.5	1.0	3.5	3.5	3.0
7	4.0	2.5	4.0	1.5	4.5	5.0	4.0	0	1.0	0.5	4.0	2.5	3.5	5.0
8	3.0	2.5	3.0	1.5	3.5	4.0	3.0	1.0	0	1.5	3.0	1.5	2.5	4.0
9	4.5	3.0	4.5	2.0	5.0	5.5	4.5	0.5	1.5	0	3.5	2.0	3.0	4.5
10	1.0	5.5	4.0	4.5	1.5	2.0	1.0	4.0	3.0	3.5	0	2.5	2.5	2.0
11	3.5	4.0	3.5	3.0	4.0	4.5	3.5	2.5	1.5	2.0	2.5	0	1.0	2.5
12	3.5	5.0	3.5	4.0	4.0	4.5	3.5	3.5	2.5	3.0	2.5	1.0	0	1.5
13	3.0	6.5	5.0	5.5	3.5	4.0	3.0	5.0	4.0	4.5	2.0	2.5	1.5	0

**Table 2:** The freight flow between two depots (one semi-trailer).

From	H	To											
		1	2	3	4	5	6	7	8	9	10	11	
H	0 <sup>b</sup>	1 <sup>a</sup>	1	1	1	1	1	1	1	1	1	1	1
1	1	0	1	0	1	1	0	0	0	0	0	0	1
2	1	1	0	0	0	0	0	0	1	0	1	1	0
3	1	0	0	0	1	0	0	0	0	0	1	0	1
4	1	0	0	1	0	0	0	0	1	1	0	0	1
5	1	0	0	1	1	0	0	0	0	0	1	0	1
6	1	0	0	0	1	1	0	0	0	0	1	1	0
7	1	0	0	0	1	0	0	0	0	0	1	1	0
8	1	0	1	0	0	0	1	0	0	1	0	1	0
9	1	0	1	0	1	0	1	0	0	0	0	1	0
10	1	1	0	0	0	1	0	1	1	0	0	0	0
11	1	0	1	0	1	0	0	0	1	0	0	0	1
12	1	0	1	0	0	1	0	0	0	0	0	1	0
13	1	0	0	0	1	0	1	0	1	1	0	0	0

<sup>a</sup>“1” denotes that there is one semi-trailer flow between two depots.<sup>b</sup>“0” denotes that there is no freight flow.

which can generate random arrays from a specified distribution, is used. By RANDOM (“norm”,1,1,10,10), a random array is generated. We select the negative positions of the array as nodes and the minimum position as the main depot. The distance between any two nodes is calculated by the gaps of rows and columns. The “RANDOM” function in MATLAB is also used to determine the freight flow between two depots. The network expressed by Table 1 and the flow expressed by Table 2 make up example No. 1. By the above generation method, we produce some transportation networks that are used to test the heuristic algorithm.

**Table 3:** The satisfactory solution of the TSRP achieved by the heuristic algorithm.

Types of the routes	Form of route	Working time needed by routes (hours)
A tractor with two drivers. Two different customer depots are passed by. The route repeats once a day.	$H-1-13-H$	17
A tractor with two drivers. Three different customer depots are passed by. The route repeats once a day.	$H-3-13-4-H; H-7-12-2-H;$ $H-9-2-8-H; H-11-4-3-H;$ $H-11-13-9-H; H-12-5-13-H$	17.5
A tractor with two drivers. Four different customer depots are passed by. The route repeats once a day.	$H-5-10-7-12-H;$ $H-8-9-2-10-H$	17
A tractor with two drivers. Five different customer depots are passed by. The route repeats once a day.	$H-2-8-6-10-5-H;$ $H-6-4-13-8-11-H;$ $H-6-5-3-12-11-H;$ $H-6-11-8-9-4-H;$ $H-10-8-2-1-5-H$	17.5
A tractor with two drivers. Six different customer depots are passed by. The route repeats once a day.	$H-6-5-4-8-9-11-H$	17
A tractor with two drivers. Two different customer depots are passed by. The route repeats twice a day.	$H-8-6-H-13-6-H$	17.5

According to some enterprise experience, a driver's on-duty time is 8.5 hours per day. One or two drivers are assigned to a tractor. A tractor with two drivers can work consecutively for no more than 17 hours in a 24-consecutive-hour period. The temporary rest time in customer depots is 0.5 hour, and the residence time in  $H$  is 1 hour. By using the approach mentioned in Section 3.1, we attain different types of tractor routes for the No. 1 example. There are 175 elements in the initial solution set. By using the 2-phase approach, we attain the satisfactory solution of the No. 1 example (Table 3). When the enterprise employs tractor routes as the satisfactory solution, it can satisfy 80 percent of all transportation demand. Sixteen tractors and thirty-two drivers are needed during a 24-consecutive-hour period. The total running time of 16 tractors is 230 hours per day. In 15 percent of the total running time, tractors run alone.

It is feasible to propose exact algorithms (e.g., integer programming) for the TSRP when the initial solution set is constructed. We proposed a 0-1 integer programming for the No. 1 example and attained the exact solution. The exact solution can satisfy 84 percent of all transportation demand. Sixteen tractors and thirty-two drivers are needed during a 24-consecutive-hour period. In 10 percent of the total running time, tractors run alone. We implemented the proposed heuristic algorithm using Matlab and the 0-1 integer programming with QS. Although the exact algorithm can attain a slightly better solution, it requires more calculating time. For the No. 1 example, the solving time using the heuristic algorithm was approximately 80 seconds while that for the exact algorithm was approximately 2000 seconds. We suggest that the heuristic algorithm has an advantage for solving the TSRP.

**Table 4:** The results of TSRP experiments achieved by the heuristic algorithm.

Transportation network			On the solution		Calculation time (seconds)
Number of nodes	Number of freight flows	Fraction of demand satisfied (%)	Number of tractors	Average time of tractors running alone (hours)	
10	54	65	10	0.9	3
11	60	85	18	1.9	5
12	66	89	17	1.8	26
13	72	92	22	2.1	45
14	78	89	23	1.6	78
15	84	71	18	1.8	47
17	96	89	28	1.7	59
18	102	81	25	1.6	86
19	108	86	32	1.7	114
21	120	74	28	1.6	198
22	126	71	28	1.4	234
23	132	86	37	1.8	310

We have repeated the generation of random arrays over 50 times to obtain some typical computational networks. The heuristic algorithm was employed on these networks. We ran the experiments of this section on a computer with an AMD Athlon(tm) X2 Dual-Core QL-65 running at 2.10GHz under Windows 7 ultimate (32 bits) with 2GB of RAM. Table 4 summarizes the characteristics of each solution in the 12-instance testbed.

## 5. Conclusions and Future Work

In this paper, we proposed a TSRP model and suggested a heuristic algorithm to solve it. The TSRP concentrated on a unit-flow network, and all tractors originated and terminated at a main depot. Unlike most approaches to the TTRP or VRP, the heuristic algorithm for the TSRP did not regard the number of vehicles as a precondition. Therefore, the solution to the TSRP was able to balance the variable costs and service level by altering the vehicle number. The main characteristics of the heuristic algorithm are the initial solution set constructed by the limitation of driver on-duty time and the combination of a two-phase filtration on candidate routes. The computational study shows that our algorithm is feasible and effective for the TSRP. Although some exact algorithms for the TSRP are feasible after the initial solution set is constructed, the heuristic algorithm is efficient because it takes relatively less time to obtain satisfactory solutions. Future research may try to extend the TSRP to include more practical considerations, such as time window constraints. Other efficient heuristics for the TSRP may also be proposed. In this regard, the benchmark instances generated in this study may serve as a testbed for future research to test the efficiency of specific algorithms for TSRP.

## Acknowledgments

This work was partially funded by the Science and Technology Plan of Transportation of Shandong Province (2009R58) and the Fundamental Research Funds for the Central Universities (YWF-10-02-059). This support is gratefully acknowledged.

## References

- [1] S. W. Lin, V. F. Yu, and S. Y. Chou, "Solving the truck and trailer routing problem based on a simulated annealing heuristic," *Computers and Operations Research*, vol. 36, no. 5, pp. 1683–1692, 2009.
- [2] F. Semet and E. Taillard, "Solving real-life vehicle routing problems efficiently using tabu search," *Annals of Operations Research*, vol. 41, no. 4, pp. 469–488, 1993.
- [3] J. C. Gerdessen, "Vehicle routing problem with trailers," *European Journal of Operational Research*, vol. 93, no. 1, pp. 135–147, 1996.
- [4] I. M. Chao, "A tabu search method for the truck and trailer routing problem," *Computers and Operations Research*, vol. 29, no. 1, pp. 33–51, 2002.
- [5] S. Scheuerer, "A tabu search heuristic for the truck and trailer routing problem," *Computers and Operations Research*, vol. 33, no. 4, pp. 894–909, 2006.
- [6] J. Renaud and F. F. Boctor, "A sweep-based algorithm for the fleet size and mix vehicle routing problem," *European Journal of Operational Research*, vol. 140, no. 3, pp. 618–628, 2002.
- [7] H. C. Lau, M. Sim, and K. M. Teo, "Vehicle routing problem with time windows and a limited number of vehicles," *European Journal of Operational Research*, vol. 148, no. 3, pp. 559–569, 2003.
- [8] F. Li, B. Golden, and E. Wasil, "A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem," *Computers and Operations Research*, vol. 34, no. 9, pp. 2734–2742, 2007.
- [9] S. Liu, W. Huang, and H. Ma, "An effective genetic algorithm for the fleet size and mix vehicle routing problems," *Transportation Research Part E*, vol. 45, no. 3, pp. 434–445, 2009.
- [10] E. Cao and M. Lai, "The open vehicle routing problem with fuzzy demands," *Expert Systems with Applications*, vol. 37, no. 3, pp. 2405–2411, 2010.
- [11] M. Caramia and F. Guerriero, "A milk collection problem with incompatibility constraints," *Interfaces*, vol. 40, no. 2, pp. 130–143, 2010.
- [12] D. Pisinger and S. Ropke, "A general heuristic for vehicle routing problems," *Computers & Operations Research*, vol. 34, no. 8, pp. 2403–2435, 2007.
- [13] Y. Marinakis and M. Marinaki, "A hybrid multi-swarm particle swarm optimization algorithm for the probabilistic traveling salesman problem," *Computers & Operations Research*, vol. 37, no. 3, pp. 432–442, 2010.
- [14] A. Garcia-Najera and J. A. Bullinaria, "An improved multi-objective evolutionary algorithm for the vehicle routing problem with time windows," *Computers & Operations Research*, vol. 38, no. 1, pp. 287–300, 2011.
- [15] L. Hong, "An improved LNS algorithm for real-time vehicle routing problem with time windows," *Computers and Operations Research*, vol. 39, no. 2, pp. 151–163, 2012.
- [16] K. C. Tan, Y. H. Chew, and L. H. Lee, "A hybrid multi-objective evolutionary algorithm for solving truck and trailer vehicle routing problems," *European Journal of Operational Research*, vol. 172, no. 3, pp. 855–885, 2006.
- [17] J. G. Villegas, C. Prins, C. Prodhon, A. L. Medaglia, and N. Velasco, "A GRASP with evolutionary path relinking for the truck and trailer routing problem," *Computers and Operations Research*, vol. 38, no. 9, pp. 1319–1334, 2011.
- [18] J. G. Villegas, C. Prins, C. Prodhon, A. L. Medaglia, and N. Velasco, "GRASP/VND and multi-start evolutionary local search for the single truck and trailer routing problem with satellite depots," *Engineering Applications of Artificial Intelligence*, vol. 23, pp. 780–794, 2010.
- [19] S. W. Lin, V. F. Yu, and S. Y. Chou, "A note on the truck and trailer routing problem," *Expert Systems with Applications*, vol. 37, no. 1, pp. 899–903, 2010.
- [20] S. W. Lin, V. F. Yu, and C. C. Lu, "A simulated annealing heuristic for the truck and trailer routing problem with time windows," *Expert Systems with Applications*, vol. 38, pp. 15244–15252, 2011.
- [21] R. W. Hall and V. C. Sabnani, "Control of vehicle dispatching on a cyclic route serving trucking terminals," *Transportation Research Part A*, vol. 36, no. 3, pp. 257–276, 2002.
- [22] U. Derigs, R. Kurowsky, and U. Vogel, "Solving a real-world vehicle routing problem with multiple use of tractors and trailers and EU-regulations for drivers arising in air cargo road feeder services," *European Journal of Operational Research*, vol. 213, no. 1, pp. 309–319, 2011.
- [23] Y. R. Cheng, B. Liang, and M. H. Zhou, "Optimization for vehicle scheduling in iron and steel works based on semi-trailer swap transport," *Journal of Central South University of Technology*, vol. 17, no. 4, pp. 873–879, 2010.
- [24] B. Liang, *Research on semi-trailer loop swap transportation applied in large-scale iron and steel works [M.S. thesis]*, Central South University, 2009.

*Research Article*

## **Combining Diffusion and Grey Models Based on Evolutionary Optimization Algorithms to Forecast Motherboard Shipments**

**Fu-Kwun Wang,<sup>1,2</sup> Yu-Yao Hsiao,<sup>2</sup> and Ku-Kuang Chang<sup>3</sup>**

<sup>1</sup> Department of Industrial Management, National Taiwan University of Science and Technology, Taipei 106, Taiwan

<sup>2</sup> Graduate Institute of Management, National Taiwan University of Science and Technology, Taipei 106, Taiwan

<sup>3</sup> Department of Logistic Management, Takming University of Science and Technology, Taipei 114, Taiwan

Correspondence should be addressed to Fu-Kwun Wang, fukwun@mail.ntust.edu.tw

Received 21 February 2012; Revised 12 June 2012; Accepted 15 June 2012

Academic Editor: Yi-Chung Hu

Copyright © 2012 Fu-Kwun 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.

It is important for executives to predict the future trends. Otherwise, their companies cannot make profitable decisions and investments. The Bass diffusion model can describe the empirical adoption curve for new products and technological innovations. The Grey model provides short-term forecasts using four data points. This study develops a combined model based on the rolling Grey model (RGM) and the Bass diffusion model to forecast motherboard shipments. In addition, we investigate evolutionary optimization algorithms to determine the optimal parameters. Our results indicate that the combined model using a hybrid algorithm outperforms other methods for the fitting and forecasting processes in terms of mean absolute percentage error.

### **1. Introduction**

Taiwanese motherboard manufacturers create 98.5% of the worldwide desktop motherboards and dominate the global desktop motherboard market [1]. However, this industry's growth rate has slowed because of the trend of replacing desktop PCs with notebooks or netbooks. In addition, the aggressive pricing by notebook/netbook manufacturers has diminished desktop motherboard sales. It is important to develop a new forecasting model for this rapidly changing market and to compare its results with other forecasting models. These results can assist manufactures in making decisions on future expansion and investment.

Several studies have proposed time-series models for industrial production, demonstrating the applicability of time-series models to industrial production forecasting. These

models typically require large amounts of data. However, Hsu [2] proved that the Grey model (GM), developed by Deng [3], requires minimal data and is the best model for limited data prediction. Furthermore, GM can forecast in a compleptive environment where decision makers have limited historical data. Chang et al. [4] applied a variable  $p$  value to a rolling Grey model (RGM) to forecast semiconductor production in Taiwan. Akay and Atak [5] used the Grey prediction model with a rolling mechanism to forecast electricity demand in Turkey. Hsu and Wang [6, 7] used the Bayesian method to improve GM(1,1) for forecasting the integrated circuit industry.

The Bass diffusion model [8] has been used to develop product life cycle curves and to forecast the sales of the initial purchases of new products. Tseng and Hu [9] combined fuzzy regression with the Bass model to develop a quadratic interval Bass diffusion model. Tsaur [10] used the fuzzy grey regression model to predict the liquid critical display (LCD) television market. Based on empirical data analysis, fuzzy grey regression is capable of accurate forecasting and can give decision makers various scenarios. Wu and Chu [11] used Gompertz, Logistic, Bass, and time-series autoregressive moving average (ARMA) models to forecast mobile telephone subscription in Taiwan. Hsiao and Wang [12] applied the GM(1,1), RGM(1,1), and Bass diffusion models to predict trends in the global copper clad laminate industry.

This study proposes a combined model based on the rolling Grey and Bass diffusion models to forecast the sale of Taiwanese motherboards more accurately. This study is organized as follows: Section 2 reviews the Bass diffusion model and the RGM(1,1); Section 3 introduces a combined model featuring an evolutionary optimization algorithm; in Section 4 we use the combined model to forecast motherboard shipments, comparing its results with those of the RGM(1,1) and Bass diffusion models; finally, we provide a conclusion.

## 2. Bass Diffusion Model and RGM(1,1)

This section presents a discussion on two forecasting models the Bass diffusion and RGM(1,1) models. The Bass diffusion model [13] is given as

$$n(t) = m \times [F(t) - F(t-1)] + \varepsilon, \quad (2.1)$$

where  $F(t) = [1 - e^{-(p+q)t}] / [1 + (q/p) \times e^{-(p+q)t}]$ ,  $n(t)$  = sales at time  $t$ ,  $m$  is the number of eventual adopters,  $F(t)$  is the cumulative distribution of adoptions at time  $t$ ,  $p$  is the coefficient of innovation,  $q$  is the coefficient of imitation, and  $\varepsilon$  is the normally distributed random error term with mean zero and variance  $\sigma^2$ . The adopter's probability density function  $f(t)$  for adoption at time  $t$  is given by

$$f(t) = \frac{\left[ ((p+q)^2 / p) \times e^{-(p+q)t} \right]}{\left[ 1 + (q/p) \times e^{-(p+q)t} \right]^2}. \quad (2.2)$$

Bass [13] used the ordinary least squares (OLS) method to estimate the parameters. Schmittlein and Mahajan [14] used the maximum likelihood estimation (MLE) method to improve the estimation. Srinivasan and Mason [15] used a nonlinear least square estimation (NLS) method to obtain the valid error estimates. Nonlinear models are more difficult to fit

than linear models. Venkatesan and Kumar [16] used genetic algorithms (GAs) to estimate the parameters, and these were consistent with the NLS method.

Grey theory is used for systems that have uncertain and imperfect information [3]. It requires only four data points to construct a prediction model. Grey prediction has three basic operations: accumulated generating operator (AGO), the inverse accumulating operator (IAGO), and the GM. The steps of the RGM(1,1) model are given as follows.

*Step 1.* Original time sequence with  $n$  samples is expressed as

$$X^{(0)} = [x^{(0)}(1), x^{(0)}(2), x^{(0)}(3), \dots, x^{(0)}(n)]. \quad (2.3)$$

An AGO operator is used to convert the original series into monotonically increasing series:

$$X^{(1)} = [x^{(1)}(1), x^{(1)}(2), x^{(1)}(3), \dots, x^{(1)}(n)], \quad (2.4)$$

where  $x^{(1)}(k) = \sum_{i=1}^k x^{(0)}(i)$ .

*Step 2.* The first-order differential equation for the GM(1,1) model is given by

$$\frac{dX^{(1)}}{dt} + a \times X^{(1)} = b, \quad (2.5)$$

where  $t$  denotes the independent variables in the system,  $a$  represents the developed coefficient, and  $b$  is the Grey controlled variable. The parameters of  $a$  and  $b$  can be obtained using the OLS method. Thus, we have

$$\hat{u} = \begin{bmatrix} a \\ b \end{bmatrix} = (B^T B)^{-1} \times B^T \times Y_N, \quad (2.6)$$

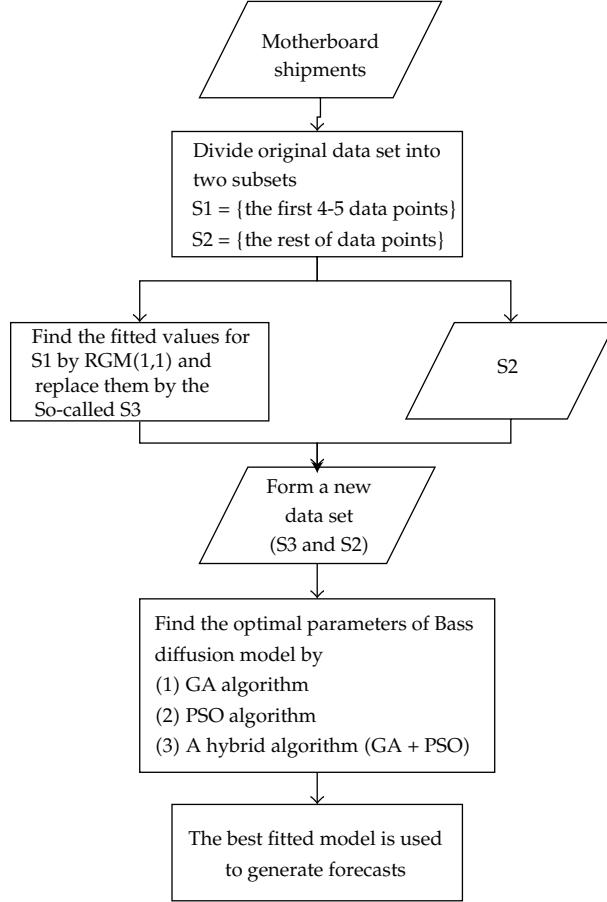
where the accumulated matrix

$$B = \begin{bmatrix} -[P_x^{(1)}(i) + (1-P)x^{(1)}(i+1)] & 1 \\ -[P_x^{(1)}(i+1) + (1-P)x^{(1)}(i+2)] & 1 \\ \vdots & \vdots \\ -[P_x^{(1)}(k-1) + (1-P)x^{(1)}(k)] & 1 \end{bmatrix}, \quad (2.7)$$

$P$  is equal 0.5 in the original model, and  $Y_N = [x^{(0)}(i+1), x^{(0)}(i+2), \dots, x^{(0)}(k)]^T$ .

*Step 3.* The approximate relationship can be obtained by substituting  $\hat{u}$  (determined in the differential equation in Step 2) as follows:

$$\hat{x}^{(1)}(t+1) = \left[ x^{(0)}(1) - \frac{b}{a} \right] \times e^{-at} + \frac{b}{a}. \quad (2.8)$$



**Figure 1:** Procedures of the proposed method.

Supposing that  $\hat{x}^{(1)}(1) = \hat{x}^{(0)}(1)$ , the sequence one-order IAGO is acquired. Thereafter, the sequence can be obtained as  $\hat{x}^{(0)}(t+1) = \hat{x}^{(1)}(t+1) - \hat{x}^{(1)}(t)$ .

Given  $t = 1, 2, \dots, k$ , the sequence of reduction is as follows:

$$\hat{X}^{(0)}(i; k) = (\hat{x}^{(0)}(1), \hat{x}^{(0)}(2), \dots, \hat{x}^{(0)}(k+1)), \quad (2.9)$$

where  $\hat{x}^{(0)}(k+1)$  is the Grey elementary forecasting value for  $\hat{x}^{(0)}(k+1)$ .

### 3. Combined Model

Combining forecasts minimizes errors [18], and many studies have demonstrated their value [19–21]. In this section, we present a combined model based on the Bass diffusion model and the RGM(1,1) (see Figure 1). The three major steps of the proposed combined model are as follows.

*Step 1.* A new data set should be formed. The RGM(1,1) usually provides better fitted values for early periods than does the Bass diffusion model. Next, the original data points are replaced by some predicted values by the RGM(1,1) to form a new data set.

*Step 2.* The optimal parameters of the Bass diffusion model should be found for the original data and a new data set. A hybrid algorithm based on the GA with PSO has been successfully applied to real-world engineering design problems [22, 23]. This study used a hybrid algorithm that couples the GA with PSO to optimize the estimates of the parameters for the Bass diffusion model. We then investigated three algorithms to obtain the parameters of the Bass diffusion model. First, we used a nonlinear algorithm to obtain the initialized estimates of the model. In addition, the confidence intervals for the parameters were used to determine the range of the parameters for PSO and the GA. The descriptions of these three algorithms are as follows.

- (1) GA: the estimated parameters can be obtained easily using Evolver Software [24]. The minimized function is defined as MAPE. Here, the population size is set by 50, and the crossover rate and the mutation rate are set as 0.6 and 0.2, respectively.
- (2) PSO algorithm: we used PSO operators (velocity and position updates) to update the individual with the worst fitness. Clerc and Kennedy [25] created constriction factor  $k$ , improving the ability of PSO to constrain and control velocities. These equations are given by

$$\begin{aligned} V_{\text{id}}^{\text{New}} &= k \times V_{\text{id}}^{\text{old}} + [c_1 \times r_1 \times (p_{\text{id}} - x_{\text{id}}^{\text{old}}) + c_2 \times r_2 \times (p_{\text{gd}} - x_{\text{id}}^{\text{old}})], \\ x_{\text{id}}^{\text{New}} &= x_{\text{id}}^{\text{old}} + V_{\text{id}}^{\text{New}}, \end{aligned} \quad (3.1)$$

where  $k$  is an inertia weight,  $c_1$  and  $c_2$  are two positive constants called acceleration coefficients, and  $r_1$  and  $r_2$  are random, uniformly distributed numbers  $[0, 1]$ . The inertia weight can be obtained by  $k = 2/[2 - c - \sqrt{c^2 - 4c}]$ , where  $c = c_1 + c_2 > 4$ . For example, if  $c = 4.1$ , then  $w = 0.729$ . As  $c$  increases above 4.0,  $k$  becomes smaller. The input parameters are particle size = 20, max iterations = 100,  $w = 0.729$ , and  $c_1 = c_2 = 2.05$ .

- (3) A hybrid algorithm: this hybrid algorithm couples the GA with PSO. First, we used the PSO algorithm to obtain the estimated parameters. Thereafter, the estimated parameters were improved using the GA.

*Step 3.* The fitted Bass diffusion model can be used to generate forecast values.

A common measure of examining the forecasting ability of a model is defined as  $\text{MAPE} = \sum_{t=1}^n (|(A_t - F_t)/A_t|)/n$ , where  $A_t$  is actual value in period  $t$ ,  $F_t$  is forecast value in period  $t$ , and  $n$  is the number of periods used in the calculation. A low MAPE value shows an excellent forecasting ability (Table 1).

#### 4. Application in Motherboard Shipments

The motherboard shipment data from 1998 to 2010 are shown in Table 2. In this study we assumed that the historical data from 1998 to 2009 are known. The holdout period is the

**Table 1:** Criteria for forecasting accuracy [17].

MAPE (%)	Forecasting power
<10	Excellent
10–20	Good
20–50	Reasonable
>50	Incorrect

**Table 2:** Motherboard shipment data [1].

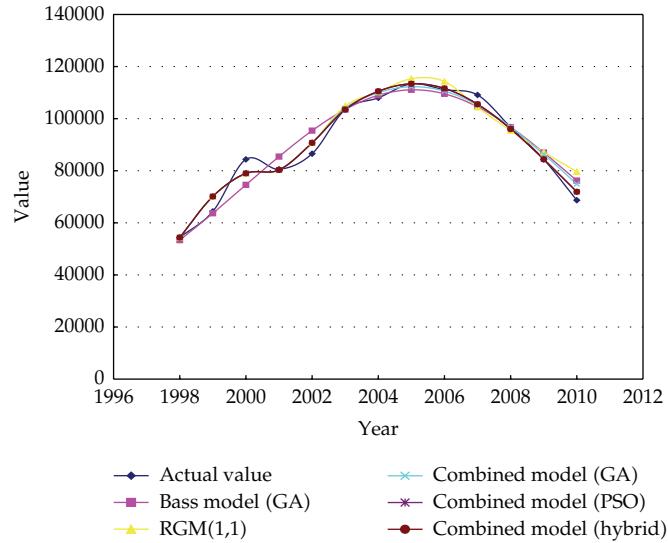
Year	Shipment (unit = thousand)
1998	54371
1999	64378
2000	84372
2001	80565
2002	86554
2003	103509
2004	107987
2005	113354
2006	111117
2007	109097
2008	96743
2009	84374
2010	68687

year 2010. First, the estimated parameters of the Bass diffusion model were obtained using the GA. Second, the rolling interval for the RGM(1,1) was set to five. Thereafter, we obtained eight parameter settings from 1998 to 2009. Finally, the estimated parameters of the combined model were obtained using three evolutionary algorithms (GA, PSO, and a hybrid). All five methods were used to compare the fitted values for the historical periods and the 1-year-ahead forecast.

The parameter values of all five methods are shown in Table 3, and their forecasting results are listed in Table 4. The MAPE values of these five methods from 1998 to 2009 are 3.692% for the Bass diffusion model using the GA, 3.395% for the RGM(1,1), 2.736% for the combined model using the GA, 2.480% for the combined model using PSO, and 2.476% for the combined model using a hybrid algorithm. For the 1-year-ahead forecast for 2010, the MAPE values of the five methods were 10.96% for the Bass diffusion model using the GA, 15.94% for the RGM(1,1), 9.06% for the combined model using the GA, 4.71% for the combined model using PSO, and 4.69% for the combined model using a hybrid algorithm. Our results demonstrate that the proposed combined model using a hybrid algorithm outperformed the other methods for the fitting and forecasting processes in terms of MAPE, as shown in Figure 2.

**Table 3:** Parameter values of all methods.

Year	Method	Parameters
1998–2009	Bass diffusion model with GA	$m = 1512404; p = 0.0321; q = 0.2255$
1998–2002		$a = -0.0776; b = 63299.8$
1999–2003		$a = -0.0748; b = 71247.8$
2000–2004		$a = -0.1042; b = 67427.2$
2001–2005	RGM(1,1)	$a = -0.0806; b = 80632.2$
2002–2006		$a = -0.0255; b = 101302.4$
2003–2007		$a = -0.0010; b = 110072.1$
2004–2008		$a = 0.0473; b = 123165.0$
2005–2009		$a = 0.0902; b = 129711.4$
2003–2009	Combined model with GA	$m = 1483972; p = 0.0311; q = 0.2369$
2003–2009	Combined model with PSO	$m = 1419290; p = 0.0288; q = 0.2594$
2003–2009	Combined model with hybrid	$m = 1420173; p = 0.0289; q = 0.2589$

**Figure 2:** Forecasting results for all models.

## 5. Conclusions

This study presented a combined model that combined the Bass diffusion model with the RGM(1,1) to forecast motherboard shipments. In addition, we investigated evolutionary optimization algorithms to determine the optimal parameters. The results indicate that the combined model using a hybrid algorithm provides excellent MAPE improvement. We conclude that the combined model, using a hybrid algorithm, is suitable for forecasting in the motherboard industry.

Future research will include a modeling comparison of our model and the Support Vector Machine (SVM) model using a DNA optimization method [26].

**Table 4:** MAPE results for all models.

Actual value (1998-2009)	Bass diffusion model with GA			RGM(1,1)			Combined model with GA			Combined model with PSO			Combined model with hybrid		
	Forecast value	APE	Forecast value	APE	Forecast value	APE	Forecast value	APE	Forecast value	APE	Forecast value	APE	Forecast value	APE	
54371	53341.4	1.89%	54371.0	0.00%	54371.0	0.00%	54371.0	0.00%	54371.0	0.00%	54371.0	0.00%	54371.0	0.00%	
64378	63672.5	1.10%	70118.2	8.92%	70118.2	8.92%	70118.2	8.92%	70118.2	8.92%	70118.2	8.92%	70118.2	8.92%	
84372	74572.8	11.61%	78981.5	6.39%	78981.5	6.39%	78981.5	6.39%	78981.5	6.39%	78981.5	6.39%	78981.5	6.39%	
80565	85418.4	6.02%	80332.1	0.29%	80332.1	0.29%	80332.1	0.29%	80332.1	0.29%	80332.1	0.29%	80332.1	0.29%	
86554	95384.7	10.20%	90733.6	4.83%	90733.6	4.83%	90733.6	4.83%	90733.6	4.83%	90733.6	4.83%	90733.6	4.83%	
103509	103536.0	0.03%	104846.9	1.29%	104025.0	0.50%	10355.9	0.05%	103508.9	0.05%	103508.9	0.05%	103508.9	0.05%	
107987	108981.6	0.92%	110227.0	2.07%	109968.1	1.83%	110500.6	2.33%	110509.9	2.33%	110509.9	2.33%	110509.9	2.34%	
113354	111064.0	2.02%	115310.7	1.73%	112262.6	0.96%	113381.9	0.02%	113353.8	0.00%	113353.8	0.00%	113353.8	0.00%	
111117	109518.5	1.44%	114251.3	2.82%	110594.3	0.47%	111646.9	0.48%	111594.8	0.43%	111594.8	0.43%	111594.8	0.43%	
109097	104542.2	4.18%	104393.6	4.31%	105192.5	3.58%	105570.4	3.23%	105510.7	3.29%	105510.7	3.29%	105510.7	3.29%	
96743	96742.4	0.00%	95386.5	1.40%	96766.3	0.02%	96066.0	0.70%	96013.4	0.75%	96013.4	0.75%	96013.4	0.75%	
84374	86987.2	3.10%	87156.6	3.30%	86316.8	2.30%	84412.1	0.05%	84376.4	0.00%	84376.4	0.00%	84376.4	0.00%	
MAPE (1998-2009)		3.692%		3.395%		2.736%		2.480%		2.476%					
Actual value (2010)	68687	76216.5	10.96%	79636.7	15.94%	74907.0	9.06%	71922.0	4.71%	71906.9	4.69%				

APE: absolute percentage error.

## Acknowledgments

The authors gratefully acknowledge the referees of this paper who helped clarify and improve this paper. They are also thankful for the financial support from NSC in Taiwan.

## References

- [1] Market Intelligence Center (MIC), *Taiwan Motherboard Statistics Data*, MIC, Taipei, Taiwan.
- [2] L. C. Hsu, "Applying the grey prediction model to the global integrated circuit industry," *Technological Forecasting and Social Change*, vol. 70, no. 6, pp. 563–574, 2003.
- [3] J. L. Deng, "Introduction to grey system theory," *Journal of Grey System*, vol. 1, no. 1, pp. 1–24, 1989.
- [4] S. C. Chang, H. C. Lai, and H. C. Yu, "A variable P value rolling grey forecasting model for Taiwan semiconductor industry production," *Technological Forecasting and Social Change*, vol. 72, no. 5, pp. 623–640, 2005.
- [5] D. Akay and M. Atak, "Grey prediction with rolling mechanism for electricity demand forecasting of Turkey," *Energy*, vol. 32, no. 9, pp. 1670–1675, 2007.
- [6] L. C. Hsu and C. H. Wang, "Forecasting the output of integrated circuit industry using a grey model improved by the Bayesian analysis," *Technological Forecasting and Social Change*, vol. 74, no. 6, pp. 843–853, 2007.
- [7] L. C. Hsu and C. H. Wang, "Forecasting integrated circuit output using multivariate grey model and grey relational analysis," *Expert Systems with Applications*, vol. 36, no. 2, pp. 1403–1409, 2009.
- [8] F. M. Bass, "Comments on: a new product growth for model consumer durables," *Management Science*, vol. 50, no. 12, pp. 1833–1840, 2004.
- [9] F. M. Tseng and Y. C. Hu, "Quadratic-interval Bass model for new product sales diffusion," *Expert Systems with Applications*, vol. 36, no. 4, pp. 8496–8502, 2009.
- [10] R. C. Tsaur, "Forecasting analysis by using fuzzy grey regression model for solving limited time series data," *Soft Computing*, vol. 12, no. 11, pp. 1105–1113, 2008.
- [11] F. S. Wu and W. L. Chu, "Diffusion models of mobile telephony," *Journal of Business Research*, vol. 63, no. 5, pp. 497–501, 2010.
- [12] Y. Y. Hsiao and F. K. Wang, "Forecasting analysis for global copper clad laminate market," *Journal of Convergence Information Technology*, vol. 7, no. 3, pp. 250–258, 2012.
- [13] F. M. Bass, "A new product growth for model consumer durables," *Management Science*, vol. 50, no. 12, pp. 1825–1832, 2004.
- [14] D. C. Schmittlein and V. Mahajan, "Maximum likelihood estimation for an innovation diffusion model of new product acceptance," *Marketing Science*, vol. 1, pp. 57–78, 1982.
- [15] V. Srinivasan and C. H. Mason, "Nonlinear least squares estimation of new product diffusion models," *Marketing Science*, vol. 5, pp. 169–178, 1986.
- [16] R. Venkatesan and V. Kumar, "A genetic algorithms approach to growth phase forecasting of wireless subscribers," *International Journal of Forecasting*, vol. 18, no. 4, pp. 625–646, 2002.
- [17] C. D. Lewis, *Industrial and Business Forecasting Methods*, Butterworth Scientific, London, UK, 1982.
- [18] J. S. Armstrong, "Combining forecasts: the end of the beginning or the beginning of the end?" *International Journal of Forecasting*, vol. 5, no. 4, pp. 585–588, 1989.
- [19] D. W. Bunn, "Combining forecasts," *European Journal of Operational Research*, vol. 33, no. 3, pp. 223–229, 1988.
- [20] F. K. Wang and K. K. Chang, "Adaptive neuro-fuzzy inference system for combined forecasts in a panel manufacturer," *Expert Systems with Applications*, vol. 37, no. 12, pp. 8119–8126, 2010.
- [21] C. Christodoulos, C. Michalakelis, and D. Varoutas, "On the combination of exponential smoothing and diffusion forecasts: an application to broadband diffusion in the OECD area," *Technological Forecasting and Social Change*, vol. 78, no. 1, pp. 163–170, 2011.
- [22] C. F. Juang, "A hybrid of genetic algorithm and particle swarm optimization for recurrent network design," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 34, no. 2, pp. 997–1006, 2004.
- [23] S. Jeong, S. Obayashi, and Y. Minemura, "Application of hybrid evolutionary algorithms to low exhaust emission diesel engine design," *Engineering Optimization*, vol. 40, no. 1, pp. 1–16, 2008.
- [24] Evolver Software, Version 4.0.2, Palisade Corporation, Newfield, NY, USA, 2000.

- [25] 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.
- [26] X. H. Yang and Y. Q. Li, "DNA optimization threshold autoregressive prediction model and its application in ice condition time series," *Mathematical Problems in Engineering*, vol. 2012, pp. 1–10, 2012.

*Research Article*

## **Adaptive Method for Solving Optimal Control Problem with State and Control Variables**

**Louadj Kahina<sup>1</sup> and Aidene Mohamed<sup>2</sup>**

<sup>1</sup> Department of Mathematics, Faculty of Sciences, Mouloud Mammeri University, Tizi-Ouzou, Algeria

<sup>2</sup> Laboratoire de Conception et Conduite de Systèmes de Production (L2CSP), Tizi-Ouzou, Algeria

Correspondence should be addressed to Louadj Kahina, louadj.kahina@yahoo.fr

Received 29 November 2011; Revised 19 April 2012; Accepted 20 April 2012

Academic Editor: Jianming Shi

Copyright © 2012 L. Kahina and A. Mohamed. 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 problem of optimal control with state and control variables is studied. The variables are: a scalar vector  $x$  and the control  $u(t)$ ; these variables are bonded, that is, the right-hand side of the ordinary differential equation contains both state and control variables in a mixed form. For solution of this problem, we used adaptive method and technology of linear programming.

### **1. Introduction**

Problems of optimal control have been intensively investigated in the world literature for over forty years. During this period, a series of fundamental results have been obtained, among which should be noted the maximum principle [1] and dynamic programming [2, 3]. Results of the theory were taken up in various fields of science, engineering, and economics.

The optimal control problem with mixed variables and free terminal time is considered. This problem is among the most difficult problems in the mathematical theory of control processes [4–7]. An algorithm based on the concept of simplex method [4, 5, 8, 9] so called support control is proposed to solve this problem.

The aim of the paper is to realize the adaptive method of linear programming [8]. In our opinion the numerical solution is impossible without using the computers of discrete controls defined on the quantized axes as accessible controls. This made, it possible to eliminate some analytical problems and reduce the optimal control problem to a linear programming problem. The obtained results show that the adequate consideration of the dynamic structure of the problem in question makes it possible to construct very fast algorithms of their solution.

The work has the following structure. In Section 2, The terminal optimal control problem with mixed variables is formulated. In Section 3, we give some definitions needed in this paper. In Section 4, the definition of support is introduced. Primal and dual ways of its dynamical identification are given. In Section 5, we calculate a value of suboptimality. In Section 6, optimality and  $\varepsilon$ -optimality criteria are defined. In Section 7, there is a numerical algorithm for solving the problem; the iteration consists in two procedures: change of control and change of a support to find a solution of discrete problem; at the end, we used a final procedure to find a solution in the class of piecewise continuous functions. In Section 8, the results are illustrated with a numerical example.

## 2. Problem Statement

We consider linear optimal control problem with control and state constraints:

$$J(x, u) = g(x(t_f)) + \int_0^{t_f} (Cx(t) + Du(t)) dt \longrightarrow \max_{x, u}, \quad (2.1)$$

subject to

$$\begin{aligned} \dot{x} &= f(x(t), u(t)) = Ax(t) + Bu(t), \quad 0 \leq t \leq t_f, \\ x(0) &= x_0, \quad x(t_f) = x_f, \\ x_{\min} &\leq x(t) \leq x_{\max}, \quad u_{\min} \leq u(t) \leq u_{\max}, \quad t \in T = [0, t_f], \end{aligned} \quad (2.2)$$

where  $A, B, C$ , and  $D$  are constant or time-dependent matrices of appropriate dimensions,  $x \in R^n$  is a state of control system (2.1)–(2.2), and  $u(\cdot) = (u(t), t \in T)$ ,  $T = [0, t_f]$ , is a piecewise continuous function. Among these problems in which state and control are variables, we consider the following problem:

$$J(x, u) = c'x + \int_0^{t^*} c(t)u(t) dt \longrightarrow \max_{x, u}, \quad (2.3)$$

subject to

$$Ax + \int_0^{t^*} h(t)u(t) dt, \quad 0 \leq t \leq t_f, \quad (2.4)$$

$$x(0) = x_0, \quad (2.5)$$

$$x_{\min} \leq x(t) \leq x_{\max}, \quad u_{\min} \leq u(t) \leq u_{\max}, \quad t \in T = [0, t_f], \quad (2.6)$$

where  $x \in R^n$  is a state of control system (2.3)–(2.6);  $u(\cdot) = (u(t), t \in T)$ ,  $T = [0, t_f]$ , is a piecewise continuous function,  $A \in R^{m \times n}$ ;  $c = c(J) = (c_j, j \in J)$ ;  $g = g(I) = (g_i, i \in I)$  is an  $m$ -vector;  $c(t)$ ,  $t \in T$ , is a continuous scalar function;  $h(t)$ ,  $t \in T$ , is an  $m$ -vector function;  $u_{\min}, u_{\max}$  are scalars;  $x_{\min} = x_{\min}(J) = (x_{\min_j}, j \in J)$ ,  $x_{\max} = x_{\max}(J) = (x_{\max_j}, j \in J)$  are  $n$ -vectors;  $I = \{1, \dots, m\}$ ,  $J = \{1, \dots, n\}$  are sets of indices.

### 3. Essentials Definitions

*Definition 3.1.* A pair  $v = (x, u(\cdot))$  formed of an  $n$ -vector  $x$  and a piecewise continuous function  $u(\cdot)$  is called a generalized control.

*Definition 3.2.* The constraint (2.4) is assumed to be controllable, that is for any  $m$ -vector  $g$ , there exists a pair  $v$ , for which the equality (2.4) is fulfilled.

A generalized control  $v = (x, u(\cdot))$  is said to be an admissible control if it satisfies constraints (2.4)–(2.6).

*Definition 3.3.* An admissible control  $v^0 = (x^0, u^0(\cdot))$  is said to be an optimal open-loop control if a control criterion reaches its maximal value

$$J(v^0) = \max_v J(v). \quad (3.1)$$

*Definition 3.4.* For a given  $\varepsilon \geq 0$ , an  $\varepsilon$ -optimal control  $v^\varepsilon = (x^\varepsilon, u^\varepsilon(\cdot))$  is defined by the inequality

$$J(v^0) - J(v^\varepsilon) \leq \varepsilon. \quad (3.2)$$

### 4. Support and the Accompanying Elements

Let us introduce a discretized time set  $T_h = \{0, h, \dots, t_f - h\}$  where  $h = t_f/N$ , and  $N$  is an integer. A function  $u(t)$ ,  $t \in T$ , is called a discrete control if

$$u(t) = u(\tau), \quad t \in [\tau, \tau + h], \quad \tau \in T_h. \quad (4.1)$$

First, we describe a method of computing the solution of problem (2.3)–(2.6) in the class of discrete control, and then we present the final procedure which uses this solution as an initial approximation for solving problem (2.3)–(2.6) in the class of piecewise continuous functions.

Definitions of admissible, optimal,  $\varepsilon$ -optimal controls for discrete functions are given in a standard form.

Choose an arbitrary subset  $T_B \subset T_h$  of  $l \leq m$  elements and an arbitrary subset  $J_B \subset J$  of  $m-l$  elements.

Form the matrix,

$$P_B = (a_{ij} = A(I, j), \quad j \in J_B; \quad d(t), \quad t \in T_B), \quad (4.2)$$

where  $d(t) = \int_t^{t+h} h(s)ds$ ,  $t \in T_h$ .

A set  $S_B = \{T_B, J_B\}$  is said to be a support of problem (2.3)–(2.6) if  $\det P_B \neq 0$ .

A pair  $\{v, S_B\}$  of an admissible control  $v(t) = (x, u(t), t \in T)$  and a support  $S_B$  is said to be a support control.

A support control  $\{v, S_B\}$  is said to be primally nonsingular if  $d_{*j} < x_j < d_j^*, j \in J_B; f_* < u(t) < f^*, t \in T_B$ .

Let us consider another admissible control  $\bar{v} = (\bar{x}, \bar{u}(\cdot)) = v + \Delta v$ , where  $\bar{x} = x + \Delta x$ ,  $\bar{u}(t) = u(t) + \Delta u(t)$ ,  $t \in T$ , and let us calculate the increment of the cost functional

$$\Delta J(v) = J(\bar{v}) - J(v) = c' \Delta x + \int_0^{t_f} c(t) \Delta u(t) dt. \quad (4.3)$$

Since

$$A \Delta x + \int_0^z h(t) \Delta u(t) dt = 0, \quad (4.4)$$

then the increment of the functional equals

$$\Delta J(v) = (c' - v' A) \Delta x + \int_0^{t_f} (c(t) - v' h(t)) \Delta u(t) dt, \quad (4.5)$$

where  $v \in R^m$  is called potentials:  $v' = q'_B Q$ ,  $q_B = (c_r j, j \in J_B; q(t), t \in T_B)$ ,  $Q = P_B^{-1}$ ,  $q(t) = \int_t^{t+h} c(s) ds$ ,  $t \in T_h$ .

Introduce an  $n$ -vector of estimates  $\Delta' = v' A - c'$  and a function of cocontrol  $\Delta(\cdot) = (\Delta(t) = v' d(t) - q(t), t \in T_h)$ . With the use of these notions, the value of the cost functional increment takes the form

$$\Delta J(v) = \Delta' \Delta x - \sum_{t \in T_h} \Delta(t) \Delta u(t). \quad (4.6)$$

A support control  $\{v, S_B\}$  is dually nonsingular if  $\Delta(t) \neq 0$ ,  $t \in T_H$ ,  $\Delta_j \neq 0$ ,  $j \in J_H$ , where  $T_H = T_h/T_B$ ,  $J_H = J/J_B$ .

## 5. Calculation of the Value of Suboptimality

The new control  $\bar{v}(t)$  is admissible, if it satisfies the constraints:

$$x_{\min} - x \leq \Delta x \leq x_{\max} - x, \quad u_{\min} - u(t) \leq \Delta u(t) \leq u_{\max} - u(t), \quad t \in T. \quad (5.1)$$

The maximum of functional (4.6) under constraints (5.1) is reached for:

$$\begin{aligned} \Delta x_j &= x_{\min_j} - x_j && \text{if } \Delta_j > 0, \\ \Delta x_j &= x_{\max_j} - x_j && \text{if } \Delta_j < 0, \\ x_{\min_j} - x_j &\leq \Delta x_j \leq x_{\max_j} - x_j && \text{if } \Delta_j = 0, \quad j \in J, \\ \Delta u(t) &= u_{\min} - u(t) && \text{if } \Delta(t) > 0 \\ \Delta u(t) &= u_{\max} - u(t) && \text{if } \Delta(t) < 0 \\ u_{\min} &\leq \Delta u(t) \leq u_{\max} && \text{if } \Delta(t) = 0, \quad t \in T_h, \end{aligned} \quad (5.2)$$

and is equal to

$$\begin{aligned}\beta &= \beta(v, S_B) \\ &= \sum_{j \in J_H^+} \Delta_j (x_j - x_{\min_j}) + \sum_{j \in J_H^-} \Delta_j (x_j - x_{\max_j}) + \sum_{t \in T^+} \Delta(t) (u(t) - u_{\min}) + \sum_{t \in T^-} \Delta(t) (u(t) - u_{\max}),\end{aligned}\tag{5.3}$$

where

$$\begin{aligned}T^+ &= \{t \in T_H, \Delta(t) > 0\}, & T^- &= \{t \in T_H, \Delta(t) < 0\}, \\ J_H^+ &= \{j \in J_H, \Delta_j > 0\}, & J_H^- &= \{j \in J_H, \Delta_j < 0\}.\end{aligned}\tag{5.4}$$

The number  $\beta(v, S_B)$  is called a value of suboptimality of the support control  $\{v, S_B\}$ . From there,  $J(\bar{v}) - J(v) \leq \beta(v, S_B)$ . Of this last inequality, the following result is deduced.

## 6. Optimality and $\varepsilon$ -Optimality Criterion

**Theorem 6.1** (see [8]). *The following relations:*

$$\begin{aligned}u(t) &= u_{\min} & \text{if } \Delta(t) > 0, \\ u(t) &= u_{\max} & \text{if } \Delta(t) < 0, \\ u_{\min} \leq u(t) \leq u_{\max} & \text{if } \Delta(t) = 0, t \in T_h, \\ x_j &= x_{\min_j} & \text{if } \Delta_j > 0, \\ x_j &= x_{\max_j} & \text{if } \Delta_j < 0, \\ x_{\min_j} \leq x_j \leq x_{\max_j} & \text{if } \Delta_j = 0, j \in J,\end{aligned}\tag{6.1}$$

are sufficient, and in the case of non degeneracy, they are necessary for the optimality of control  $v$ .

**Theorem 6.2.** *For any  $\varepsilon \geq 0$ , the admissible control  $v$  is  $\varepsilon$ -optimal if and only if there exists a support  $S_B$  such that  $\beta(v, S_B) \leq \varepsilon$ .*

## 7. Primal Method for Constructing the Optimal Controls

A support is used not only to identify the optimal and  $\varepsilon$ -optimal controls, but also it is the main tool of the method. The method suggested is iterative, and its aim is to construct an  $\varepsilon$ -solution of problem (2.3)–(2.6) for a given  $\varepsilon \geq 0$ . As a support will be changing during the iterations together with an admissible control, it is natural to consider them as a pair.

Below to simplify the calculations, we assume that on the iterations, only primally and dually nonsingular support controls are used.

The iteration of the method is a change of an “old” control  $\{v, S_B\}$  for the “new” one  $\{\bar{v}, \bar{S}_B\}$  so that  $\beta\{\bar{v}, \bar{S}_B\} \leq \beta\{v, S_B\}$ . The iteration consists of two procedures:

- (1) change of an admissible control  $v \rightarrow \bar{v}$ ,
- (2) change of support  $S_B \rightarrow \bar{S}_B$ .

Construction of the initial support control concerns with the first phase of the method and can be performed with the use of the algorithm described below.

At the beginning of each iteration the following information is stored:

- (1) an admissible control  $v$ ,
- (2) a support  $S_B = \{T_B, J_B\}$ ,
- (3) a value of suboptimality  $\beta = \beta(v, S_B)$ .

Before the beginning of the iteration, we make sure that a support control  $\{v, S_B\}$  does not satisfy the criterion of  $\varepsilon$ -optimality.

## 7.1. Change of an Admissible Control

The new admissible control is constructed according to the formulas:

$$\begin{aligned}\bar{x}_j &= x_j + \theta^0 l_j, \quad j \in J, \\ \bar{u}(t) &= u(t) + \theta^0 l(t), \quad t \in T_h,\end{aligned}\tag{7.1}$$

where  $l = (l_j, j \in J, l(t), t \in T_h)$  is an admissible direction of changing a control  $v$ ;  $\theta^0$  is the maximum step along this direction.

### 7.1.1. Construct of the Admissible Direction

Let us introduce a pseudocontrol  $\tilde{v} = (\tilde{x}, \tilde{u}(t), t \in T)$ .

First, we compute the nonsupport values of a pseudocontrol

$$\begin{aligned}\tilde{x}_j &= \begin{cases} x_{\min_j} & \text{if } \Delta_j \geq 0, \\ x_{\max_j} & \text{if } \Delta_j \leq 0, \end{cases} \quad j \in J_H, \\ \tilde{u}(t) &= \begin{cases} u_{\max} & \text{if } \Delta(t) \leq 0, \\ u_{\min} & \text{if } \Delta(t) \geq 0, \end{cases} \quad t \in T_H.\end{aligned}\tag{7.2}$$

Support values of a pseudocontrol  $\{\tilde{x}_j, j \in J_B; \tilde{u}(t), t \in T_B\}$  are computed from the equation

$$\sum_{j \in J_B} A(I, j) \tilde{x}_j + \sum_{t \in T_B} d(t) \tilde{u}(t) = g - \sum_{j \in J_H} A(I, j) \tilde{x}_j + \sum_{t \in T_H} d(t) \tilde{u}(t).\tag{7.3}$$

With the use of a pseudocontrol, we compute the admissible direction  $l$ :  $l_j = \tilde{x}_j - x_j$ ,  $j \in J$ ;  $l(t) = \tilde{u}(t) - u(t)$ ,  $t \in T_h$ .

### 7.1.2. Construct of Maximal Step

Since  $\bar{v}$  is to be admissible, the following inequalities are to be satisfied:

$$x_{\min} \leq \bar{x} \leq x_{\max}; \quad u_{\min} \leq \bar{u}(t) \leq u_{\max}, \quad t \in T_h, \quad (7.4)$$

that is,

$$\begin{aligned} x_{\min} &\leq x_j + \theta^0 l_j \leq x_{\max}, \quad j \in J, \\ u_{\min} &\leq u(t) + \theta^0 l(t) \leq u_{\max}, \quad t \in T_h. \end{aligned} \quad (7.5)$$

Thus, the maximal step  $\theta^0$  is chosen as  $\theta^0 = \min\{1; \theta(t_0); \theta_{j_0}\}$ .

Here,  $\theta_{j_0} = \min \theta_j$ :

$$\theta_j = \begin{cases} \frac{x_{\max_j} - x_j}{l_j} & \text{if } l_j > 0, \\ \frac{x_{\min_j} - x_j}{l_j} & \text{if } l_j < 0, \\ +\infty & \text{if } l_j = 0, \quad j \in J_B, \end{cases} \quad (7.6)$$

and  $\theta(t_0) = \min_{t \in T_B} \theta(t)$ :

$$\theta(t) = \begin{cases} \frac{u_{\max} - u(t)}{l(t)} & \text{if } l(t) > 0, \\ \frac{u_{\min} - u(t)}{l(t)} & \text{if } l(t) < 0, \\ +\infty & \text{if } l(t) = 0, \quad t \in T_B. \end{cases} \quad (7.7)$$

Let us calculate the value of suboptimality of the support control  $\{\bar{v}, S_B\}$  with  $\bar{v}$  computed according to (7.1):  $\beta(\bar{v}, S_B) = (1 - \theta^0)\beta(v, S_B)$ . Consequently,

- (1) if  $\theta^0 = 1$ , then  $\bar{v}$  is an optimal control,
- (2) if  $\beta(\bar{v}, S_B) \leq \varepsilon$ , then  $\bar{v}$  is an  $\varepsilon$ -optimal control,
- (3) if  $\beta(\bar{v}, S_B) > \varepsilon$ , then we perform a change of support.

## 7.2. Change of Support

For  $\varepsilon > 0$  given, we assume that  $\beta(\bar{v}, S_B) > \varepsilon$  and  $\theta^0 = \min(\theta(t_0), t_0 \in T_B; \theta_{j_0}, j_0 \in J_B)$ . We will distinguish between two cases which can occur after the first procedure:

- (a)  $\theta^0 = \theta_{j_0}, j_0 \in J_B$ ,
- (b)  $\theta^0 = \theta(t_0), t_0 \in T_B$ .

Each case is investigated separately.

We perform change of support  $S_B \rightarrow \bar{S}_B$  that leads to decreasing the value of suboptimality  $\beta(v, S_B)$ . The change of support is based on variation of potentials, estimates, and cocontrol:

$$v' = v + \Delta v; \quad \bar{\Delta}_j = \Delta_j + \sigma^0 \delta_j, \quad j \in J, \quad \bar{\Delta}(t) = \Delta(t) + \sigma^0 \delta(t), \quad t \in T_h, \quad (7.8)$$

where  $(\delta_j, j \in J, \delta(t), t \in T_h)$  is an admissible direction of change  $(\Delta, \Delta(\cdot))$  and  $\sigma^0$  is a maximal step along this direction.

### 7.2.1. Construct of an Admissible Direction $(\delta_j, j \in J, \delta(t), t \in T_h)$

First, construct the support values  $\delta_B = (\delta_j, j \in J_B, \delta(t), t \in T_B)$  of admissible direction

(a)  $\theta^0 = \theta_{j_0}$ . Let us put

$$\begin{aligned} \delta(t) &= 0 && \text{if } t \in T_B, \\ \delta_j &= 0 && \text{if } j \neq j_0, \quad j \in J_B, \\ \delta_{j_0} &= 1 && \text{if } \bar{x}_{j_0} = x_{\min_{j_0}}, \\ \delta_{j_0} &= -1 && \text{if } \bar{x}_{j_0} = x_{\max_{j_0}}, \end{aligned} \quad (7.9)$$

(b)  $\theta^0 = \theta(t_0)$ . Let us put

$$\begin{aligned} \delta_j &= 0 && \text{if } j \in J_B, \\ \delta(t) &= 0 && \text{if } t \in \frac{T_B}{t_0}, \\ \delta(t_0) &= 1 && \text{if } \bar{u}(t_0) = u_{\min}, \\ \delta(t_0) &= -1 && \text{if } \bar{u}(t_0) = u_{\max}. \end{aligned} \quad (7.10)$$

Using the values  $\delta_B = (\delta_j, j \in J_B, \delta(t), t \in T_B)$ , we compute the variation  $\Delta v$  of potentials as  $\Delta v' = \delta'_B Q$ . Finally, we get the variation of nonsupport components of the estimates and the cocontrol:

$$\begin{aligned} \delta_j &= \Delta v' A(I, j), \quad j \in J_H, \\ \delta(t) &= \Delta v' d(t), \quad t \in T_H. \end{aligned} \quad (7.11)$$

### 7.2.2. Construct of a Maximal Step $\sigma^0$

A maximal step equals  $\sigma^0 = \min(\sigma_j^0, \sigma_t^0)$  with  $\sigma_j^0 = \sigma_{j_1} = \min \sigma_j, j \in J_H$ ;  $\sigma_t^0 = \sigma(t_1) = \min \sigma(t), t \in T_H$ , where

$$\begin{aligned}\sigma_j &= \begin{cases} -\frac{\Delta_j}{\delta_j} & \text{if } \Delta_j \delta_j < 0, \\ +\infty & \text{if } \Delta_j \delta_j \geq 0, j \in J_H, \end{cases} \\ \sigma(t) &= \begin{cases} -\frac{\Delta(t)}{\delta(t)} & \text{if } \Delta(t) \delta(t) < 0, \\ +\infty & \text{if } \Delta(t) \delta(t) \geq 0, t \in T_H. \end{cases}\end{aligned}\tag{7.12}$$

### 7.2.3. Construct of a New Support

For constructing a new support, we consider the four following cases:

(1)  $\theta^0 = \theta(t_0)$ ,  $\sigma^0 = \sigma(t_1)$ : a new support  $\bar{S}_B = \{\bar{T}_B, \bar{J}_B\}$  has two following components:

$$\bar{T}_B = \frac{T_B}{\{t_0\}} \cup \{t_1\}, \quad \bar{J}_B = J_B, \tag{7.13}$$

(2)  $\theta^0 = \theta(t_0)$ ,  $\sigma^0 = \sigma_{j_1}$ : a new support  $\bar{S}_B = \{\bar{T}_B, \bar{J}_B\}$  has the two following components:

$$\bar{T}_B = \frac{T_B}{\{t_0\}}, \quad \bar{J}_B = J_B \cup \{j_1\}, \tag{7.14}$$

(3)  $\theta^0 = \theta_{j_0}$ ,  $\sigma^0 = \sigma_{j_1}$ : a new support  $\bar{S}_B = \{\bar{T}_B, \bar{J}_B\}$  has two following components:

$$\bar{T}_B = T_B, \quad \bar{J}_B = \frac{J_B}{\{j_0\}} \cup \{j_1\}, \tag{7.15}$$

(4)  $\theta^0 = \theta_{j_0}$ ,  $\sigma^0 = \sigma(t_1)$ : a new support  $\bar{S}_B = \{\bar{T}_B, \bar{J}_B\}$  has two following components:

$$\bar{T}_B = T_B \cup \{t_1\}, \quad \bar{J}_B = \frac{J_B}{\{j_0\}}, \tag{7.16}$$

A value of suboptimality for support control  $\beta(\bar{v}, \bar{S}_B)$  takes the form

$$\beta(\bar{v}, \bar{S}_B) = (1 - \theta^0)\beta(v, S_B) - \alpha\sigma^0, \tag{7.17}$$

where

$$\alpha = \begin{cases} |\tilde{u}(t_0) - \bar{u}(t_0)| & \text{if } \theta^0 = \theta(t_0), \\ |\tilde{x}_{j_0} - \bar{x}_{j_0}| & \text{if } \theta^0 = \theta_{j_0}. \end{cases} \quad (7.18)$$

- (1) If  $\beta(\bar{v}, \bar{S}_B) > \varepsilon$ , then we perform the next iteration starting from the support control  $\{\bar{v}, \bar{S}_B\}$ .
- (2) If  $\beta(\bar{v}, \bar{S}_B) = 0$ , then the control  $\bar{v}$  is optimal for problem (2.3)–(2.6) in the class of discrete controls.
- (3) If  $\beta(\bar{v}, \bar{S}_B) < \varepsilon$ , then the control  $\bar{v}$  is  $\varepsilon$ -optimal for problem (2.3)–(2.6) in the class of discrete controls.

If we would like to get the solution of problem (2.3)–(2.6) in the class of piecewise continuous control, we pass to the final procedure when case 2 or 3 takes place.

### 7.3. Final Procedure

Let us assume that for the new control  $\bar{v}$ , we have  $\beta(\bar{v}, \bar{S}_B) > \varepsilon$ . With the use of the support  $\bar{S}_B$  we construct a quasicontrol  $\hat{v} = (\hat{x}, \hat{u}(t), t \in T)$ ,

$$\begin{aligned} \hat{x}_j &= \begin{cases} x_{\min_j} & \text{if } \Delta_j > 0, \\ x_{\max_j} & \text{if } \Delta_j < 0, \\ \in [x_{\min_j}, x_{\max_j}] & \text{if } \Delta_j = 0, j \in J. \end{cases} \\ \hat{u}(t) &= \begin{cases} u_{\min}, & \text{if } \Delta(t) < 0 \\ u_{\max}, & \text{if } \Delta(t) > 0, \\ \in [u_{\min}, u_{\max}], & \text{if } \Delta(t) = 0, t \in T_h. \end{cases} \end{aligned} \quad (7.19)$$

If

$$A(I, J)\hat{x} + \int_0^{t_f} h(t)\hat{u}(t)dt = g, \quad (7.20)$$

then  $\hat{v}$  is optimal, and if

$$A(I, J)\hat{x} + \int_0^{t_f} h(t)\hat{u}(t)dt \neq g, \quad (7.21)$$

then denote  $T^0 = \{t_i \in T, \Delta(t_i) = 0\}$ , where  $t_i$  are zeros of the optimal cocontrol, that is,  $\Delta(t_i) = 0$ ,  $i = \overline{1, s}$ , with  $s \leq m$ . Suppose that

$$\dot{\Delta}(t_i) \neq 0, \quad i = \overline{1, s}. \quad (7.22)$$

Let us construct the following function:

$$\begin{aligned} f(\Theta) = & A(I, J_B)x(J_B) \\ & + A(I, J_H)x(J_H) + \sum_{i=0}^s \left( \frac{u_{\max} + u_{\min}}{2} - \frac{u_{\max} - u_{\min}}{2} \operatorname{sign} \Delta(t_i) \right) \int_{t_i}^{t_{i+1}} h(t) dt - g, \end{aligned} \quad (7.23)$$

where

$$\begin{aligned} x_j = & \frac{x_{\min_j} + x_{\max_j}}{2} - \frac{x_{\max_j} - x_{\min_j}}{2} \operatorname{sign} \Delta_j, \quad j \in J_H, \quad t_0 = 0, \quad t_{s+1} = t_f, \\ \Theta = & (t_i, i = \overline{1, s}; x_j, j \in J_B). \end{aligned} \quad (7.24)$$

The final procedure consists in finding the solution

$$\Theta^0 = (t_i^0, i = \overline{1, s}; x_j^0, j \in J_B) \quad (7.25)$$

of the system of  $m$  nonlinear equations

$$f(\Theta) = 0. \quad (7.26)$$

We solve this system by the Newton method using as an initial approximation of the vector

$$\Theta^{(0)} = (\bar{t}_i, i = \overline{1, s}; \bar{x}_j, j \in J_B). \quad (7.27)$$

The  $(k+1)^{\text{th}}$  approximation  $\Theta^{(k+1)}$ , at a step  $k+1 \geq 1$ , is computed as

$$\Theta^{(k+1)} = \Theta^{(k)} + \Delta\Theta^{(k)}, \quad \Delta\Theta^{(k)} = -\frac{\partial f^{-1}(\Theta^{(k)})}{\partial \Theta^{(k)}} \cdot f(\Theta^{(k)}). \quad (7.28)$$

Let us compute the Jacobi matrix for (7.26)

$$\frac{\partial f(\Theta^{(k)})}{\partial \Theta^{(k)}} = (A(I, J_B); (u_{\min} - u_{\max}) \operatorname{sign} \Delta(t_i^{(k)}) h(t_i^{(k)}), i = \overline{1, s}) \quad (7.29)$$

As  $\det P_B \neq 0$ , we can easily show that

$$\det \frac{\partial f(\Theta^{(0)})}{\partial \Theta^{(0)}} \neq 0. \quad (7.30)$$

For instants  $t \in T_B$ , there exists a small  $\mu > 0$  that for any  $\tilde{t}_i \in [t_i - \mu, t_i + \mu], i = \overline{1, s}$ , the matrix  $(h(\tilde{t}_i), i = \overline{1, s})$  is nonsingular and the matrix  $\partial f(\Theta^{(k)}) / \partial \Theta^{(k)}$  is also nonsingular if elements  $t_i^{(k)}, i = \overline{1, s}, k = 1, 2, \dots$  do not leave the  $\mu$ -vicinity of  $t_i, i = \overline{1, s}$ .

Vector  $\Theta^{(k*)}$  is taken as a solution of (4.6) if

$$\|f(\Theta^{(k*)})\| \leq \eta, \quad (7.31)$$

for a given  $\eta > 0$ . So we put  $\theta^0 = \theta^{(k*)}$ .

The suboptimal control for problem (2.3)–(2.6) is computed as

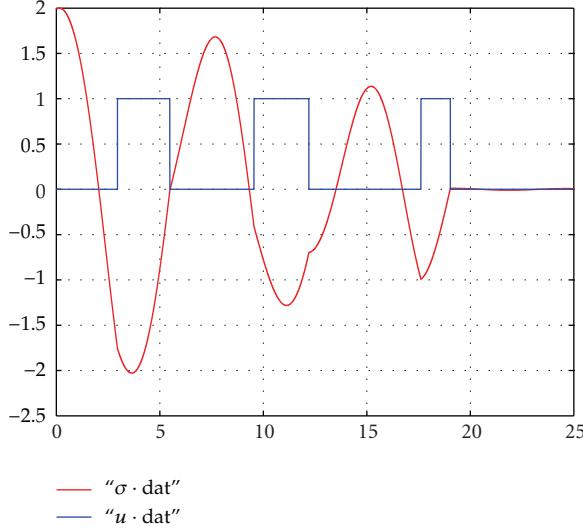
$$\begin{aligned} x_j^0 &= \begin{cases} \bar{x}_j^0, & j \in J_B, \\ \hat{x}_j, & j \in J_H \end{cases} \\ u^0(t) &= \frac{u_{\max} + u_{\min}}{2} - \frac{u_{\max} - u_{\min}}{2} \operatorname{sign} \Delta(t_i^0), \quad t \in [t_i^0, t_{i+1}^0], \quad i = \overline{1, s}. \end{aligned} \quad (7.32)$$

If the Newton method does not converge, we decrease the parameter  $h > 0$  and perform the iterative process again.

## 8. Example

We illustrate the results obtained in this paper using the following example:

$$\begin{aligned} \int_0^{25} u(t) dt &\longrightarrow \min, \\ \dot{x}_1 &= x_3, \\ \dot{x}_2 &= x_4, \\ \dot{x}_3 &= -x_1 + x_2 + u, \\ \dot{x}_4 &= 0.1x_1 - 1.01x_2, \\ x_1(0) &= 0.1, \quad x_2(0) = 0.25, \quad x_3(0) = 2, \quad x_4(0) = 1, \\ x_1(25) &= x_2(25) = x_3(25) = x_4(25) = 0, \\ x_{\min} \leq x &\leq x_{\max}, \quad 0 \leq u(t) \leq 1, \quad t \in [0, 25]. \end{aligned} \quad (8.1)$$



**Figure 1:** Optimal control  $u(t)$  and switching function.

Let the matrix be

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 1 & 0 & 0 \\ 0.1 & -1.01 & 0 & 0 \end{pmatrix}, \quad h(t) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad g = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad (8.2)$$

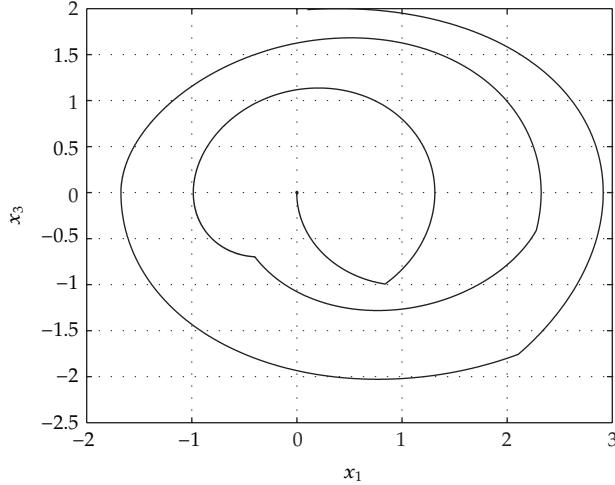
$$x_{\min} = \begin{pmatrix} -4 \\ -4 \\ -4 \\ -4 \end{pmatrix}, \quad x_{\max} = \begin{pmatrix} 4 \\ 4 \\ 4 \\ 4 \end{pmatrix}.$$

We introduce the adjoint system which is defined as

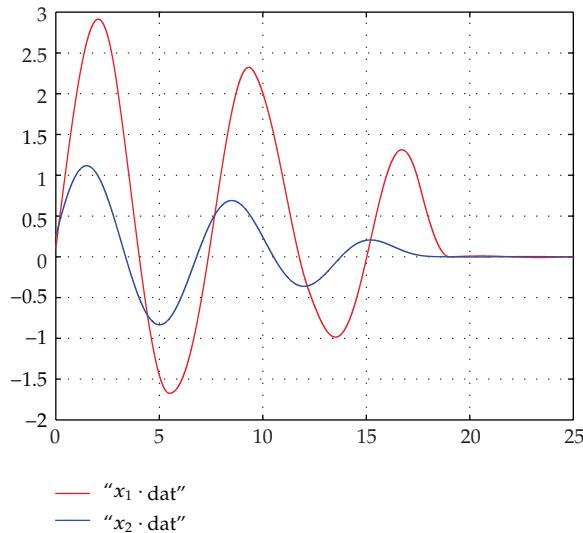
$$\begin{aligned} \psi_1 &= -\psi_3 + 0.1\psi_4, \\ \psi_2 &= \psi_3 - 1.01\psi_4, \\ \psi_3 &= \psi_1, \\ \psi_4 &= \psi_2, \\ \psi_1(t_f) &= 0, \quad \psi_2(t_f) = 0, \quad \psi_3(t_f) = 0, \quad \psi_4(t_f) = 0. \end{aligned} \quad (8.3)$$

Problem (8.1) is reduced to canonical form (2.3)–(2.6) by introducing the new variable  $\dot{x}_5 = u$ ,  $x_5(0) = 0$ . Then, the control criterion takes the form  $-x_5(t_f) \rightarrow \max$ . In the class of discrete controls with quantization period  $h = 25/1000 = 0.0025$ , problem (8.1) is equivalent to LP problem of dimension  $4 \times 1000$ .

To construct the optimal open-loop control of problem (8.1), as an initial support, a set  $T_B = \{5, 10, 15, 20\}$  was selected. This support corresponds to the set of nonsupport zeroes of the cocontrol  $T_{n0} = \{2.956, 5.4863, 9.55148, 12.205, 17.6190, 19.0372\}$ . The problem was solved in 26 iterations, that is, to construct the optimal open-loop control, a support  $4 \times 4$  matrix was



**Figure 2:** Phaseportrait  $x_1(t), x_3(t)$ .



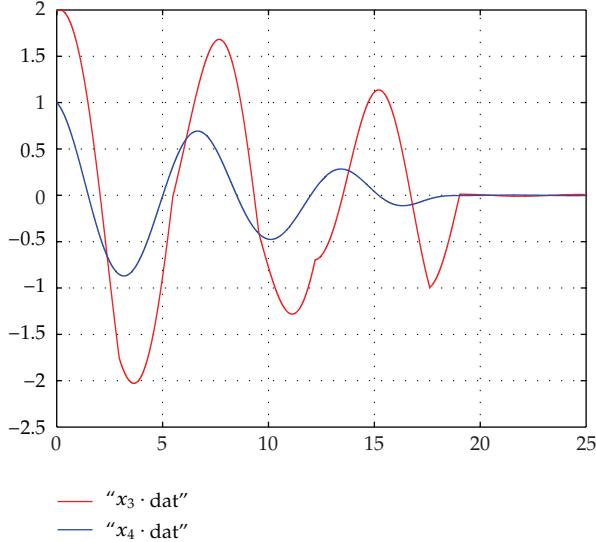
**Figure 3:** Optimal state variables  $x_1(t), x_2(t)$ .

**Table 1**

$h$	Number of iterations	Value of the control criterion	Time
0.25	11	6.624333	2.72
0.025	18	6.602499	2.85
0.001	32	6.602050	3.33

changed 26 times. The optimal value of the control criterion was found to be equal to 6.602050 in time 2.92.

Table 1 contains some information on the solution of problem (8.1) for other quantization periods.



**Figure 4:** Optimal state variables  $x_3(t), x_4(t)$ .

Of course, one can solve problem (8.1) by LP methods, transforming the problem (4.6)–(7.8). In doing so, one integration of the system is sufficient to form the matrix of the LP problem. However, such “static” approach is concerned with a large volume of required operative memory, and it is fundamentally different from the traditional “dynamical” approaches based on dynamical models (2.3)–(2.6). Then, problem (2.3)–(2.6) was solved.

In Figure 1, there are control  $u(t)$  and switching function for minimum principle. In Figure 2, there is phaseportrait  $(x_1, x_3)$  for a system (8.1). In Figure 3, there are state variables  $x_1(t), x_2(t)$  for a system (8.1). In Figure 3, state variables  $x_3(t), x_4(t)$  for a system (8.1). In Figure 4, state variables  $x_1(t), x_2(t)$  for a system (8.1).

## References

- [1] L. S. Pontryagin, V. G. Boltyanski, R. Gamkrelidze, and E. F. Mischenko, *The Mathematical Theory of Optimal Processes*, Interscience Publishers, New York, NY, USA, 1962.
- [2] R. E. Bellman, *Dynamic Programming*, Princeton University Press, Princeton, NJ, USA, 1963.
- [3] R. E. Bellman, I. Glicksberg, and O. A. Gross, *Some Aspects of the Mathematical Theory of Control Processes*, Rand Corporation, Santa Monica, Calif, USA, 1958, Report R-313.
- [4] N. V. Balashevich, R. Gabasov, and F. M. Kirillova, “Calculating an optimal program and an optimal control in a linear problem with a state constraint,” *Computational Mathematics and Mathematical Physics*, vol. 45, no. 12, pp. 2030–2048, 2005.
- [5] R. Gabasov, F. M. Kirillova, and N. V. Balashevich, “On the synthesis problem for optimal control systems,” *SIAM Journal on Control and Optimization*, vol. 39, no. 4, pp. 1008–1042, 2000.
- [6] R. Gabasov, N. M. Dmitruk, and F. M. Kirillova, “Decentralized optimal control of a group of dynamical objects,” *Computational Mathematics and Mathematical Physics*, vol. 48, no. 4, pp. 561–576, 2008.
- [7] N. V. Balashevich, R. Gabasov, and F. M. Kirillova, “Numerical methods of program and positional optimization of the linear control systems,” *Zh Vychisl Mat Mat Fiz*, vol. 40, no. 6, pp. 838–859, 2000.
- [8] R. Gabasov, *Adaptive Method of Linear Programming*, University of Karsruhe, Institute of Statistics and Mathematics, Karsruhe, Germany, 1993.
- [9] L. Kahina and A. Mohamed, “Optimization of a problem of optimal control with free initial state,” *Applied Mathematical Sciences*, vol. 4, no. 5–8, pp. 201–216, 2010.

## Research Article

# A Hybrid Network Model to Extract Key Criteria and Its Application for Brand Equity Evaluation

**Chin-Yi Chen and Chung-Wei Li**

*Department of Business Administration, Chung Yuan Christian University, 200 Chung Pei Road, Zhongli 32023, Taiwan*

Correspondence should be addressed to Chung-Wei Li, research.cwli@gmail.com

Received 23 February 2012; Revised 16 May 2012; Accepted 21 May 2012

Academic Editor: Jung-Fa Tsai

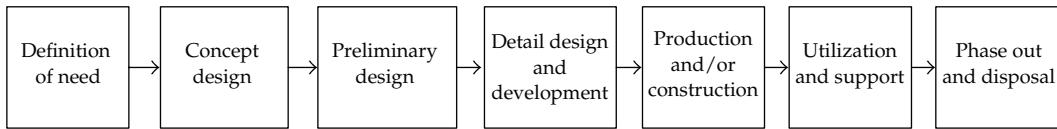
Copyright © 2012 C.-Y. Chen and C.-W. Li. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Making a decision implies that there are alternative choices to be considered, and a major challenge of decision-making is to identify the adequate criteria for program planning or problem evaluation. The decision-makers' criteria consists of the characteristics or requirements each alternative must possess and the alternatives are rated on how well they possess each criterion. We often use criteria developed and used by different researchers and institutions, and these criteria have similar means and can be substituted for one another. Choosing from existing criteria offers a practical method to engineers hoping to derive a set of criteria for evaluating objects or programs. We have developed a hybrid model for extracting evaluation criteria which considers substitutions between the criteria. The model is developed based on Social Network Analysis and Maximum Mean De-Entropy algorithms. In this paper, the introduced methodology will also be applied to analyze the criteria for assessing brand equity as an application example. The proposed model demonstrates that it is useful in planning feasibility criteria and has applications in other evaluation-planning purposes.

## 1. Introduction

System engineering is an interdisciplinary field of engineering focusing on how complex engineering projects should be designed and managed over their life cycles, and it overlaps with both technical and human-centered disciplines such as control engineering, industrial engineering, organizational studies, and project management [1]. The principles of system engineering provided system thinking and have been applied to most projects and industry fields. In the system life-cycle process, as Figure 1 illustrates, system assessment-analysis and evaluation have to be undertaken in distinct phases.

When we want to execute an evaluation project, determining adequate criteria is critical to achieve the evaluation. Traditionally, criteria are derived through discussion with engineers, experts, researchers and, especially in the field of business administration, by



**Figure 1:** System assessment-analysis and evaluation in the system life-cycle process.

finding out about an object through consultation with customers. In this process, deciding upon consistent evaluation criteria is time-consuming work. In many situations when the assessing objects are similar, existing criteria could be referred to. If the criteria which can be referred to are numerous, it is necessary to choose the most appropriate basis of the benchmark assessment. After an in-depth literature review about evaluation criteria, a researcher established an adequacy criteria algorithm that reasonably satisfied the request of "face validity" or "content validity."

As an example application, brand equity was chosen as it is one of the important intangible assets that can bring competitive advantage for many enterprises. However, until now, measuring the value of a brand has been relatively abstractive and subjective and there are many different methods to explore the potential value of the brand in academia and industry. Many criteria have similar meanings and can be partially or totally substituted for one another. Based on the assumption that "a criterion can be substituted partially or totally by another criterion", in this paper, we develop a hybrid model for choosing the adequate evaluation criteria. By using the methods of the Social Network Analysis (SNA) and the Maximum Mean De-Entropy (MMDE) algorithm, the degree of substitutability of existing reference criteria will be judged to derive a criteria list for evaluation. In this paper, the issue of the assessment of brand equity will be addressed as an example to demonstrate the application of the proposed research model in the planning of other evaluation projects.

The rest of this paper is organized as follows: Section 2 describes the issues associated with choosing the criteria, the theories of SNA and MMDE methods, and explains how the model for this study was constructed. We also use an example, choosing the criteria to evaluate the value of brand equity, to illustrate the steps of our model and the applications of the model are discussed in Section 3. Finally, in Section 4, we will discuss the advantages of the proposed model and the feasibility of its application. We draw conclusions and offer some discussion related to future work in Section 5.

## 2. Criterion, Algorithm, and Hybrid Model

In this section, the importances of criteria for evaluation, especially for brand equity analysis, are described. Then we will explain the methods we applied to construct our hybrid model, and its feasibility in the work of extracting criteria.

### 2.1. Criteria for Brand Equity Evaluation

A criterion is the standard or test by which individual things or people may be compared and judged. A fair and just evaluation criteria set ensures the performance of an evaluation project. It is not easy to perform those processes in a completely objective way, especially in

the assessment of intangible assets such as technology, patents, or brand equity or concerning the characteristics of invisibility and abstractness, and as such, those characteristics also have a great influence on the assessment framework itself.

There are many diverse models for brand equity available in the academic world and in practice. These two sectors have already proposed many different assessment frameworks which constitute various dimensions and criteria, and various analytical perspectives accompanied by various dimensions and criteria in the assessment framework. However, aside from the complexity of the implementation of the assessment process in reality, the naming or definitions of dimensions and criteria are sometimes so similar making distinguishing them with certainty very difficult. Moreover, the existence of such ambiguity in the framework may bring about the issue of double counting resulting in bias of the assessment result. Therefore, it can be seen that developing a series of processes to arrange and select proper dimensions and criteria within the established framework to cover the full meaning of the evaluation structure and to prevent measurement bias will be crucial to building an appropriate evaluation model.

Hence, we determine the full dimensions to be applied to well-known brand equity assessment models and use the research methodology addressed in this paper to extract appropriate dimensions, to propose a reasonable and practical assessment framework of brand equity and hope to provide a reference for establishing relative models in the future.

## 2.2. Research Methods

### 2.2.1. Social Network Analysis

Social network analysis presumes social relationships in terms of network theory consisting of nodes and ties [2]. Graph theory is the main field of mathematics to provide social network as a model of a system consisting of a set of nodes with ties between them. Nodes are considered as the individual participants within the networks, and ties are the relationships between the nodes. The results of the analysis are graph-based structures for explaining the whole interrelated group [3]. The simplest social network is a map of specified ties, such as friendship, between the nodes being studied. Thus, the nodes to which an individual is connected are the social contacts of that individual. The network can also be used to measure social capital, that is, the value that an individual gets from the social network. These concepts are often displayed in a social network diagram, where nodes are the points and ties are the lines.

SNA has emerged as a key technique in modern sociology. It also has gained a significant following in anthropology, biology, communication studies, economics, geography, information science, organizational studies, social psychology, and sociolinguistics [4–8], and it has become a popular topic of speculation and study [2]. In this study, the SNA was used to view the referred criteria and substitutions between them (links of the social relations between nodes), which are useful in determining relevance, impact direction, and, more importantly, the degree of substitution on the criteria. Using this method in this study, we also can review the crucial role of the key criteria.

In the theory of social network, there are three properties that will be used in our study: “walk,” “distance,” and “centrality” of position [3, 9, 10]. For a network, the adjacencies of the network refer to the connections from one actor to another. Shown in Figure 2(a), two persons

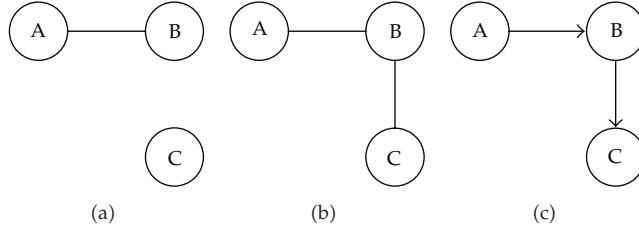


Figure 2

A and B have a relationship, between them but C is an isolated point with no relationship between C and the other two. If the relationship is not directed, then the relation is symmetric. In Figure 2(b), there is a relationship between A and B, and a relationship between B and C, but there is no relationship between A and C. If the relationship is not directed, the attribute of *transitive* is not suitable for analysis. A directed relation, such as Figure 2(c), will have the attribute of *transitive* and *asymmetric*.

A *walk* is an alternating sequence of incident nodes and lines. A walk begins and ends with nodes. The length of a walk is the number of occurrences of lines in it. Because some nodes and some lines may be included more than once, the definition of a *path* as a walk in which all nodes and all lines are distinct is necessary for our paper. If a node  $x$  can walk to another node  $y$ , we can say that node  $x$  and node  $y$  are connected. It is likely that there are several different connections between a given pair of nodes and that these connections differ in length. A connection with the shortest distance between two nodes is referred to as a *geodesic*, or the distance between two nodes.

“*Centrality*” is a concept which is used in social network analysis to identify the “importance” of nodes in a social network. There are some indices, such as “*degree*,” “*closeness*,” and “*betweenness*,” which have been commonly used as the indices of centrality [11]. The definitions of these indices attempt to quantify the importance of an individual actor/node embedded in a network. Now, many indices have been established for identifying the important of a node. In this paper, we use the index “*Bonacich Power Centrality*” as the centrality scores of network nodes.

The original degree centrality approach argues that actors who have more connections are more likely to be powerful because they can directly affect a greater number of other actors. Bonacich proposed a modification of the degree centrality approach to include the concept that the same degree does not necessarily make actors equally important [12, 13]. A node is likely to be more influential if it is connected to another central node which can quickly reach a lot of other nodes with its message. But if the nodes that you are connected to are well connected, they are not dependent on you and that means you are not so important although there are a lot nodes that in fact connect with you. Connected to others makes a node central, but not powerful. One node being connected to others that are not well connected makes one powerful, because these other actors are dependent on you.

Bonacich proposed that both centrality and power were a function of the connections of the nodes in one’s neighborhood. The more connections the nodes in your neighborhood have, the more central you are. The fewer the connections the actors in your neighborhood have, the more powerful you are. Bonacich’s power centrality measure, or BP score, is defined as (2.1) where  $\mathbf{A}$  is the graph adjacency matrix,  $\mathbf{1}$  is column vector of 1. The  $\beta$  is the

“attenuation parameter,” and the  $\alpha$  is a scaling parameter which is set as the sum of squared scores equal to the number of nodes [14]:

$$C_{BP}(v) = \alpha(\mathbf{I} - \beta\mathbf{A})^{-1}\mathbf{A} \cdot \mathbf{1}. \quad (2.1)$$

For any criterion with influence from node  $x$  to node  $y$  through node  $z$ , the influence from  $x$  to node  $z$  includes direct and indirect influence recursively. The BP score is the notion that the power of a node is recursively defined by the sum of the power of its alter. The nature of the recursion involved is then controlled by the power exponent: positive values imply that vertices become more powerful as their alters become more powerful (as occurs in cooperative relations), while negative values imply that nodes become more powerful only as their alters become weaker (as occurs in competitive or antagonistic relations). The magnitude of the exponent indicates the tendency of the effect to decay across long walks; higher magnitudes imply slower decay. The BP score is suitable for this paper because of the purpose of finding out the degree of substitution between criteria.

### 2.2.2. Maximum Mean De-Entropy (MMDE) Algorithm

We used the MMDE algorithm to determine the threshold value for delineating the network diagram that was derived from the SNA. The MMDE algorithm was developed from the basis of entropy theory. Entropy is a physical measurement of thermal-dynamics and has become an important concept in the social sciences. In information theory, entropy is used to measure the expected information content of certain messages and is a criterion for measuring the amount of “uncertainty” represented by a discrete probability distribution.

*Definition 2.1.* Let a random variable with  $n$  elements denoted as  $X = \{x_1, x_2, \dots, x_n\}$ , with a corresponding probability  $P = \{p_1, p_2, \dots, p_n\}$ , then one defines the entropy,  $H$ , of  $X$  as follows:

$$H(p_1, p_2, \dots, p_n) = - \sum p_i \lg p_i, \quad (2.2)$$

subject to constraints

$$\sum_{i=1}^n p_i = 1, \quad (2.3)$$

$$p_i \lg p_i = 0 \quad \text{if } p_i = 0.$$

The function “lg” means the logarithms which are taken to an arbitrary but fixed base. The value of  $H(p_1, p_2, \dots, p_n)$  is the largest when  $p_1 = p_2 = \dots = p_n$ , and we denote this largest entropy value as  $H(1/n, 1/n, \dots, 1/n)$ . Now we will define another measure for the decreased level of entropy: *de-entropy*.

*Definition 2.2.* For a given finite discrete scheme of  $X$ , the *de-entropy* of  $X$  is denoted as  $H_n^D$  and defined as

$$H_n^D = H\left(\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n}\right) - H(p_1, p_2, \dots, p_n). \quad (2.4)$$

By Definition 2.2, the value of  $H^D$  is equal to or larger than 0. Unlike entropy, which is used for the measure of uncertainty, the  $H_n^D$  can explain the amount of useful information derived from a specific dataset, which reduces the “uncertainty” of the information. We define the de-entropy for the purpose of searching the threshold value in order to assess the effect of information content when adding a new node to an existing impact-relations map. By Definition 2.1, Formula (2.5) can be proven (the proof can be found in [15]):

$$H_n = H\left(\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n}\right) \leq H\left(\frac{1}{n+1}, \frac{1}{n+1}, \dots, \frac{1}{n+1}\right) = H_{n+1}. \quad (2.5)$$

Based on Definitions 2.1 and 2.2, the MMDE algorithm is developed to obtain a threshold value for delineating a network diagram. This algorithm can be used to derive a set of dispatch-nodes, the factors that strongly dispatch influences to others, and a set of receive-nodes, which are easily influenced by another factor. According to these two sets and a unique threshold value, we can obtain an influence network of criteria.

We propose the maximum mean de-entropy (MMDE) algorithm to find a threshold value for delineating the impact relations between criteria. The MMDE algorithm has some properties that differ from the traditional method of determining a network. First, the MMDE algorithm serves mainly to decide the “node” rather than the “network,” and this is helpful for understanding a problem in that it decreases the uncertainty of information. Second, the MMDE algorithm considers the properties of the dispatch influence and the received influence of a node, and this is useful to the analyst in determining the “nodes” or criteria, which are easily influenced by other factors. The MMDE algorithm can also obtain a unique threshold value, which is helpful in solving the problems that a researcher may confront regarding the selection of a consistent threshold value and is decided by searching a suitable criteria set. The theory and steps of this algorithm are described In Li and Tzeng (2009) [16, 17], and we summarize the steps of the MMDE as follows.

*Step 1.* Transforming the relation matrix into an ordered triplets set.

*Step 2.* Taking the second element from the ordered triplets set to establish a dispatch-node set.

*Step 3.* Calculating the mean de-entropy of dispatch-node set.

*Step 4.* Finding the maximum mean de-entropy.

*Step 5.* Similar to Steps 2 to 4, an ordered receive-node set and a maximum mean de-entropy receive-node set can be derived.

*Step 6.* Finding the threshold value.

### **2.3. The Hybrid Model**

The purpose of our proposed model is to resolve the problem when researchers are faced with excessive available criteria with a considerable degree of substitution, by using a mathematical algorithm to obtain key criteria following a preliminary judgment of degree of substitution. Hence, first, the experts' judgments about the degree of substitution between each other should be obtained. The measure of degree of substitution should be recognized as distinct. For example, the value 1 means a criterion can almost be totally substituted/replaced by another and 2 means that a criterion can be partially substituted/replaced. The value of scale, or the degree of substitution, can be considered as "distance" between two criteria. This allows us to determine the number and lengths of geodesics between all nodes. With edge values being interpreted as distances, where edge values reflect proximity or tie strength, we can construct the substitution degree matrix. Based on this substitution degree matrix, there are steps to achieve the purpose of the purposed hybrid model.

First, we use the BP score to find the "powerful" criteria. Because the BP scores imply both centrality and power of the nodes in one's neighborhood, we can separate the original criteria into the powerful criteria set, whose element's alter will influence the weaker criterion, and the weaker set. The elements in the weaker set are considered as replaceable criteria. But the BP score was calculated based on the connections, whose value is binary; however, the degree of substitution between the elements in the powerful set also should be considered.

Secondly, we use the MMDE method to obtain a substitution network and find the criteria which can be substituted by another. When we apply the MMDE algorithm to the substitution degree matrix, there were some steps to obtain the maximum mean de-entropy values. After the substitution degree matrix is normalized, a continuous decrease of the indirect effects of substitution along the powers of matrix  $D$ , for example,  $D^2, D^3, \dots, D^\infty$ , guarantees convergent solutions to the matrix inversion, a process similar to an absorbing Markov chain matrix. We can also use the MMDE algorithm to determine the threshold value and draw the network which can explain the replaceable criteria [18].

In this paper, the SNA was used to view the referred criteria and substitutions between them (links of the social relations between nodes), which are useful in determining relevance, impact direction, and, more importantly, the degree of substitution on the criteria. Using this method in this paper, we also can review the crucial role of the key brand equity evaluation criteria. The usage of the Bonacich power score was to group the similar criteria by the metric of each criterion (node) with the others. This metric is a nondirectional relations' measure, which is why we applied the MMDE tool to find the evaluation structure with consideration of the direction of relationships between criteria.

After we obtain the results from the SNA and MMDE algorithms, we can exclude the weaker criteria and replaceable criteria. The remainders in the powerful criteria set, which are important and cannot be substituted, are the criteria that meet the objectives of the proposed hybrid model. We summarize the steps of the proposed model in Figure 3.

## **3. The Case for Model Application**

Brand equity is an important intangible asset since it can bring competitive advantage for many enterprises. However, until now, measuring the value of a brand is relatively abstractive and subjective and many different methods have been developed to explore

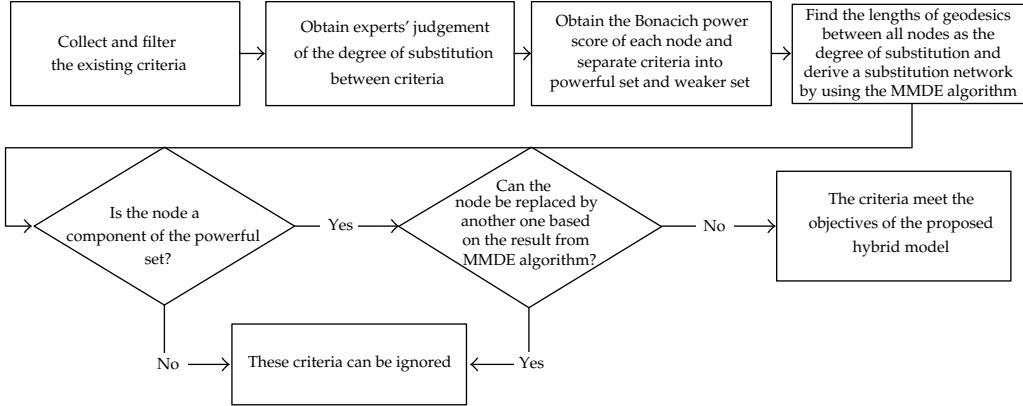


Figure 3: The flow of the proposed hybrid model.

the potential value of brand, both from academic or practical viewpoints. Generally, from an analytical point of view, those methods can be divided into financial, marketing, and a combination of perspectives.

Briefly, the financial perspective applies the real data from the financial statements of the company to evaluate the brand equity; the marketing perspective tends to extend the meaningfulness of a brand to customers, that is, consumers' feelings toward a brand and its potential influences on a company's profit, is usually divided into several dimensions to explore a brand's potential value to a company; the combination perspective integrates the two former's concepts and is viewed both from financial and marketing perspectives which are important but to a different extent. The two methods should be considered simultaneously in the measurement process.

In practice, there are many different models to evaluate brand equity, the most famous and most reliable models named Interbrand, BBDO, and HIROSE. The model developed by Interbrand, a British global branding consultancy company, covers market segmentation, financial analysis, demand, and competition analysis and takes both marketing and financial perspectives into consideration. It can be viewed as a conscientious measurement structure of brand equity, but the calculating process of the model has not been completely disclosed, preventing its widespread adoption [19].

The BBDO model, developed by a worldwide advertising agency belonging to the Omnicom group, puts market quality, dominance of relevant market, international orientation, brand status, and monetary basis into the measurement process. The data for the model are relatively easy to collect, but the calculation is comparatively complex, especially the repetition of meanings between dimensions, which could result in the possibility of double counting [20, 21], since these three models all reflect the marketing and financial aspects that previous researchers consider important when it comes to evaluating the brand equity. Moreover, these three models are widely applied in practice and accepted by many stock exchanges in the U.S. and Europe. Therefore, considering the theoretical integrity and representativeness, we chose these three models as the empirical cases to analyze.

HIROSE is constructed by the Japanese Government to reflect the development background and put more emphasis on industrial competitiveness [22]. Therefore, this model includes the three drivers of prestige, loyalty, and expansion and is inclined to include

**Table 1:** Fifteen criteria used for assessing the value of brand equity.

Notation	Criteria description
$x_1$	Brand leadership
$x_2$	Brand stability
$x_3$	Market
$x_4$	International reach of brand
$x_5$	Brand trend
$x_6$	Marketing support
$x_7$	Legal protection of brand
$x_8$	Market quality
$x_9$	Dominance of relevant market
$x_{10}$	International orientation
$x_{11}$	Brand status
$x_{12}$	Monetary basis
$x_{13}$	Prestige driver
$x_{14}$	Loyalty driver
$x_{15}$	Expansion driver

financial statements and can be regarded as a reliable value measurement index. Even so, focus on the financial statements also limits its application scope in practice.

To conduct our research, a list of 15 criteria was created after comparison of these three models, shown as Table 1. After we interviewed senior technical personnel and marketing managers to determine the degree of substitution between each other, we can obtain the matrix as shown in Table 2. In Table 2,  $a_{ij}$  is the element located at row  $i$  and column  $j$ . If  $a_{ij}$  is 0, it means that the criterion  $i$  has no interrelationship with criterion  $j$ . If  $a_{ij}$  is 1, it means that the criterion  $i$  can almost totally substitute/replace criterion  $j$ . If  $a_{ij}$  is 2, it means that the criterion  $i$  can partially substitute/replace criterion  $j$ . The values 0, 1, and 2 are ordinal scale by the means of distance.

Our hypothesis was analyzed using the SNA method. With the directed line that implies the degree of substitution from one node to another, we divided the support measures into a group that dispatches influence and a second group that receives influence so that we could understand the influence relationships better. The purposes of the SNA enquiry in this research, with the expert's knowledge for contributing to a deeper comprehension of the criteria, are the analysis of the structure and interrelationships of the criteria and the identification of the key feasible and efficient criteria for evaluating the value of brand equity.

### 3.1. Social Network Analysis Results

In this step, we applied the software "sna," an R language package for social network analysis. This tool's functions include node and graph-level indices, structural distance and covariance methods, structural equivalence detection,  $p^*$  modeling, random graph generation, and 2D/3D network visualization [23, 24]. After we applied the tools to Table 2, we obtained some key indices for each criteria (Table 3).

According to the normalized BP score of each node, we can find that twelve criteria have positive scores and three criteria have negative scores, as shown as Figure 4. There are twelve powerful criteria with positive BP scores but three criteria in the weaker criteria set.

**Table 2:**  $15 \times 15$  degree of substitution matrices between criteria.

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	$x_{11}$	$x_{12}$	$x_{13}$	$x_{14}$	$x_{15}$
$x_1$	1	2	2	2	2	0	0	2	1	0	2	2	2	2	0
$x_2$	2	1	0	0	2	2	0	0	2	0	2	2	2	2	0
$x_3$	2	0	1	2	2	0	0	2	2	0	0	2	2	2	0
$x_4$	2	0	2	1	2	0	0	2	2	1	0	0	2	0	2
$x_5$	2	2	2	0	1	0	0	2	2	0	0	2	2	2	0
$x_6$	2	2	0	0	2	1	0	2	2	0	2	2	2	2	0
$x_7$	0	2	0	0	0	0	1	0	0	0	2	0	0	0	2
$x_8$	2	0	2	0	2	0	0	1	2	0	2	2	2	2	0
$x_9$	2	0	0	0	0	0	0	2	1	0	2	2	2	2	0
$x_{10}$	0	0	0	2	0	0	0	0	0	1	0	0	0	0	2
$x_{11}$	1	1	0	0	0	0	0	2	2	0	1	2	2	1	2
$x_{12}$	2	2	2	2	2	2	2	2	2	2	1	2	2	2	2
$x_{13}$	2	2	0	0	0	0	0	2	1	0	2	2	1	2	0
$x_{14}$	2	1	0	0	2	0	0	2	2	0	2	2	0	1	0
$x_{15}$	0	0	0	2	0	0	0	0	0	1	0	2	0	0	1

**Table 3:** Key indices of the network of support measures.

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	$x_{11}$	$x_{12}$	$x_{13}$	$x_{14}$	$x_{15}$
Degree	21	14	10	10	16	4	2	20	20	4	18	22	20	19	10
Betweenness	0.08	0.08	0.01	0.12	0.03	0.00	0.00	0.04	0.02	0.00	0.07	0.45	0.03	0.02	0.07
Closeness	0.78	0.70	0.70	0.64	0.70	0.74	0.54	0.70	0.64	0.45	0.70	1.00	0.67	0.67	0.56
BP score (normalized)	0.02	-0.05	0.60	0.11	-0.04	-0.02	0.05	0.11	0.19	0.13	0.18	0.04	0.07	0.07	0.09

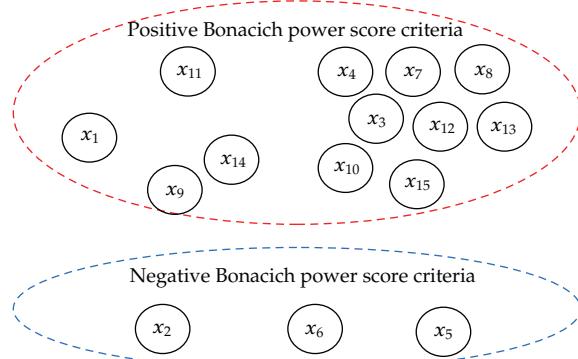
### 3.2. Usage of the MMDE Algorithm

Based on Table 2 and the degree of substitution between criteria judged by experts, we applied the MMDE algorithm to these data. When applied to the matrix, there were some steps to obtain the de-entropy values matrix, shown as Table 4.

Following the steps of MMDE algorithm, we can obtain the threshold value, 0.379. Based on this threshold value, we derived the substitution networks, as shown in Figure 5. According to Figure 5, we found that criteria  $x_1$ ,  $x_2$ ,  $x_9$ , and  $x_{14}$  are all influenced by criterion  $x_{11}$  directly or indirectly. In other words, although  $x_1$ ,  $x_9$ , and  $x_{14}$  are elements of the powerful set with positive BP scores, all these three criteria can be substituted by criterion  $x_{11}$ .

### 3.3. Key Criteria

After we obtain the results from the SNA and MMDE algorithms, we analyzed the results according to the flow chart shown in Figure 3. At first, we exclude the weaker criteria,  $x_2$ ,  $x_6$ , and  $x_5$ , then we can exclude the replaceable criteria,  $x_1$ ,  $x_9$ , and  $x_{14}$ . The nine remainders in the powerful criteria set, which are important and cannot be substituted, are the criteria that meet the objectives of the proposed hybrid model, as shown as Figure 6. The original fifteen criteria can be replaced by nine criteria, as the result of the proposed model in this paper.



**Figure 4:** Criteria separated into positive and negative BP score sets.

**Table 4:** The de-entropy values matrix.

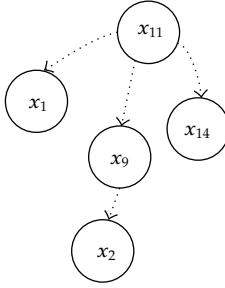
	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	$x_{11}$	$x_{12}$	$x_{13}$	$x_{14}$	$x_{15}$
$x_1$	0.238	0.295	0.232	0.229	0.257	0.176	0.160	0.274	<b>0.379</b>	0.227	0.263	0.282	0.277	0.292	0.194
$x_2$	0.267	0.202	0.176	0.175	0.229	0.187	0.142	0.213	0.283	0.189	0.235	0.251	0.246	0.260	0.170
$x_3$	0.268	0.243	0.147	0.208	0.232	0.156	0.143	0.245	0.286	0.204	0.205	0.252	0.248	0.259	0.172
$x_4$	0.261	0.223	0.207	0.145	0.226	0.141	0.129	0.240	0.279	0.285	0.198	0.217	0.242	0.221	0.202
$x_5$	0.263	0.260	0.206	0.174	0.165	0.154	0.141	0.241	0.281	0.187	0.203	0.249	0.244	0.256	0.168
$x_6$	0.276	0.273	0.183	0.180	0.237	0.128	0.147	0.250	0.293	0.195	0.243	0.259	0.253	0.268	0.176
$x_7$	0.205	0.223	0.140	0.147	0.164	0.131	0.086	0.173	0.205	0.172	0.199	0.181	0.176	0.198	0.175
$x_8$	0.270	0.247	0.210	0.178	0.232	0.157	0.143	0.183	0.286	0.191	0.237	0.253	0.248	0.262	0.172
$x_9$	0.253	0.231	0.167	0.165	0.188	0.147	0.134	0.230	0.205	0.179	0.224	0.238	0.233	0.246	0.161
$x_{10}$	0.160	0.145	0.125	0.156	0.138	0.099	0.091	0.146	0.170	0.108	0.128	0.150	0.148	0.143	0.153
$x_{11}$	0.391	0.390	0.233	0.232	0.265	0.207	0.177	0.299	0.355	0.250	0.228	0.312	0.303	0.381	0.243
$x_{12}$	0.310	0.305	0.239	0.238	0.266	0.212	0.196	0.281	0.329	0.259	0.271	0.228	0.285	0.300	0.233
$x_{13}$	0.275	0.271	0.181	0.180	0.204	0.160	0.146	0.249	0.351	0.194	0.243	0.258	0.189	0.267	0.175
$x_{14}$	0.281	0.337	0.187	0.184	0.242	0.176	0.149	0.253	0.297	0.198	0.247	0.264	0.239	0.210	0.179
$x_{15}$	0.194	0.189	0.150	0.184	0.167	0.131	0.121	0.177	0.206	0.260	0.168	0.212	0.179	0.186	0.117

## 4. Discussion

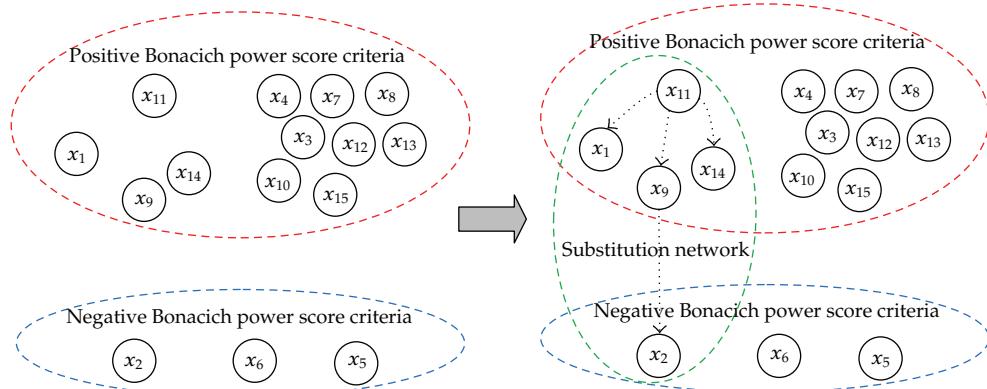
In this section, we will discuss the advantages of the proposed model and its application in the brand equity evaluation.

### 4.1. Hybrid Model for Extracting Key Criteria

In this study, the proposed hybrid model combined both SNA and the MMDE algorithm to extract the key criteria from the existing criteria, especially when researchers were faced with excessive available criteria and with a considerable degree of possible substitution. The main purpose of using SNA to calculate the BP scores was to separate existing criteria into powerful and weaker criteria sets. Because the BP scores imply both centrality and power of the nodes in one's neighborhood, we can exclude the criteria which are not "powerful" or "central." If there are too many criteria without obvious substitution, it is especially useful and reasonable for the analyst to simplify the criteria by groups.



**Figure 5:** Steps to obtain the substitution network.



**Figure 6:** Steps to extract nine criteria from the original fifteen criteria.

The MMDE algorithm was used to set an appropriate threshold value and obtain adequate information to delineate the substitution network for further analysis. The results of the MMDE algorithm can be tracked and evaluated easily because of the obvious causal relationship. It is useful for a researcher if some specific criteria are considered as necessary criteria for evaluation. The researcher can easily find out which criteria should also be included in evaluating if the specific criterion has to be included.

After we obtained the results from the SNA and MMDE algorithms, we can exclude the weaker criteria and replaceable criteria and obtain the criteria which meet the objectives of the proposed hybrid model. In this paper, we demonstrated the effect and feasibility of the hybrid model, the first model to reduce criteria and consider the power, centrality, and substitution of criteria, for extracting key criteria.

#### 4.2. The Application of Proposed Model in Brand Equity Evaluation

Compared to other assessment frameworks, the strength of three dimensions of Interbrand, namely, brand stability, brand trend, and marketing support is relatively weak. This can be explained by the fact that they are easily substituted by other dimensions, which means that they share similar meaning with each other. Take brand stability as an example, aside from the fact that the contents of its measurement criteria are closely related to brand leadership, market, and brand trend of Interbrand, they are related to the dominance of the relevant

market, psychographic and identity status of BBDO, and also overlap with the loyalty driver of HIROSE. Brand trend measures the long-term performance of market share, expected brand performance, the sensitivity of the brand planning, and competitive action. Those four criteria are relative to brand leadership; market and marketing support those dimensions of Interbrand; therefore, both brand trend and marketing support are regarded as comparative weak and suggest to be ruled out of the assessment framework.

In the second analytical phase, we found that brand leadership, dominance of relevance market, and loyalty drivers were the three dimensions that can be substituted by brand status of BBDO. If we look carefully at the breakdown of BBDO in brand status, it contains two indicators of brand strength and brand appeal; five different levels of the dimension were constructed to measure the functional, market, psychographic, identify, and legendary status of the brand. However, brand leadership of Interbrand mainly measures market share, visibility, market position, and competitors outline which are very similar to the market status of BBDO. Then the dimension dominance of relevance market from BBDO mainly investigates the size of the brand revenue relative to the market leader which can also be included in brand strength and brand appeal of the market status of the brand status. Finally, the loyalty driver of HIROSE is also highly related to brand trust and brand loyalty of the identity status of the brand status.

According to data analysis, the original 15 dimensions are extracted into the final 9 dimensions using appropriate research methods to establish an assessment framework which can integrate the complete concept from various models of brand equity. From the analysis above we can conclude that the brand status of BBDO has relatively strong substitution capability since this dimension is more completely compared to other dimensions among different models since it can both cover the brand leadership of Interbrand and dominance of relevance market of BBDO. Therefore, it can be seen that, with the exception of dimensions between the different models, there are possibilities for mutual substitution and that the proposed model can efficiently extract the key criteria required.

## 5. Conclusions

In this paper, we used social network analysis and the maximum mean de-entropy algorithm to extract key criteria from numerous existing criteria. We also applied the proposed model to evaluating brand equity. The effect and feasibility of the hybrid model were demonstrated. However, we were able to have discussions with only a few experts and applied the model only to evaluating brand equity. In a follow-up study, we recommend that a further criteria extraction project(s) be executed using the proposed hybrid model to demonstrate increased feasibility of our model and allow the provision of more reliable conclusions.

## References

- [1] B. S. Blanchard and W. J. Fabrycky, *Systems Engineering and Analysis*, Prentice Hall, New Jersey, NJ, USA, 4th edition, 2005.
- [2] Wikipedia, Social Network Analysis, <http://en.wikipedia.org/>, 2012.
- [3] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*, Cambridge University Press, New York, NY, USA, 1994.
- [4] M. Balconi, S. Breschi, and F. Lissoni, "Networks of inventors and the role of academia: an exploration of Italian patent data," *Research Policy*, vol. 33, no. 1, pp. 127–145, 2004.
- [5] M. Balconi and A. Laboranti, "University-industry interactions in applied research: the case of microelectronics," *Research Policy*, vol. 35, no. 10, pp. 1616–1630, 2006.

- [6] F. Murray, "Innovation as co-evolution of scientific and technological networks: exploring tissue engineering," *Research Policy*, vol. 31, no. 8-9, pp. 1389–1403, 2002.
- [7] C. S. Wagner and L. Leydesdorff, "Network structure, self-organization, and the growth of international collaboration in science," *Research Policy*, vol. 34, no. 10, pp. 1608–1618, 2005.
- [8] L. C. Yin, H. Kretschmer, R. A. Hanneman, and Z. Y. Liu, "Connection and stratification in research collaboration: an analysis of the COLLNET network," *Information Processing and Management*, vol. 42, no. 6, pp. 1599–1613, 2006.
- [9] J. P. Scott, *Social Network Analysis: A Handbook*, Sage Publications, London, UK, 2nd edition, 2000.
- [10] D. Knoke and S. Yang, *Social Network Analysis*, Sage Publications, London, UK, 2nd edition, 2007.
- [11] L. C. Freeman, "Centrality in social networks conceptual clarification," *Social Networks*, vol. 1, no. 1, pp. 215–239, 1978.
- [12] P. Bonacich, "Factoring and weighting approaches to status scores and clique identification," *The Journal of Mathematical Sociology*, vol. 2, no. 1, pp. 113–120, 1972.
- [13] P. Bonacich, "Power and centrality: a family of measures," *American Journal of Sociology*, vol. 92, no. 5, pp. 1170–1182, 1987.
- [14] C. T. Butts and K. M. Carley, "Some simple algorithms for structural comparison," *Computational and Mathematical Organization Theory*, vol. 11, no. 4, pp. 291–305, 2005.
- [15] I. K. Aleksandr, *Mathematical Foundations of Information Theory*, Courier Dover, New York, NY, USA, 1957.
- [16] C. W. Li and G. H. Tzeng, "Identification of a threshold value for the DEMATEL method using the maximum mean de-entropy algorithm to find critical services provided by a semiconductor intellectual property mall," *Expert Systems with Applications*, vol. 36, no. 6, pp. 9891–9898, 2009.
- [17] C. W. Li and G. H. Tzeng, "Identification of interrelationship of key customers' needs based on structural model for services/capabilities provided by a semiconductor-intellectual-property mall," *Applied Mathematics and Computation*, vol. 215, no. 6, pp. 2001–2010, 2009.
- [18] P. Bonacich and T. M. Liggett, "Asymptotics of a matrix valued Markov chain arising in sociology," *Stochastic Processes and their Applications*, vol. 104, no. 1, pp. 155–171, 2003.
- [19] N. Penrose, "Valuation of brand names and trade marks," in *Brand Valuation: Establishing a True and Fair View*, pp. 32–45, 1989.
- [20] R. Zimmermann, U. Klein-Bolting, B. Sander, and T. Murad-Aga, *Brand Equity Review*, BBDO Group, Berlin, Germany, 2001.
- [21] R. Zimmermann, *Brand Equity Excellence*, BBDO Group, Berlin, Germany, 2002.
- [22] Department of Trade and Industry, "Report of brand valuation," in *Proceedings of the Seminar of Corporate Legal System in the Ministry of Economy*, Tokyo, Japan, 2002.
- [23] C. T. Butts, "Tools for social network analysis," <http://cran.r-project.org/>, 2010.
- [24] C. T. Butts, "Software manual for the R sna package," <http://erzuli.ss.uci.edu/R.stuff/>, 2010.

*Research Article*

## **Solving Packing Problems by a Distributed Global Optimization Algorithm**

**Nian-Ze Hu,<sup>1</sup> Han-Lin Li,<sup>2</sup> and Jung-Fa Tsai<sup>3</sup>**

<sup>1</sup> Department of Information Management, National Formosa University, Yunlin 632, Taiwan

<sup>2</sup> Institute of Information Management, National Chiao Tung University, No. 1001, Ta Hsueh Road, Hsinchu 300, Taiwan

<sup>3</sup> Department of Business Management, National Taipei University of Technology, No. 1, Sec. 3, Chung Hsiao E. Road, Taipei 10608, Taiwan

Correspondence should be addressed to Jung-Fa Tsai, jftsai@ntut.edu.tw

Received 23 February 2012; Accepted 9 May 2012

Academic Editor: Yi-Chung Hu

Copyright © 2012 Nian-Ze Hu 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.

Packing optimization problems aim to seek the best way of placing a given set of rectangular boxes within a minimum volume rectangular box. Current packing optimization methods either find it difficult to obtain an optimal solution or require too many extra 0-1 variables in the solution process. This study develops a novel method to convert the nonlinear objective function in a packing program into an increasing function with single variable and two fixed parameters. The original packing program then becomes a linear program promising to obtain a global optimum. Such a linear program is decomposed into several subproblems by specifying various parameter values, which is solvable simultaneously by a distributed computation algorithm. A reference solution obtained by applying a genetic algorithm is used as an upper bound of the optimal solution, used to reduce the entire search region.

### **1. Introduction**

A packing optimization problem is to seek a minimal container, which can hold a given number of smaller rectangular boxes. This problem is also referred to as a container loading problem. Packing cartons into a container is concerning material handling in the manufacturing and distribution industries. For instance, workers in the harbor have to pack more than one type of cartons into a container, and they often deal with this problem by the rule of thumb but a systematic approach. Therefore, the utilization of the container is low, which will cause additional costs.

Similar issues can be found in fields such as knapsack [1, 2], cutting stock [3, 4], assortment problems [5, 6], rectangular packing [7], pallet loading [8, 9], and container loading problems [10, 11]. In addition, researchers have dealt with various related problems. For instance, Dowsland [12] and Egeblad and Pisinger [1], He et al. [13], Wu et al. [14], de Almeida and Figueiredo [15], Miyazawa and Wakabayashi [16], and Crainic et al. [17] proposed different heuristic methods for solving three-dimensional packing problems, Chen et al. [10] formulated a mixed integer program for container loading problems, and Li and Chang [6] developed a method for finding the approximate global optimum of the assortment problem. However, Li and Chang's method [6] requires using numerous 0-1 variables to linearize the polynomial objective function in their model, which would involve heavy computation in solving packing problems. Moreover, Chen et al.'s approach [10] can only find a local optimum of packing problems with nonlinear objective function. Recently, many global optimization methods have been developed, where Floudas' method [18] is one of most promising methods for solving general optimization problems. Although Floudas' method [18] can be applied to solve a packing problem to reach finite  $\varepsilon$ -convergence to the global minimum, it requires successively decomposing the concave part of the original problem into linear functions. Besides, it adopts lower boundary techniques, which is time consuming. Li et al. [19] designed a distributed optimization method to improve the computational efficiency for solving packing problems. Tsai and Li [20] also presented an enhanced model with fewer binary variables and used piecewise linearization techniques to transform the nonconvex packing problem into a mixed-integer linear problem, which is solvable to find a global solution.

Due to the complexity and hardness of three-dimensional packing problems, most results on this topic are based on heuristics [21]. Furthermore, parallel computing [22] was adopted to improve the efficiency of combinatorial computation. Parallel Genetic Algorithm (GA) [23], parallel with heuristic [24], and parallel Tabu Search Algorithm (TSA) [25] were proposed to solve container-packing problems under some conditions. These methods are capable of obtaining solutions with good performance relative to test examples in the literature. However, the algorithms cannot guarantee to get a global optimum.

This paper proposes another method for finding the optimum of the packing problem. The major advantage of this method is that it can reformulate the nonlinear objective function of original packing problem as an increasing function with single variable and two given parameters. In addition, distributed computation and genetic algorithm are adopted to improve the efficiency and ensure the optimality. The proposed method then solves the reformulated programs by specifying the parameters sequentially to reach the globally optimal solution on a group of network-connected computers.

## 2. Problem Formulation

Given  $n$  rectangular boxes with fixed lengths, widths, and heights, a packing optimization problem is to allocate these  $n$  boxes within a rectangular container having minimal volume. Denote  $x$ ,  $y$ , and  $z$  as the width, length, and height of the container; the packing optimization problem discussed here is stated as follows:

$$\begin{aligned} & \text{minimize } xyz \\ & \text{subject to (1) all of } n \text{ boxes are nonoverlapping.} \end{aligned}$$

- (2) all of  $n$  boxes are within the range of  $x, y$ , and  $z$ .  
(3)  $\underline{x} \leq x \leq \bar{x}, \underline{y} \leq y \leq \bar{y}, \underline{z} \leq z \leq \bar{z}$   
 $(\underline{x}, \underline{y}, \underline{z}, \bar{x}, \bar{y}, \bar{z} \text{ are constants}).$

(2.1)

According to Chen et al. [10], the current packing model adopts the terminologies as follows.

$(p_i, q_i, r_i)$ : Dimension of box  $i$ ,  $p_i$  is the length,  $q_i$  is the width, and  $r_i$  is the height, and  $p_i, q_i$ , and  $r_i$  are integral constants.  $i \in J, J = \{1, 2, 3, \dots, n\}$  is the set of the given boxes.

$(x, y, z)$ : Variables indicating the length, width, and height of the container.

$(x_i, y_i, z_i)$ : Variables indicating the coordinates of the front-left-bottom corner of box  $i$ .

$(l_{xi}, l_{yi}, l_{zi})$ : Binary variables indicating whether the length of box  $i$  is parallel to the X-axis, Y-axis, or Z-axis. The value of  $l_{xi}$  is equal to 1 if the length of box  $i$  is parallel to the X-axis; otherwise, it is equal to 0. It is clear that  $l_{xi} + l_{yi} + l_{zi} = 1$ .

$(w_{xi}, w_{yi}, w_{zi})$ : Binary variables indicating whether the width of box  $i$  is parallel to the X-axis, Y-axis, or Z-axis. The value of  $w_{xi}$  is equal to 1 if the width of box  $i$  is parallel to the X-axis; otherwise, it is equal to 0. It is clear that  $w_{xi} + w_{yi} + w_{zi} = 1$ .

$(h_{xi}, h_{yi}, h_{zi})$ : Binary variables indicating whether the height of box  $i$  is parallel to the X-, Y-, or Z-axis. The value of  $h_{xi}$  is equal to 1 if the height of box  $i$  is parallel to the X-axis; otherwise, it is equal to 0. It is clear that  $h_{xi} + h_{yi} + h_{zi} = 1$ .

For a pair of boxes  $(i, k)$ , where  $i < k$ , there is a set of 0-1 vector  $(a_{ik}, b_{ik}, c_{ik}, d_{ik}, e_{ik}, f_{ik})$  defined as

$$\begin{aligned} a_{ik} &= 1 \text{ if box } i \text{ is on the left of box } k, \text{ otherwise } a_{ik} = 0, \\ b_{ik} &= 1 \text{ if box } i \text{ is on the right of box } k, \text{ otherwise } b_{ik} = 0, \\ c_{ik} &= 1 \text{ if box } i \text{ is behind box } k, \text{ otherwise } c_{ik} = 0, \\ d_{ik} &= 1 \text{ if box } i \text{ is in front of box } k, \text{ otherwise } d_{ik} = 0, \\ e_{ik} &= 1 \text{ if box } i \text{ is below box } k, \text{ otherwise } e_{ik} = 0, \\ f_{ik} &= 1 \text{ if box } i \text{ is above box } k, \text{ otherwise } f_{ik} = 0. \end{aligned}$$

The front-left-bottom corner of the container is fixed at the origin. The interpretation of these variables is illustrated in Figure 1. Figure 1 contains two boxes  $i$  and  $k$ , where box  $i$  is located with its length along the X-axis and the width parallel to the Z-axis, and box  $k$  is located with its length along the Z-axis and the width parallel to the X-axis. We then have  $l_{xi}, w_{zi}, h_{yi}, l_{zk}, w_{xk}$ , and  $h_{yk}$  equal to 1. In addition, since box  $i$  is located on the left-hand side of and in front of box  $k$ , it is clear that  $a_{ik} = d_{ik} = 1$  and  $b_{ik} = c_{ik} = e_{ik} = f_{ik} = 0$ .

According to Chen et al. [10] and Tsai and Li [20], the packing problem can be formulated as follows.

*Problem 1.*

$$\text{Minimize } \text{Obj} = xyz \quad (2.2)$$

$$\text{subject to } x_i + p_i l_{xi} + q_i w_{xi} + r_i h_{xi} \leq x_k + (1 - a_{ik})M \quad \forall i, k \in J, i < k, \quad (2.3)$$

$$x_k + p_k l_{xk} + q_k w_{xk} + r_k h_{xk} \leq x_i + (1 - b_{ik})M \quad \forall i, k \in J, i < k, \quad (2.4)$$

$$y_i + p_i l_{yi} + q_i w_{yi} + r_i h_{yi} \leq y_k + (1 - c_{ik})M, \quad \forall i, k \in J, i < k, \quad (2.5)$$

$$y_k + p_k l_{yk} + q_k w_{yk} + r_k h_{yk} \leq y_i + (1 - d_{ik})M, \quad \forall i, k \in J, i < k, \quad (2.6)$$

$$z_i + p_i l_{zi} + q_i w_{zi} + r_i h_{zi} \leq z_k + (1 - e_{ik})M, \quad \forall i, k \in J, i < k, \quad (2.7)$$

$$z_k + p_k l_{zk} + q_k w_{zk} + r_k h_{zk} \leq z_i + (1 - f_{ik})M, \quad \forall i, k \in J, i < k, \quad (2.8)$$

$$a_{ik} + b_{ik} + c_{ik} + d_{ik} + e_{ik} + f_{ik} \geq 1, \quad \forall i, k \in J, i < k, \quad (2.9)$$

$$x_i + p_i l_{xi} + q_i w_{xi} + r_i h_{xi} \leq x, \quad \forall i \in J, \quad (2.10)$$

$$y_i + p_i l_{yi} + q_i w_{yi} + r_i h_{yi} \leq y, \quad \forall i \in J, \quad (2.11)$$

$$z_i + p_i l_{zi} + q_i w_{zi} + r_i h_{zi} \leq z, \quad \forall i \in J, \quad (2.12)$$

$$l_{xi} + l_{yi} + l_{zi} = 1, \quad \forall i \in J, \quad (2.13)$$

$$w_{xi} + w_{yi} + w_{zi} = 1, \quad \forall i \in J, \quad (2.14)$$

$$h_{xi} + h_{yi} + h_{zi} = 1, \quad \forall i \in J, \quad (2.15)$$

$$l_{xi} + w_{xi} + h_{xi} = 1, \quad \forall i \in J, \quad (2.16)$$

$$l_{yi} + w_{yi} + h_{yi} = 1, \quad \forall i \in J, \quad (2.17)$$

$$l_{zi} + w_{zi} + h_{zi} = 1, \quad \forall i \in J, \quad (2.18)$$

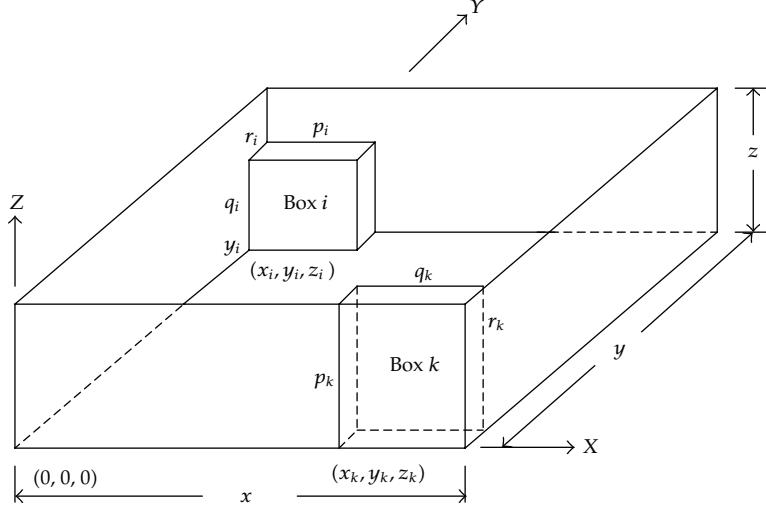
where

$$l_{xi}, l_{yi}, l_{zi}, w_{xi}, w_{yi}, w_{zi}, h_{xi}, h_{yi}, h_{zi}, a_{ik}, b_{ik}, c_{ik}, d_{ik}, e_{ik}, f_{ik} \text{ are 0-1 variables,} \quad (2.19)$$

$$M = \max\{\bar{x}, \bar{y}, \bar{z}\}, x_i, y_i, z_i \geq 1, 1 \leq \underline{x} \leq x \leq \bar{x}, 1 \leq \underline{y} \leq y \leq \bar{y}, 1 \leq \underline{z} \leq z \leq \bar{z}, \\ \underline{x}, \underline{y}, \underline{z}, \bar{x}, \bar{y}, \bar{z} \text{ are constants,} \quad (2.20)$$

$$x, y, z \text{ are positive variables.} \quad (2.21)$$

The objective of this model is to minimize the volume of the container. The constraints (2.3)–(2.9) are nonoverlapping conditions used to ensure that none of these  $n$  boxes overlaps with each other. Constraints (2.10)–(2.12) ensure that all boxes are within the enveloping container. Constraints (2.13)–(2.18) describe the allocation restrictions among logic variables. For instance, (2.13) implies that the length of box  $i$  is parallel to one of the axes. (2.16) implies that only one of the length, the width, and the height of box  $i$  is parallel to X-axis.



**Figure 1:** Graphical illustration.

Since the objective function of Problem 1 is a product term, Problem 1 is a nonlinear mixed 0-1 program, which is difficult to be solved by current optimization methods. Chen et al. [10] can only solve linear objective function. Tsai and Li's method [20] can solve Problem 1 at the price of adding many extra 0-1 variables.

### 3. Proposed Method

Consider the objective function  $\text{Obj} = xyz$  in (2.2), where  $x \geq y \geq z$ ,  $1 \leq \underline{x} \leq x \leq \bar{x}$ ,  $1 \leq \underline{y} \leq \bar{y}$ ,  $1 \leq \underline{z} \leq z \leq \bar{z}$  and  $x, y, z$  are positive variables. Denote  $r$  and  $s$  as two variables defined as  $r = x - y + z$ ,  $s = x - z$ . Replace  $y$  by  $2x - s - r$  and replace  $z$  by  $x - s$ .  $xyz$  then becomes  $\text{Obj}'$  as follows.

$$\text{Obj}' = x[2x^2 - (r + 3s)x + rs + s^2]. \quad (3.1)$$

We then have the following propositions.

**Proposition 3.1.** Suppose  $r$  and  $s$  in (3.1) are fixed values, then  $\text{Obj}'$  is an increasing function.

*Proof.* Since  $\partial\text{Obj}'/\partial x = 6x^2 - 2x(r + 3s) + rs + s^2 = xy + yz + 2xz > 0$ , it is clear that  $\text{Obj}'$  is an increasing function.  $\square$

**Proposition 3.2.** The optimal solution of Problem 1 is integral.

*Proof.* Since dimensions of box  $i$ ,  $p_i$ ,  $q_i$ ,  $r_i$ , are integral constants for  $i = 1, 2, \dots, n$  and all of  $n$  boxes are nonoverlapping, therefore,  $x^*$ ,  $y^*$ , and  $z^*$  that indicate the optimal solution of the container must be integral.  $\square$

**Proposition 3.3.**  $\sqrt[3]{\sum_{i=1}^n p_i q_i r_i} \leq (x + y + z)/3$ , where  $p_i$  is the length,  $q_i$  is the width, and  $r_i$  is the height of the given box  $i$ . [19, 20]

*Proof.* Since  $\sum_{i=1}^n p_i q_i r_i \leq xyz$  and  $\sqrt[3]{xyz} \leq (x + y + z)/3$ , then we can have  $\sqrt[3]{\sum_{i=1}^n p_i q_i r_i} \leq (x + y + z)/3$ .  $\square$

According to the above propositions and given values of  $r$  and  $s$  (denoted as  $\hat{r}$  and  $\hat{s}$ ), consider the following program.

*Problem 2.*

$$\begin{aligned} \text{Minimize } & \text{Obj}'(x) = 2x^3 - (\hat{r} + 3\hat{s})x^2 + (\hat{r}\hat{s} + \hat{s}^2)x \\ \text{subject to } & \sqrt[3]{\sum_{i=1}^n p_i q_i r_i} \leq \frac{x + y + z}{3}, \\ & \hat{r} = x - y + z, \quad \hat{s} = x - z, (2.3) \sim (2.21), \\ & \hat{r} \text{ and } \hat{s} \text{ are fixed values.} \end{aligned} \tag{3.2}$$

**Proposition 3.4.** If  $(x^\Delta, y^\Delta, z^\Delta)$  is the solution of Problem 2 found by a genetic algorithm and  $(x^*, y^*, z^*)$  is the globally optimal solution of Problem 2, then  $x^* \leq x^\Delta$ .

*Proof.* Since  $\text{Obj}'$  is an increasing function with single variable  $x$  (following Proposition 3.1) and  $2(x^*)^3 - (\hat{r} + 3\hat{s})(x^*)^2 + (\hat{r}\hat{s} + \hat{s}^2)x^* \leq 2(x^\Delta)^3 - (\hat{r} + 3\hat{s})(x^\Delta)^2 + (\hat{r}\hat{s} + \hat{s}^2)x^\Delta$ . Hence,  $x^* \leq x^\Delta$ .  $\square$

Adding the constraint  $x \leq x^\Delta$  to Problem 2 for reducing the search region of the optimal solution, we can have the following two programs.

*Problem 3.*

$$\begin{aligned} \text{Minimize } & \text{Obj}'(x) = 2x^3 - (\hat{r} + 3\hat{s})x^2 + (\hat{r}\hat{s} + \hat{s}^2)x \\ \text{subject to } & x \leq x^\Delta, \\ & \hat{r} = x - y + z, \quad \hat{s} = x - z, (2.3) \sim (2.21), (3.2), \\ & \hat{r} \text{ and } \hat{s} \text{ are fixed values.} \end{aligned} \tag{3.3}$$

*Problem 4.*

$$\begin{aligned} \text{Minimize } & x \\ \text{subject to } & \text{all the constraints in Problem 3.} \end{aligned} \tag{3.4}$$

**Proposition 3.5.** Let  $(x^*, y^*, z^*)$  be the global optimum of Problem 3, then  $(x^*, y^*, z^*)$  is also the global optimum of Problem 4.

*Proof.* Since  $\text{Obj}'(x)$  is an increasing function with single variable  $x$  (following Proposition 3.1), Problems 2 and 3 have the same global optimum  $(x^*, y^*, z^*)$ .  $\square$

According to the above propositions, a packing optimization problem, which is a nonlinear 0-1 programming problem, can be transformed into a linear 0-1 program by introducing two parameters  $r$  and  $s$ . Then we can guarantee to obtain the global optimum of a packing problem by solving the transformed linear 0-1 programs. The distributed computation scheme is also proposed to enhance the computational efficiency.

#### 4. Distributed Algorithm

The solution procedure for solving Problem 1 to obtain a global optimum is presented in the following with a flow chart shown in Figure 2.

*Step 1.* (Find an initial solution by GA). From Proposition 3.4, the obtained solution is  $(x^\Delta, y^\Delta, z^\Delta)$  and the constraint  $x \leq x^\Delta$  can be utilized to reduce the searching space of the global solution.

*Step 2.* Denote  $s(m)$  and  $s(1)$  as the upper and lower bounds of  $s$ , respectively. Find the bounds of  $s = x - z$  by solving the following linear programs:

$$\begin{aligned} s(m) &= \text{Max}\{x - z \mid \text{subject to (2.3) } \sim \text{(2.21), (3.2)}\}, \\ s(1) &= \text{Min}\{x - z \mid \text{subject to (2.3) } \sim \text{(2.21), (3.2)}\}. \end{aligned} \quad (4.1)$$

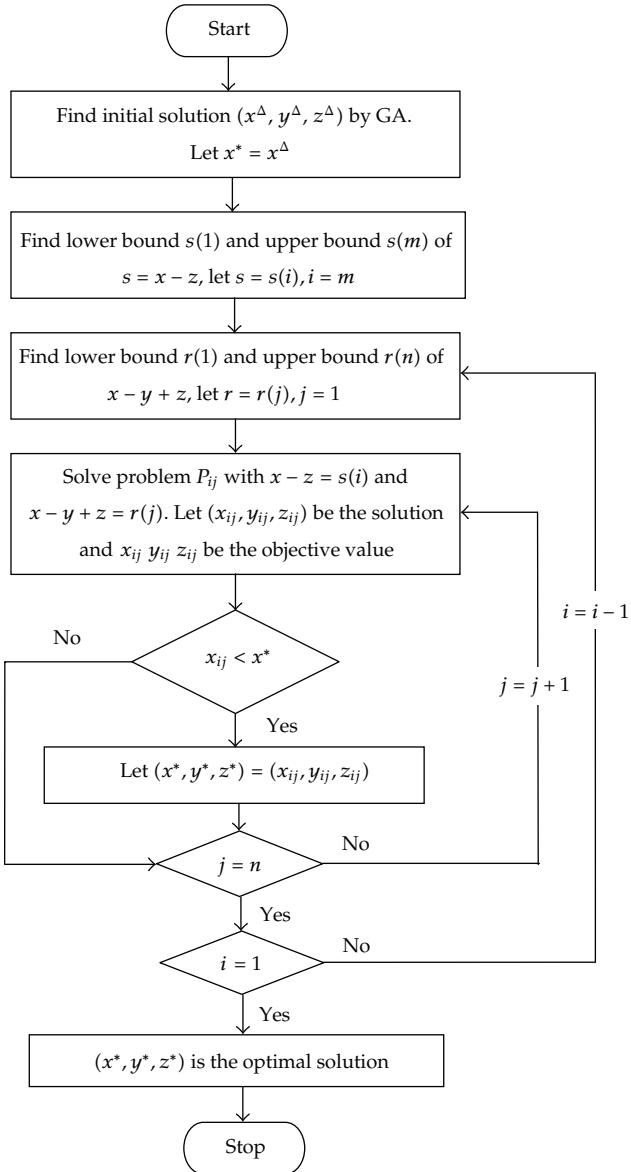
Let  $x - z = s(i)$ ,  $i = m$ , and go to Step 3.

*Step 3.* Denote  $r(n)$  and  $r(1)$  as the upper and lower bounds of  $r$ , respectively. Find the bounds of  $r = x - y + z$  by solving the following linear programs:

$$\begin{aligned} r(n) &= \text{Max}\{x - y + z \mid \text{subject to (2.3) } \sim \text{(2.21), (3.2), } x - z = s(i)\}, \\ r(1) &= \text{Min}\{x - y + z \mid \text{subject to (2.3) } \sim \text{(2.21), (3.2), } x - z = s(i)\}, \end{aligned} \quad (4.2)$$

Let  $r = r(j)$ ,  $j = 1$ , and go to Step 4.

*Step 4.* Decompose main problem and perform distributed packing algorithm. According to verity of  $r$  and  $s$ , the main problem can be decomposed into several subproblems. The transformed subproblem of each iterative process is listed as follows.



**Figure 2:** Solution algorithm.

### Problem $P_{ij}$ .

Minimize  $\text{Obj}_{ij} = x$

subject to (2.3) – (2.21), (3.2), (3.3),  $x - z = s(i)$ ,  $x - y + z = r(j)$ ,

$$\text{Obj}' = x \left[ 2x^2 - (r(j) + 3s(i))x + r(j)s(i) + s(i)^2 \right],$$

$$1 \leq i \leq m, 1 \leq j \leq n. \quad (4.3)$$

Every sub-problem can be submitted to client computer and solved independently. Server computer controls the whole process and compares the solutions  $(x_{ij}, y_{ij}, z_{ij})$  of the problem  $P_{ij}$  with the initial solution  $(x^\Delta, y^\Delta, z^\Delta)$ . If  $x_{ij}$  is smaller than  $x^\Delta$ , then  $x^\Delta$  is replaced by  $x_{ij}$ .

The structure of distributed packing algorithm is developed based on star schema. Owing to reduce the network loading and improve the computational performance of each client computer, all results found on all clients are directly sent to host computer.

*Step 5.* Let  $j = j + 1$ . If  $j > n$ , then go to Step 6. Otherwise, go to Step 4.

*Step 6.* Let  $i = i - 1$ . If  $i < 1$ , then go to Step 3. Otherwise, go to Step 7.

*Step 7.* The whole process is finished and the host computer obtains the optimal solution  $(x^*, y^*, z^*)$  with the objective value  $x^*y^*z^*$ .

## 5. Numerical Examples

To validate the proposed method, several examples with different number of boxes are solved by LINGO 11.0 [26] with the distributed algorithm. Two of the test problems denoted as Problems 1 and 2 are taken from Chen et al. [10]. The other examples are arbitrarily generated. Solving these problems by the proposed method, the obtained globally optimal solutions are listed in Tables 1 and 3. Comparison results between GA and the proposed method are shown in Table 2, and the associated graphs are presented in Figures 3 and 4.

Packing problems often arise in logistic application. The following example (Problem 5) demonstrates how to apply the proposed algorithm in transportation problem and compare the result with traditional genetic algorithm.

*Problem 5.* Several kinds of goods are packed into a container so as to deliver to 6 different stores on a trip. The dimensions of width and height of the container are 5 and 4. All goods are packed in cubic boxes, which have three different sizes. In order to take less time during unloading, boxes sent to the same store must be packed together. Different groups of boxes cannot overlap each other. Moreover, the packing order to each group must be ordered of the arriving time to each store. The boxes required to be sent to each store are listed in Table 4. The objective is to determine the smallest dimensions of the container.

*Solution 1.* The arrangement of boxes can be treated as level assortment. The boxes packed in the same level will be delivered to the same store. After performing the proposed method, list of the optimal solutions are shown in Table 5, and illustrated graph is presented in Figure 5.

**Table 1:** Computational results.

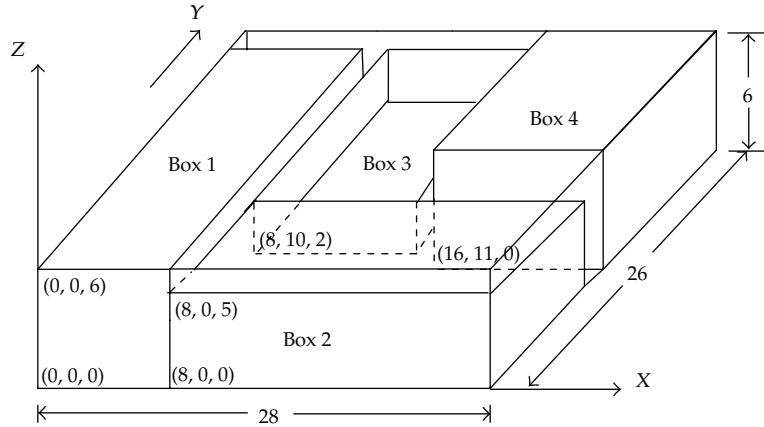
Problem number	Box number	$p_i$	$q_i$	$r_i$	$x_i$	$y_i$	$z_i$	$(x, y, z)$	Objective value
1	1	25	8	6	0	0	0	(28, 26, 6)	4368
	2	20	10	5	8	0	0		
	3	16	7	3	8	10	2		
	4	15	12	6	16	11	0		
2	1	25	8	6	10	20	0	(35, 28, 6)	5880
	2	20	10	5	25	0	0		
	3	16	7	3	0	0	3		
	4	15	12	6	10	8	0		
	5	22	8	3	5	0	0		
	6	20	10	4	0	8	0		
3	1	25	8	6	0	10	0	(31, 16, 12)	5952
	2	20	10	5	0	0	0		
	3	16	7	3	9	0	9		
	4	15	12	6	25	0	0		
	5	22	8	3	3	7	9		
	6	20	10	4	0	0	5		
	7	10	8	4	31	0	0		

**Table 2:** Solution comparison of the proposed algorithm and genetic algorithm (GA).

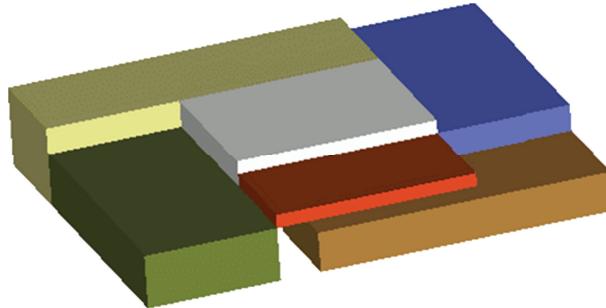
Problem number	GA		Proposed method	
	$(x, y, z)$	Objective value	$(x, y, z)$	Objective value
1 (4 boxes)	(30, 30, 6)	5400	(28, 26, 6)	4368
2 (6 boxes)	(33, 26, 7)	6006	(35, 28, 6)	5880
3 (7 boxes)	(25, 25, 10)	6250	(31, 16, 12)	5952

**Table 3:** Computational results for all boxes are cubic.

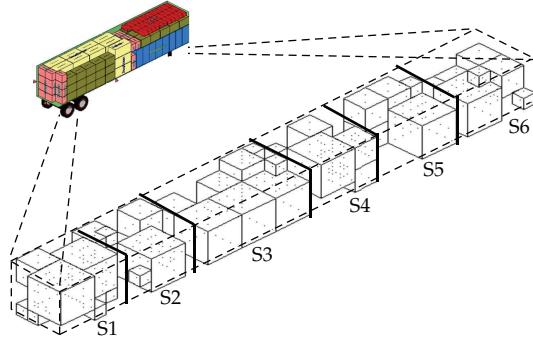
Problem number	Number of cubes	Side $p_i \times q_i \times r_i$	$(x, y, z)$	CPU time (h:min:s)	Objective value
4	3	$1 \times 1 \times 1$	(8, 5, 5)	00:00:08	200 Global optimum
	3	$2 \times 2 \times 2$			
	1	$3 \times 3 \times 3$			
	1	$5 \times 5 \times 5$			
5	4	$1 \times 1 \times 1$	(8, 6, 5)	00:01:26	240 Global optimum
	2	$2 \times 2 \times 2$			
	3	$3 \times 3 \times 3$			
	1	$5 \times 5 \times 5$			
6	4	$1 \times 1 \times 1$	(8, 8, 5)	00:10:12	320 Global optimum
	3	$2 \times 2 \times 2$			
	4	$3 \times 3 \times 3$			
	4	$1 \times 1 \times 1$			
7	3	$2 \times 2 \times 2$	(12, 7, 5)	01:37:44	420 Global optimum
	3	$3 \times 3 \times 3$			
	1	$4 \times 4 \times 4$			
	1	$5 \times 5 \times 5$			



**Figure 3:** The graphical representation of 4 boxes.



**Figure 4:** The solid graphical representation of 6 boxes.



**Figure 5:** The graphical presentation of the 48 boxes for the 6 stores.

## 6. Conclusions

This paper proposes a new method to solve a packing optimization problem. The proposed method reformulates the nonlinear objective function of the original packing problem into a linear function with two given parameters. The proposed method then solves the reformulated linear 0-1 programs by specifying the parameters sequentially to reach the globally optimal solution. Furthermore, this study adopts a distributed genetic algorithm and distributed

**Table 4:** List of stores and boxes (48 boxes).

Store	Goods
S1	A, A, A, B, B, B, C
S2	A, A, B, B, B, C, C
S3	A, A, A, B, B, B, C
S4	A, A, B, B, C, C, C
S5	A, A, A, A, B, B, C
S6	A, A, A, A, A, B, C, C

A: 1-inch cubic box; B: 2-inch cubic box; C: 3-inch cubic box.

**Table 5:** List of optimal arrangement of the boxes.

Store	S1	S2	S3	S4	S5	S6
$(x_1, y_1, z_1)$	A (1, 2, 0)	A (0, 1, 3)	A (4, 0, 2)	A (0, 0, 3)	A (4, 0, 0)	A (0, 0, 0)
$(x_2, y_2, z_2)$	A (3, 3, 3)	A (5, 0, 0)	A (0, 1, 0)	A (0, 2, 3)	A (4, 1, 3)	A (5, 2, 0)
$(x_3, y_3, z_3)$	A (0, 4, 0)	B (3, 0, 0)	A (1, 2, 0)	B (3, 0, 1)	A (0, 2, 0)	A (0, 3, 0)
$(x_4, y_4, z_4)$	B (4, 0, 0)	B (1, 0, 2)	B (3, 0, 0)	B (7, 0, 1)	A (4, 4, 1)	A (2, 4, 0)
$(x_5, y_5, z_5)$	B (0, 0, 1)	B (3, 0, 2)	B (0, 3, 0)	B (1, 0, 2)	B (2, 0, 0)	A (5, 1, 0)
$(x_6, y_6, z_6)$	B (2, 0, 0)	B (1, 0, 0)	B (1, 0, 2)	C (0, 2, 0)	B (2, 0, 2)	B (3, 0, 2)
$(x_7, y_7, z_7)$	C (0, 2, 1)	C (0, 2, 0)	B (0, 3, 2)	C (6, 2, 0)	B (0, 0, 2)	C (0, 1, 1)
$(x_8, y_8, z_8)$	C (3, 2, 0)	C (3, 2, 0)	C (2, 2, 1)	C (3, 2, 0)	C (1, 2, 0)	C (3, 2, 1)
Dimension of $s_i$	$6 \times 5 \times 4$	$6 \times 5 \times 4$	$5 \times 5 \times 4$	$9 \times 5 \times 4$	$5 \times 5 \times 4$	$6 \times 5 \times 4$
Volume of $s_i$	120	120	100	180	100	120

The global solution is (37, 5, 4), and the minimal volume of the container is 740.

packing algorithm to enhance the computational efficiency. Numerical examples demonstrate that the proposed method can be applied to practical problems and solve the problems to obtain the global optimum.

## Acknowledgments

The authors would like to thank the anonymous referees for contributing their valuable comments regarding this paper and thus significantly improving its quality. The paper is partly supported by Taiwan NSC Grant NSC 99-2410-H-027-008-MY3.

## References

- [1] J. Egeblad and D. Pisinger, "Heuristic approaches for the two- and three-dimensional knapsack packing problem," *Computers and Operations Research*, vol. 36, no. 4, pp. 1026–1049, 2009.
- [2] D. Fayard and V. Zissimopoulos, "An approximation algorithm for solving unconstrained two-dimensional knapsack problems," *European Journal of Operational Research*, vol. 84, no. 3, pp. 618–632, 1995.
- [3] M. Hifi and R. Ouafi, "Best-first search and dynamic programming methods for cutting problems: the cases of one or more stock plates," *Computers and Industrial Engineering*, vol. 32, no. 1, pp. 187–205, 1997.
- [4] J. F. Tsai, P. L. Hsieh, and Y. H. Huang, "An optimization algorithm for cutting stock problems in the TFT-LCD industry," *Computers and Industrial Engineering*, vol. 57, no. 3, pp. 913–919, 2009.
- [5] J. E. Beasley, "An algorithm for the two-dimensional assortment problem," *European Journal of Operational Research*, vol. 19, no. 2, pp. 253–261, 1985.

- [6] H. L. Li and C. T. Chang, "An approximately global optimization method for assortment problems," *European Journal of Operational Research*, vol. 105, no. 3, pp. 604–612, 1998.
- [7] J. F. Tsai, P. C. Wang, and M. H. Lin, "An efficient deterministic optimization approach for rectangular packing problems," *Optimization*. In press.
- [8] F. H. F. Liu and C. J. Hsiao, "A three-dimensional pallet loading method for single-size boxes," *Journal of the Operational Research Society*, vol. 48, no. 7, pp. 726–735, 1997.
- [9] J. Terno, G. Scheithauer, U. Sommerweiß, and J. Riehme, "An efficient approach for the multi-pallet loading problem," *European Journal of Operational Research*, vol. 123, no. 2, pp. 372–381, 2000.
- [10] C. S. Chen, S. M. Lee, and Q. S. Shen, "An analytical model for the container loading problem," *European Journal of Operational Research*, vol. 80, no. 1, pp. 68–76, 1995.
- [11] G. Scheithauer, "LP-based bounds for the container and multi-container loading problem," *International Transactions in Operational Research*, vol. 6, pp. 199–213, 1999.
- [12] W. B. Dowsland, "Three-dimensional packing-solution approaches and heuristic development," *International Journal of Production Research*, vol. 29, no. 8, pp. 1673–1685, 1991.
- [13] Y. He, Y. Wu, and R. de Souza, "A global search framework for practical three-dimensional packing with variable carton orientations," *Computers and Operations Research*, vol. 39, no. 10, pp. 2395–2414, 2012.
- [14] Y. Wu, W. Li, M. Goh, and R. de Souza, "Three-dimensional bin packing problem with variable bin height," *European Journal of Operational Research*, vol. 202, no. 2, pp. 347–355, 2010.
- [15] A. de Almeida and M. B. Figueiredo, "A particular approach for the Three-dimensional Packing Problem with additional constraints," *Computers and Operations Research*, vol. 37, no. 11, pp. 1968–1976, 2010.
- [16] F. K. Miyazawa and Y. Wakabayashi, "Three-dimensional packings with rotations," *Computers and Operations Research*, vol. 36, no. 10, pp. 2801–2815, 2009.
- [17] T. G. Crainic, G. Perboli, and R. Tadei, "TS<sup>2</sup>PACK: a two-level tabu search for the three-dimensional bin packing problem," *European Journal of Operational Research*, vol. 195, no. 3, pp. 744–760, 2009.
- [18] C. A. Floudas, *Deterministic Global Optimization: Theory, Computational Methods and Applications*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000.
- [19] H. L. Li, J. F. Tsai, and N. Z. Hu, "A distributed global optimization method for packing problems," *Journal of the Operational Research Society*, vol. 54, no. 4, pp. 419–425, 2003.
- [20] J. F. Tsai and H. L. Li, "A global optimization method for packing problems," *Engineering Optimization*, vol. 38, no. 6, pp. 687–700, 2006.
- [21] G. Wäscher, H. Hauffner, and H. Schumann, "An improved typology of cutting and packing problems," *European Journal of Operational Research*, vol. 183, no. 3, pp. 1109–1130, 2007.
- [22] B. Bourreau, T. Gabriel Crainic, and B. Gendron, "Branch-and-bound parallelization strategies applied to a depot location and container fleet management problem," *Parallel Computing*, vol. 26, no. 1, pp. 27–46, 2000.
- [23] H. Gehring and A. Bortfeldt, "A parallel genetic algorithm for solving the container loading problem," *International Transactions in Operational Research*, vol. 9, pp. 497–511, 2002.
- [24] J. Błazewicz and R. Walkowiak, "A new parallel approach for multi-dimensional packing problems," *Lecture Notes in Computer Science*, vol. 2328, pp. 585–591, 2006.
- [25] A. Bortfeldt, H. Gehring, and D. Mack, "A parallel tabu search algorithm for solving the container loading problem," *Parallel Computing*, vol. 29, no. 5, pp. 641–662, 2003.
- [26] LINDO System, *LINGO Release 11.0*, LINDO System, Chicago, Ill, USA, 2008.

*Research Article*

## **Hybrid Optimization Approach for the Design of Mechanisms Using a New Error Estimator**

**A. Sedano,<sup>1</sup> R. Sancibrian,<sup>2</sup> A. de Juan,<sup>2</sup> F. Viadero,<sup>2</sup> and F. Egaña<sup>3</sup>**

<sup>1</sup> Calculation Department, MTOI, C/Maria Viscarret 1, Ártica (Berrioplano), 31013 Navarra, Spain

<sup>2</sup> Department of Structural and Mechanical Engineering, University of Cantabria, Avenida de los Castros s/n, 39005 Santander, Spain

<sup>3</sup> Mechatronics Department, Tekniker, Avenida Otaola 20, 20600 Eibar, Spain

Correspondence should be addressed to R. Sancibrian, sancibrr@unican.es

Received 24 February 2012; Revised 24 April 2012; Accepted 14 May 2012

Academic Editor: Yi-Chung Hu

Copyright © 2012 A. Sedano 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 hybrid optimization approach for the design of linkages is presented. The method is applied to the dimensional synthesis of mechanism and combines the merits of both stochastic and deterministic optimization. The stochastic optimization approach is based on a real-valued evolutionary algorithm (EA) and is used for extensive exploration of the design variable space when searching for the best linkage. The deterministic approach uses a local optimization technique to improve the efficiency by reducing the high CPU time that EA techniques require in this kind of applications. To that end, the deterministic approach is implemented in the evolutionary algorithm in two stages. The first stage is the fitness evaluation where the deterministic approach is used to obtain an effective new error estimator. In the second stage the deterministic approach refines the solution provided by the evolutionary part of the algorithm. The new error estimator enables the evaluation of the different individuals in each generation, avoiding the removal of well-adapted linkages that other methods would not detect. The efficiency, robustness, and accuracy of the proposed method are tested for the design of a mechanism in two examples.

### **1. Introduction**

The mechanical design of modern machines is often very complex and needs very sophisticated tools to meet technological requirements. The design of linkages is no exception, and modern applications in this field have increasingly demanding requirements. The design of linkages consists in obtaining the best mechanism to fulfil a specific motion characteristic demanded by a specific task. In many engineering design fields there are three common requirements known as function generation, path generation, or rigid-body guidance [1, 2]. Dimensional synthesis deals with the determination of the kinematic parameters of the mechanism necessary to satisfy the required motion characteristics.

Different techniques have been used for the synthesis of mechanisms including graphical and analytical techniques [1–3]. Graphical and analytical methods developed in the literature are relatively restricted because they find the exact solution for a reduced number of prescribed poses and variables. However, during the last decades numerical methods have enabled an increase in the complexity of the problems by using numerical optimization techniques [4–8].

Despite the work done in dimensional synthesis over recent decades, the design of mechanisms is still a task where the intuition and experience of the engineers play an important role. One of the main reasons for this is the large number of variables involved in a strongly nonlinear problem. Under these circumstances the design variable space contains too many local minima and only some of them can be identified as local solutions. These local solutions provide an error below a limit established by the designer and can be considered acceptable. However, only the global minimum leads to the solution that provides the greatest accuracy and this should be the main objective in the design of mechanisms.

The application of local optimization techniques to the synthesis of mechanisms took place mainly during the 80s and 90s. Although other techniques have become more important in recent years, they remain important so far. Some local search methods have been described in references [4–6]. The main disadvantage of these methods is their dependence on the initial point, or initial guess, although they also require a differentiable objective function. Several research works have been done to achieve exact differentiation, which improve the accuracy and efficiency of these methods. For example, in [5] exact gradient is determined to optimize a six-bar and eight-bar mechanism. In [6] a general synthesis procedure is obtained by using exact differentiation and it is applied to different kinds of problems. However, the dependence on the initial point cannot be avoided and therein lies the weakest point of local search methods.

Global search methods avoid the dependence on the initial point, but there is a sharp increase in the computational time necessary to achieve convergence. Genetic algorithms (GAs) [7, 8], evolutionary algorithms (EAs) [9], and Particle Swarm (PS) are some of the most frequently used optimization techniques in the literature. All these techniques mimic the behaviour of processes found in nature and are based on biological processes.

Genetic and evolutionary algorithms apply the principles of evolution found in nature to the problem of finding an optimal solution. Holland [10] was the first to introduce the GA and DeJong [11] verified the usage. In GA the genes are usually encoded using a binary language whereas in EA the decision variables and objective function are used directly. As coding is not necessary, EAs are less complex and easier to implement for solving complicated optimization problems. Cabrera et al. [8] used GAs applied to a four-bar linkage in a path generation problem. Some years later Cabrera et al. [9] used EAs to solve more complex problems in the design of mechanisms. In this case a multiobjective problem is formulated including mechanical advantage in the objective function as a design requirement. In [12] a genetic algorithm is used for the Pareto optimum synthesis of a four-bar linkage considering the minimization of two objective functions simultaneously.

Hybrid algorithms with application to the synthesis of linkages have been studied in recent years. Lin [13] developed an evolutionary algorithm by combining differential evolution and the real-valued genetic algorithm. Khorshidi [14] developed a hybrid approach where a local search is employed to accelerate the convergence of the algorithm. However, these methods are limited to the four-bar mechanism and their application is restricted to path generation problems.

The objective function is based on the synthesis error estimation. The most widely used error estimator in the literature is Mean Square Distance (MSD). The MSD is used to

measure the difference between the desired parameters and the generated ones. However, this formulation has proven to be ineffective when seeking the optimal solution [15]. In many cases the MSD misleads the direction of the design and good linkages generated by the algorithm can be underestimated. Therefore, error estimation is of the utmost importance for deterministic and stochastic optimization. In EA the error estimator must be evaluated for each individual in each generation and for this reason the lack of accuracy could lead to poor efficiency in the optimization process. To avoid these problems Ullah and Kota [15] proposed the use of Fourier Descriptors that evaluate only the shape differences between the generated and desired paths. However, the proposed formulation is limited to closed paths in path generation problems. An energy-based error function is used in [16] where the finite element method is used to assess the synthesis error. This formulation reduces the drawbacks of MSD, but problems with the relative distance between desired and generated parameters remain.

The aim of this work is to propose a new hybrid algorithm that combines an evolutionary technique with a local search optimization approach. Some of the fundamentals in mechanism synthesis studied in this paper have been extensively discussed in the literature. However, the originality of this work lies in two aspects: the first one is the introduction of a new error estimator which accurately compares the function generated by the candidate mechanism with desired function. The second one is a novel approach based on the combination of deterministic and stochastic optimization techniques in the so-called hybrid methods. The flowchart for the optimization process is presented in the paper together with the results and conclusions.

## 2. Objective Function and Deterministic Optimization Approach

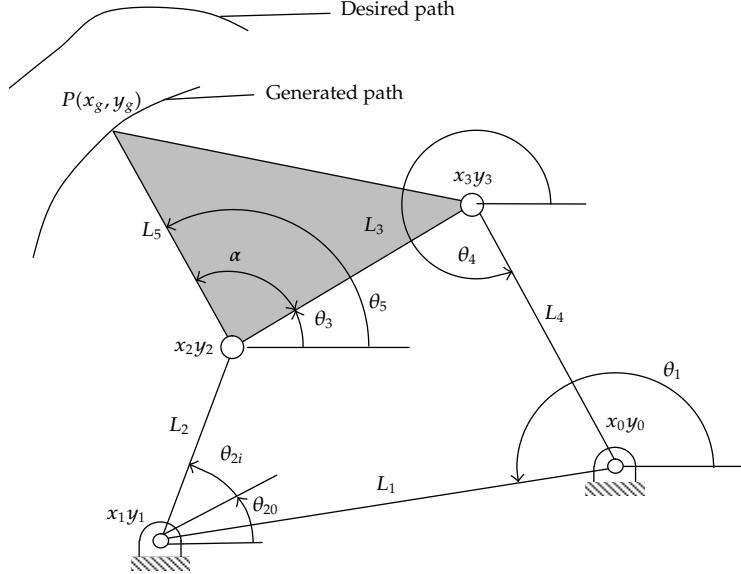
In optimal synthesis of linkages the optimization problem is defined as follows:

$$\begin{aligned} & \text{minimize } F[\mathbf{q}(\mathbf{w}), \mathbf{w}] \\ & \text{subject to } \Phi[\mathbf{q}(\mathbf{w}), \mathbf{w}] = 0, \\ & \quad \mathbf{g}[\mathbf{q}(\mathbf{w}), \mathbf{w}] \leq 0, \end{aligned} \tag{2.1}$$

where the objective function  $F[\mathbf{q}(\mathbf{w}), \mathbf{w}]$  formulates the technological requirements of the mechanism to be designed. The equality constraints  $\Phi[\mathbf{q}(\mathbf{w}), \mathbf{w}]$  formulate the kinematic restrictions during the motion, and the inequality constraints  $\mathbf{g}[\mathbf{q}(\mathbf{w}), \mathbf{w}]$  establish the limitations in the geometrical dimensions. Vector  $\mathbf{q}(\mathbf{w})$  is the vector of dependent coordinates and  $\mathbf{w}$  is the n-dimensional vector of design variables.

To illustrate the formulation the scheme of a four-bar mechanism in a path generation synthesis problem is shown in Figure 1. The proposed method can be applied effortlessly to any type of planar mechanism; however, the example in Figure 1 enables the formulation to be easily understood. The equality constraints are formulated as follows:

$$\Phi[\mathbf{q}(\mathbf{w}), \mathbf{w}]_i = \left\{ \begin{array}{l} L_1 \cos \theta_1 + L_2 \cos(\theta_{20} + \theta_{2i}) + L_3 \cos \theta_3 + L_5 \cos \theta_5 \\ L_1 \sin \theta_1 + L_2 \sin(\theta_{20} + \theta_{2i}) + L_3 \sin \theta_3 + L_5 \sin \theta_5 \\ x_g - x_0 - L_1 \cos \theta_1 - L_2 \cos(\theta_{20} + \theta_{2i}) - L_5 \cos(\theta_5 + \alpha) \\ y_g - y_0 - L_1 \sin \theta_1 - L_2 \sin(\theta_{20} + \theta_{2i}) - L_5 \sin(\theta_5 + \alpha) \end{array} \right\} = 0, \tag{2.2}$$



**Figure 1:** Scheme of the four-bar linkage.

where the vector of design variables contains the geometrical dimensions of the link. That is

$$\mathbf{w}^T = [x_0 \ y_0 \ \theta_1 \ \theta_{20} \ L_1 \ L_2 \ L_3 \ L_4 \ L_5 \ \alpha], \quad (2.3)$$

and dependent variables are defined as

$$\mathbf{q}^T = [\theta_3 \ \theta_4 \ x_g \ y_g]. \quad (2.4)$$

An important aspect in dimensional synthesis of linkages is the formulation of the goal or objective function. The objective function is capable of expressing the difference between the desired and generated paths (see Figure 1), providing an estimation of the error between the two curves irrespective of location, orientation, and size. The minimization of this function obliges the design variables to be changed and leads to the optimal dimensions of the linkage which can be expressed as  $\mathbf{w}^*$ . The generated and desired paths can be either open or closed curves. Figure 2 shows the two paths for the case of two closed curves. In this work the definition of both curves is assumed to be specified by a number of points named precision points. The precision points are selected by the designer by using vector notation and Cartesian coordinates as follows:

$$\begin{aligned} \mathbf{d}^{iT} &= [x_d^i \ y_d^i] \\ \mathbf{g}^{iT} &= [x_g^i \ y_g^i] \quad i = 1, 2, \dots, p, \end{aligned} \quad (2.5)$$

where subscript  $d$  stands for desired points,  $g$  stands for the generated ones and  $p$  is the number of precision points. Most of the works in dimensional synthesis propose the Mean Square Distance to assess the error between the two curves. That is

$$F = \frac{1}{2} \sum_{i=1}^p \left[ (\mathbf{g}^i - \mathbf{d}^i)^T (\mathbf{g}^i - \mathbf{d}^i) \right] = \frac{1}{2} [(\mathbf{g} - \mathbf{d})^T (\mathbf{g} - \mathbf{d})]. \quad (2.6)$$

The deterministic approach used in this paper is based on a local search procedure which uses first-order differentiation to obtain the search direction. In the synthesis problem the generated precision points depend on the vector of design variables and (2.6) should be rewritten as follows:

$$F[\mathbf{q}(\mathbf{w}), \mathbf{w}] = \frac{1}{2} [\mathbf{g}\{\mathbf{q}(\mathbf{w}), \mathbf{w}\} - \mathbf{d}]^T [\mathbf{g}\{\mathbf{q}(\mathbf{w}), \mathbf{w}\} - \mathbf{d}] \quad (2.7)$$

which is the objective function that must be minimized subject to the equality and inequality constraints to obtain the optimal dimensions of the mechanism. Differentiating (2.7) with respect to the design variables and equating it to zero provides

$$\nabla F[\mathbf{q}(\mathbf{w}), \mathbf{w}] = \mathbf{J}^T [\mathbf{g}\{\mathbf{q}(\mathbf{w}), \mathbf{w}\} - \mathbf{d}] = 0, \quad (2.8)$$

where  $\mathbf{J}$  is the Jacobian that can be expressed as,

$$\mathbf{J} = \frac{\partial \mathbf{g}[\mathbf{q}(\mathbf{w}), \mathbf{w}]}{\partial \mathbf{w}} = \frac{\partial \mathbf{g}[\mathbf{q}(\mathbf{w}), \mathbf{w}]}{\partial \mathbf{q}(\mathbf{w})} \frac{\partial \mathbf{q}(\mathbf{w})}{\partial \mathbf{w}}. \quad (2.9)$$

With the aim of greater clarity hereafter the dependence on the variables is omitted. The term between brackets in (2.8) can be expanded using Taylor series expansion as

$$\nabla F \approx \mathbf{J}^T \mathbf{g} - \mathbf{J}^T \mathbf{d} + \mathbf{J}^T \mathbf{J} \Delta \mathbf{w} = 0. \quad (2.10)$$

From (2.10) a recursive formula can be obtained as follows:

$$\mathbf{w}_{j+1} = \mathbf{w}_j - \alpha \mathbf{J}^T \mathbf{J} [\mathbf{J}^T \mathbf{g} - \mathbf{J}^T \mathbf{d}]. \quad (2.11)$$

In this formula the stepsize  $\alpha$  has been included in order to control the distance along the search direction. In [6] the determination of the stepsize and the exact Jacobian is described.

Differentiation of equality constraints given in (2.1) using the chain rule yields

$$\frac{\partial \Phi}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial \mathbf{w}} + \frac{\partial \Phi}{\partial \mathbf{w}} = \mathbf{0}. \quad (2.12)$$

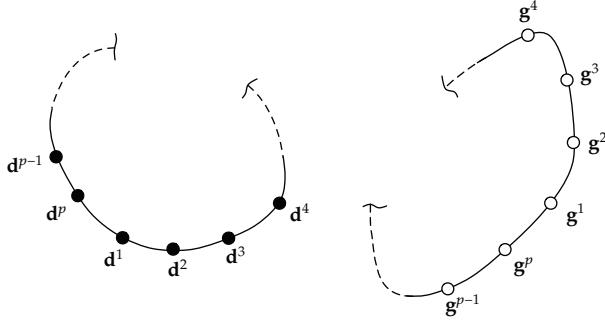


Figure 2: Desired ( $\mathbf{d}$ ) and generated ( $\mathbf{g}$ ) closed paths.

Thus, (2.9) can be rewritten as

$$\mathbf{J} = -\frac{\partial \mathbf{g}}{\partial \mathbf{q}} \left( \frac{\partial \Phi}{\partial \mathbf{q}} \right)^{-1} \frac{\partial \Phi}{\partial \mathbf{w}}. \quad (2.13)$$

All terms in (2.13) can be easily obtained from the objective function and constraints, and they enable the exact Jacobian to be determined for use in the deterministic optimization method.

If there are inequality constraints in the optimization problem, they can be converted to equality constraints through the addition of so-called slack variables. That is

$$g_i(\mathbf{w}) + v_i^2 = 0. \quad (2.14)$$

In this way, each inequality constraint adds a new variable that must be included in the formulation.

### 3. New Error Estimator for EA Algorithms

In EA, lack of accuracy in the error estimation could lead to overestimation of the error and removal of good individuals from the optimization process. On the other hand, underestimation of the error could lead to selecting individuals who are not better adapted than others in fulfilling the goal.

Equation (2.7) is used in many works as the objective function [4–9]. It has been widely used in deterministic approaches, but it is also used in probabilistic optimization. However, the function itself is an estimator of the error, not a representation of the actual error. This function depends on the relative position of the two curves and under certain circumstances the approximation may not be good enough. For instance, in the case shown in Figure 2 the error given by (2.7) can be increased or decreased if the generated curve is translated closer to the desired one or away from it, respectively. Moreover, rotation and scaling can be added to the transformation in order to reduce the error. For practical applications in engineering the translation of the curve only entails the translation of the linkage even as the rotation only needs to change the mechanism orientation. The lack of accuracy of (2.7) can be reduced by selecting the appropriate initial guess linkage in local optimization. However, in EA this option is not available and it should be solved using other strategies.

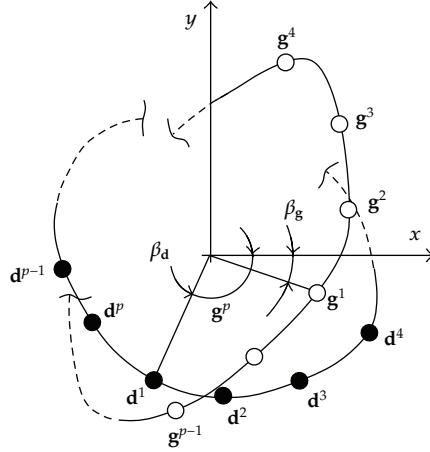


Figure 3: Translation of the desired ( $\mathbf{d}$ ) and generated ( $\mathbf{g}$ ) paths.

Thus, one can say that the error between two curves is minimum if they are compared by the translation, rotation, and scaling. Therefore, (2.7) could underestimate the error unless some transformations are introduced. The first transformation consists of the translation of the generated curve towards the desired one. To do that, the geometric centroids of both curves are determined by using the precision points as follows:

$$\begin{aligned}\mathbf{d}_c &= \frac{1}{p} \sum_{i=1}^p \mathbf{d}^i, \\ \mathbf{g}_c &= \frac{1}{p} \sum_{i=1}^p \mathbf{g}^i,\end{aligned}\tag{3.1}$$

where  $\mathbf{d}_c$  and  $\mathbf{g}_c$  are the coordinates of the geometric centroids for desired and generated curves, respectively. The new coordinates of the precision points for the two paths are obtained by translating the geometric centroids to the origin of the reference frame. That is

$$\begin{aligned}\mathbf{d}_0^i &= \mathbf{d}^i - \mathbf{d}_c, \\ \mathbf{g}_0^i &= \mathbf{g}^i - \mathbf{g}_c.\end{aligned}\tag{3.2}$$

Figure 3 shows the translation of both curves. Thus, the error estimation can be reformulated in the following way:

$$E_0 = \frac{1}{2} \sum_{i=1}^p \left[ (\mathbf{g}_0^i - \mathbf{d}_0^i)^T (\mathbf{g}_0^i - \mathbf{d}_0^i) \right].\tag{3.3}$$

Obviously, (3.3) reduces the error and is more accurate than (2.7). However, it should be pointed out that it still depends on the order chosen for numbering the precision points. In

other words, (3.3) provides a comparison of the precision points with the same superscript, which depends on the arbitrary choice previously made by the designer. Therefore, it could be possible to reduce the error when the order of numbering is changed. Therefore, removing the effect of the numbering requires the formulation of  $p$  error estimators. For the case of two closed curves, as shown in Figure 3, the following matrix can be written:

$$\begin{bmatrix} \mathbf{g}_0^1 - \mathbf{d}_0^1 & \mathbf{g}_0^2 - \mathbf{d}_0^1 & \cdots & \mathbf{g}_0^{p-1} - \mathbf{d}_0^1 & \mathbf{g}_0^p - \mathbf{d}_0^1 \\ \mathbf{g}_0^2 - \mathbf{d}_0^2 & \mathbf{g}_0^3 - \mathbf{d}_0^2 & \cdots & \mathbf{g}_0^p - \mathbf{d}_0^2 & \mathbf{g}_0^1 - \mathbf{d}_0^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{g}_0^{p-1} - \mathbf{d}_0^{p-1} & \mathbf{g}_0^p - \mathbf{d}_0^{p-1} & \cdots & \mathbf{g}_0^{p-3} - \mathbf{d}_0^{p-1} & \mathbf{g}_0^{p-2} - \mathbf{d}_0^{p-1} \\ \mathbf{g}_0^p - \mathbf{d}_0^p & \mathbf{g}_0^1 - \mathbf{d}_0^p & \cdots & \mathbf{g}_0^{p-2} - \mathbf{d}_0^p & \mathbf{g}_0^{p-1} - \mathbf{d}_0^p \end{bmatrix}_{p \times p}. \quad (3.4)$$

Each column of (3.4) gives the terms of the error estimator for each possible combination. Thus, each error estimator can be formulated as the summatory function defined by

$$F_j = \sum_{i=1}^{p-j+1} (\mathbf{g}_0^{i+j-1} - \mathbf{d}_0^i)^2 + \sum_{i=p-j+2}^p (\mathbf{g}_0^{i+j-p-1} - \mathbf{d}_0^i)^2; \quad j = 1, 2, \dots, p, \quad (3.5)$$

where subscript  $j$  stands for the estimator index. Therefore,  $F_j$  is a single-valued function providing the estimation of the error. A vector can be formulated with all the values given by (3.5)

$$\mathbf{F}^T = [F_1 F_2 \cdots F_p]. \quad (3.6)$$

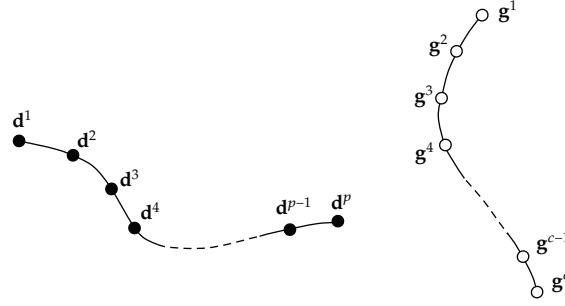
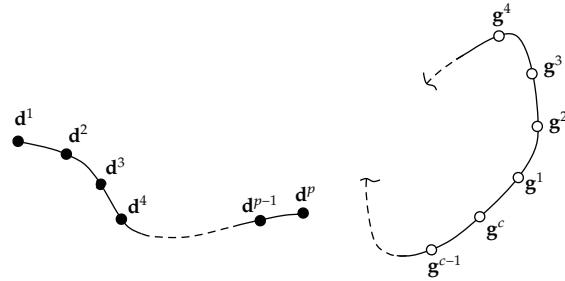
Only one of the terms in this vector provides the minimum error and will be selected to form the objective function. That is

$$F_m = \min(\mathbf{F}). \quad (3.7)$$

The matrix given by (3.4) and the summatory given by (3.5) are only valid for the comparison of closed-closed curves. However, it is possible to have two other situations: open-open or open-closed paths. The former case is shown in Figure 4, while the latter is shown in Figure 5. In both cases the number of precision points may be different for the desired and generated curves. Thus, the precision points are redefined as follows:

$$\begin{aligned} \mathbf{d}^{iT} &= [x_d^i \ y_d^i]; \quad i = 1, 2, \dots, p \\ \mathbf{g}^{rT} &= [x_g^r \ y_g^r]; \quad r = 1, 2, \dots, c \quad c \geq p, \end{aligned} \quad (3.8)$$

where  $c$  is the number of precision points for the generated curve. Similarly to the closed-closed case, the centroid of the precision points is determined and the curves are translated

Figure 4: Desired ( $\mathbf{d}$ ) and generated ( $\mathbf{g}$ ) open paths.Figure 5: Desired open path ( $\mathbf{d}$ ) and generated closed path ( $\mathbf{g}$ ).

to the origin of the reference frame. The possible combinations that allow the estimation of the error are given by the following matrix:

$$\mathbf{E} = \begin{bmatrix} \mathbf{g}_0^1 - \mathbf{d}_0^1 & \mathbf{g}_0^2 - \mathbf{d}_0^1 & \cdots & \mathbf{g}_0^{c-p} - \mathbf{d}_0^1 & \mathbf{g}_0^{c-p+1} - \mathbf{d}_0^1 \\ \mathbf{g}_0^2 - \mathbf{d}_0^2 & \mathbf{g}_0^3 - \mathbf{d}_0^2 & \cdots & \mathbf{g}_0^{c-p+1} - \mathbf{d}_0^2 & \mathbf{g}_0^{c-p+2} - \mathbf{d}_0^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{g}_0^{p-1} - \mathbf{d}_0^{p-1} & \mathbf{g}_0^p - \mathbf{d}_0^{p-1} & \cdots & \mathbf{g}_0^{c-2} - \mathbf{d}_0^{p-1} & \mathbf{g}_0^{c-1} - \mathbf{d}_0^{p-1} \\ \mathbf{g}_0^p - \mathbf{d}_0^p & \mathbf{g}_0^{p+1} - \mathbf{d}_0^p & \cdots & \mathbf{g}_0^{c-1} - \mathbf{d}_0^p & \mathbf{g}_0^c - \mathbf{d}_0^p \end{bmatrix}_{p \times (c-p+1)}. \quad (3.9)$$

The sum of the squared elements of each column in (3.9) leads to

$$F_j = \sum_{i=1}^p (\mathbf{g}_0^{i+j-1} - \mathbf{d}_0^i)^2; \quad j = 1, 2, \dots, c-p+1. \quad (3.10)$$

Equation (3.10) provides the different error estimators and the minimum value given by this formula is selected as the objective function.

When the desired path is an open curve and the generated path is a closed curve (see Figure 5), the aforementioned process can be used. However, the error estimator should be adapted to this situation. That is

$$\mathbf{E} = \begin{bmatrix} \mathbf{g}_0^1 - \mathbf{d}_0^1 & \mathbf{g}_0^2 - \mathbf{d}_0^1 & \cdots & \mathbf{g}_0^{c-1} - \mathbf{d}_0^1 & \mathbf{g}_0^c - \mathbf{d}_0^1 \\ \mathbf{g}_0^2 - \mathbf{d}_0^2 & \mathbf{g}_0^3 - \mathbf{d}_0^2 & \cdots & \mathbf{g}_0^c - \mathbf{d}_0^2 & \mathbf{g}_0^1 - \mathbf{d}_0^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{g}_0^{p-1} - \mathbf{d}_0^{p-1} & \mathbf{g}_0^p - \mathbf{d}_0^{p-1} & \cdots & \mathbf{g}_0^{p-3} - \mathbf{d}_0^{p-1} & \mathbf{g}_0^{p-2} - \mathbf{d}_0^{p-1} \\ \mathbf{g}_0^p - \mathbf{d}_0^p & \mathbf{g}_0^{p+1} - \mathbf{d}_0^p & \cdots & \mathbf{g}_0^{p-2} - \mathbf{d}_0^p & \mathbf{g}_0^{p-1} - \mathbf{d}_0^p \end{bmatrix}_{p \times c}. \quad (3.11)$$

Thus, the error estimators can be formulated as follows:

$$\begin{aligned} F_j &= \sum_{i=1}^p (\mathbf{g}_0^{i+j-1} - \mathbf{d}_0^i)^2, \quad 1 \leq j \leq c-p+1, \\ F_j &= \sum_{i=1}^{c-j+1} (\mathbf{g}_0^{i+j-1} - \mathbf{d}_0^i)^2 + \sum_{i=c-j+2}^p (\mathbf{g}_0^{i+j-c-1} - \mathbf{d}_0^i)^2, \quad c-p+1 \leq j \leq c. \end{aligned} \quad (3.12)$$

Equations (3.5), (3.10), and (3.12) provide a better comparison because they remove the effect of the translation and avoid the influence of the numbering. However, the error estimation can be enhanced by rotation and scaling. Indeed, if the generated curve is rotated and scaled with respect to the desired one, the difference between the two curves could be reduced. To do this, two new parameters must be introduced. The first one is a reference angle which provides the orientation of each curve. In Figure 2 the orientation angles are given by  $\beta_d$  and  $\beta_g$ . In practical design of mechanisms the modification of  $\beta_g$  implies the rotation of the whole linkage in the plane, which is allowed for most of the cases. The second parameter is the scaling factor  $s$ . This parameter allows the generated curve to be expanded or contracted to reduce the difference with respect to the desired path. For the case of closed-closed curves the introduction of the rotation and scaling factor in the formulation modifies equations as follows:

$$F_m(\beta_g, s) = \sum_{i=1}^{p-j+1} [s \mathbf{A} \mathbf{g}_0^{i+j-1} - \mathbf{d}_0^i]^2 + \sum_{i=p-j+2}^p (s \mathbf{A} \mathbf{g}_0^{i+j-p-1} - \mathbf{d}_0^i)^2; \quad j = 1, 2, \dots, p, \quad (3.13)$$

where

$$\mathbf{A}(\beta_g) = \mathbf{A} = \begin{bmatrix} \cos \beta_g & -\sin \beta_g \\ \sin \beta_g & \cos \beta_g \end{bmatrix}, \quad (3.14)$$

is the rotation matrix and provides the rotation of the generated precision points.

The error estimator given by (3.13) is now the objective function in a local optimization subproblem with two variables,  $\beta_g$  and  $s$ . This optimization subproblem attempts to find the best orientation and size of the generated curve (and also the linkage) in order to reduce the

error with respect to the desired path. The objective functions for the open-open curves may be readily derived as

$$F_m(\beta_g, s) = \sum_{i=1}^p (s \mathbf{A} \mathbf{g}_0^{i+j-1} - \mathbf{d}_0^i)^2; \quad j = 1, 2, \dots, c-p+1, \quad (3.15)$$

and (3.12) for the open-closed curves becomes

$$\begin{aligned} F_m(\beta_g, s) &= \sum_{i=1}^p (s \mathbf{A} \mathbf{g}_0^{i+j-1} - \mathbf{d}_0^i)^2, \quad 1 \leq j \leq c-p+1, \\ F_m(\beta_g, s) &= \sum_{i=1}^{c-j+1} (s \mathbf{A} \mathbf{g}_0^{i+j-1} - \mathbf{d}_0^i)^2 + \sum_{i=c-j+2}^p (s \mathbf{A} \mathbf{g}_0^{i+j-c-1} - \mathbf{d}_0^i)^2, \quad c-p+1 \leq j \leq c. \end{aligned} \quad (3.16)$$

These expressions might suggest that the problem must be solved as an optimization with two variables. However, the authors' experience shows that better results are obtained when the problem is solved independently for each variable. In other words, the results obtained are very accurate when the rotation optimization problem is solved before the scaling problem.

In summary, the aforementioned transformations are the core of the comparison between the desired curve and the candidate, avoiding the influence of location, orientation, and size all at once. This provides an important contribution that improves the efficiency in the exploration of the search space when using evolutionary algorithms.

## 4. Hybrid Approach for the Synthesis of Mechanisms

The design space of linkages contains a large number of local minima. Deterministic approaches based on local optimization start from a random point converging to the nearest local minimum. Thus, the solution may be an unsatisfactory solution because the design space is not sufficiently explored. The strength of stochastic optimization approaches lies in searching the entire design space of the design variables in order to locate a region with the lowest values of the objective function. This region probably contains the global minimum. However, the cost of the computational time required to achieve the convergence by using EA could be very expensive when an accurate solution is demanded. Local search approaches need less time to achieve solutions, but the accuracy depends on the quality of the initial guess. To ensure convergence and enhance its ratio hybrid methods combine the benefits of both techniques. The main advantages expected from this approach are the generality and total independence of the initial guess. The evolutionary process for searching among the optima is briefly outlined below.

### 4.1. Evolutionary Strategy

It should be highlighted that the efficiency of an evolutionary algorithm is given by both the quality of the objective function and the structure of the chromosomes and their genes. In this work the objective function is formulated as was described in the previous section. The

chromosomes are encoded using real-valued genes instead of a binary code because several works [9, 13] have demonstrated the advantages of this procedure in the design of linkages. Thus, each gene gives the real value of a design variable in the mechanism to be synthesized and all genes are grouped in a chromosome which in classical optimization is known as the vector of design variables. That is

$$\mathbf{w}_{r,g}^T = [w_{1,g} \ w_{2,g} \ \cdots \ w_{m,g}]; \quad r = 1, 2, \dots, r_{\max}, \quad (4.1)$$

where  $m$  represents the dimensionality of  $\mathbf{w}_{r,g}$ ,  $g$  is the generation subscript, and  $r_{\max}$  is the number of individuals in each generation. The dimension of  $\mathbf{w}$  is given by the type of mechanism to be synthesized and the kind of coordinates used in their definition. In this work natural coordinates are used for this purpose, as well as in the definition of the generated and desired paths. The starting and successive populations are randomly generated:

$$\mathbf{P}_g = \mathbf{w}_{r,g}; \quad r = 1, 2, \dots, r_{\max}; \quad g = 1, 2, \dots, g_{\max}, \quad (4.2)$$

where  $g_{\max}$  is the number of generations. In this work  $r_{\max}$  does not change during the optimization process so the population neither increases nor decreases. After a generation has been created, the fitness of each individual is evaluated in order to sort them for the selection. The evaluation of the fitness depends on the type of curves involved in the problem, selecting (3.13), (3.15) or (3.16) according to the case. The algorithm uses an elitism strategy in order to preserve the best individuals for the next generation. To obtain the number of best individuals an elitism factor,  $ef$ , is used as follows:

$$nE = \text{Round } (ef \ r_{\max}), \quad (4.3)$$

where  $nE$  is the number of individuals whose genetic information is preserved for the following generation. After that, the tournament selection starts and the parents are chosen for reproduction. The first step in reproduction is to establish the number of offspring generated by the crossover, whose valued is given by the following formula:

$$nC = \text{Round } [\text{rf}(r_{\max} - nE)], \quad (4.4)$$

where  $nC$  is the number of offspring generated by the crossover operator and  $\text{rf}$  is the reproduction factor. Mutation is another operator used to change the genes randomly during the reproduction. The number of offspring affected by mutation is given by

$$nM = r_{\max} - nE - nC. \quad (4.5)$$

Thus, the number of parents is twice the number of offspring selected for crossover plus the number of individuals selected for mutation. To decide whether or not it should become a member for reproduction, the roulette wheel method [8] is used for the selection of parents from the complete population. The number of slots in the roulette is equal to the number of individuals and the size of the slots is equal to their expectation. Once the parents are selected, crossover is used to increase the diversity of the individuals in the complete population.

Crossover generates the offspring by taking genetic information from the two parents. The chromosomes of the descendants are obtained using the arithmetic mean of the same genes taken from each parent using a random coefficient with normal distribution. The mutation operator is controlled by two coefficients. The first one is the scale,  $sM$ , which controls the range of the variation allowed in the genes. The second one is the shrink coefficient,  $hM$ . This coefficient allows mutations with a wider interval of variation in the first generations but gradually reduces this interval in the following generations. In this way the algorithm provides exploration and exploitation of the global optimum and maintains a suitable balance during the optimization process. The authors have verified that the control of the shrink coefficient is fundamental to obtain the optimal solution when the range of the design variables is very different.

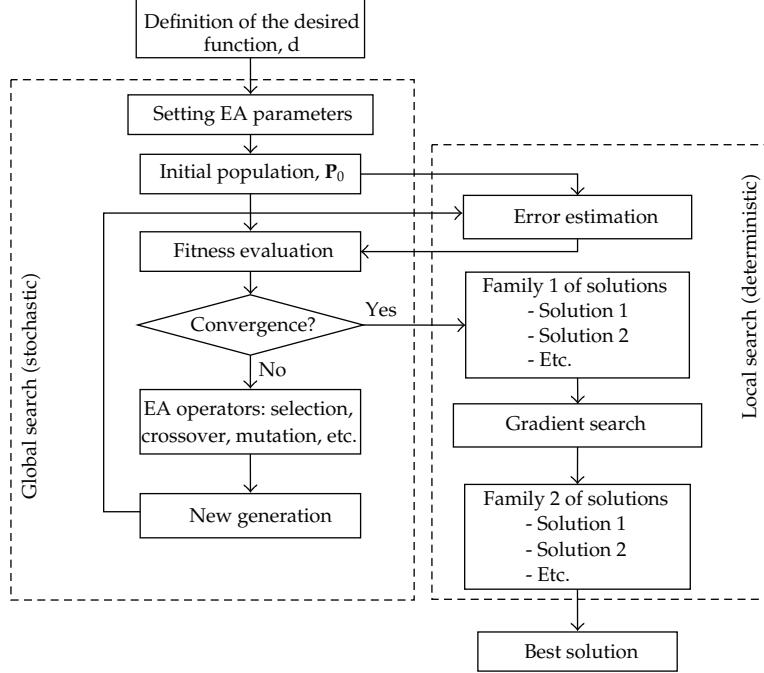
After the reproduction has finished for one generation, the new generation is evaluated by using the fitness value for every individual and the same process is repeated until the convergence of the evolutionary algorithm is achieved. Different convergence criteria may be used to stop the algorithm. The first one is based on the accuracy obtained for the best individual in the last generation, but a limit in the number of generations is also established to stop the process. Once the convergence is achieved, the fitness of the last generation is evaluated and a family of best individuals is obtained. The family of best individuals is selected from those linkages whose fitness value is below a threshold. This family of linkages is used as the initial guess for the deterministic approach to form the hybridization process which is described in the following subsection.

#### **4.2. Hybrid Algorithm**

Figure 6 shows the flowchart of the hybrid algorithm including the stochastic and deterministic optimization. On the left-hand side of Figure 6, the scheme of the evolutionary technique is shown. The right side in the same figure shows the deterministic part of the hybrid algorithm. The algorithm starts with the definition by the designer of the desired function based on the required motion for the linkage. The designer also establishes the EA parameters that will be used in the algorithm (e.g. the operators for selection, crossover, etc.). After that the optimization process starts with the generation of the first population. The fitness evaluation of this first generation requires the estimation of the error by using deterministic optimization to obtain the orientation angle,  $\beta_g$ , and scaling factor,  $s$ . If the fitness value is below a threshold, a family of linkages is selected to be optimized by the deterministic approach. This rarely occurs in the first generation and several generations are necessary to cross from the probabilistic approach to the deterministic optimization as is shown in Figure 6. The deterministic approach uses the best individuals selected from the evolutionary algorithm which are called Family 1. These individuals are optimized irrespective of their fitness values because local optimization could lead to obtaining better individuals among those with worse initial expectation. The deterministic approach optimizes each individual independently to obtain a second family called Family 2. The solution is selected as the best linkage of this second family.

### **5. Numerical Examples**

In this section two examples are presented in order to demonstrate the capacity of the hybrid algorithm. In the first example a four-bar mechanism is selected to be synthesized to



**Figure 6:** Flowchart of the hybrid algorithm.

generate a right angle path. The example does not correspond to any actual implementation in engineering design, but this type of path is a challenging objective and demonstrates the accuracy, robustness, and efficiency of the proposed approach. The second example is a practical application in the design of an actual machine. The results in these examples are divided into two stages. The first one is the result obtained by the evolutionary algorithm and the second one is the result obtained by the complete hybrid algorithm which includes the local optimization approach in the dimensional synthesis.

### 5.1. Four-Bar Linkage Generating a Right Angle Path

In this example the methodology is applied to the synthesis of a four-bar mechanism. The scheme of the mechanism is the same as that used in Section 2 (see Figure 1). Likewise the constraints and design variables are given by (2.2) and (2.3), respectively. The aim of the problem is that the coupler point,  $P$ , of the synthesized linkage describes a right angle path during the motion. The path is defined by 11 prescribed points whose coordinates are shown in the first two rows in Table 1. Table 2 shows the values of the operator factors used in the evolutionary algorithm. It is important to highlight the small size of the population and the maximum number of generations.

The best resulting mechanism and the path followed by the coupler point in the evolutionary part of the algorithm is shown in Figure 7(a), in addition to the desired precision points. The evolutionary algorithm takes 189.59 seconds to achieve the convergence with an error of  $2.439 \text{ mm}^2$  using an Intel Core i5 PC. As can be observed in this figure, the generated path approximates well to the desired one; however, there is clearly a lack of accuracy. In

**Table 1:** Desired path and the path generated at the convergence with the proposed algorithm.

Paths	1	2	3	4	5	6	7	8	9	10	11
Desired (mm)	$x_d$	0	0	0	0	0	3.00	6.00	9.00	12.00	15.00
	$y_d$	15.00	12.00	9.00	6.00	3.00	0	0	0	0	0
EA (mm)	$x_g$	0.21	0.06	0.08	-0.24	-0.29	0.82	2.96	5.70	8.77	11.92
	$y_g$	15.33	12.20	8.75	5.27	2.76	1.14	0.03	-0.59	-0.68	-0.17
Hybrid (mm)	$x_s$	0.22	-0.16	-0.18	0.04	0.11	0.00	3.00	5.99	9.00	12.01
	$y_s$	14.95	12.01	8.99	6.00	2.98	0.24	0.05	-0.17	-0.26	-0.11

**Table 2:** Values of the different factors used in the evolutionary algorithm.

EA factors	$r_{\max}$	$g_{\max}$	ef	rf	sM	hM
Values	150	10	0.02	0.8	0.8	0.4

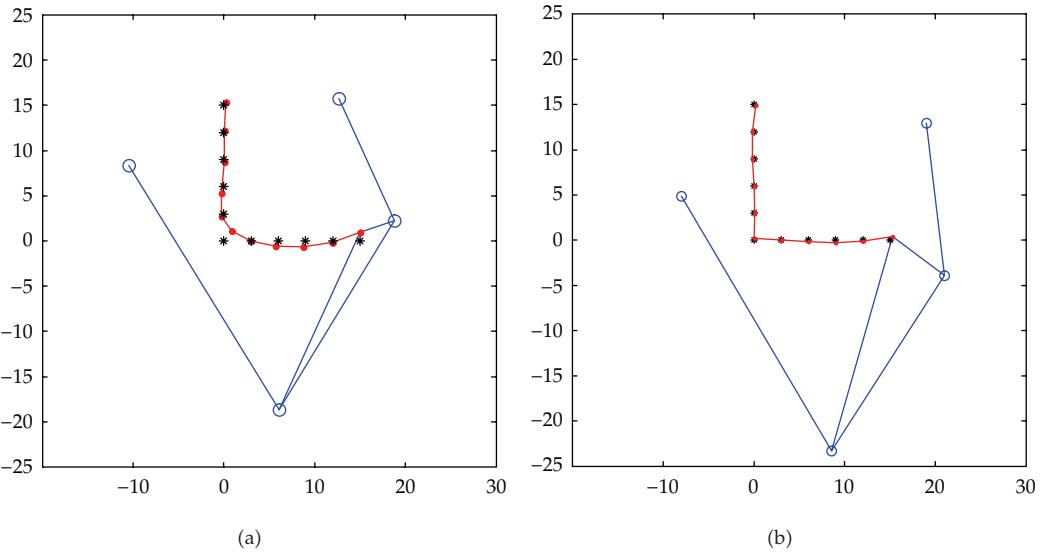
order to compare the result with the desired path, the third and fourth rows of Table 1 show the coordinates of the generated points. The solution for the hybrid algorithm is shown in Figure 7(b) where the path followed by the coupler point fits very well with the desired one. In the last two rows of Table 1 the coordinates of the generated path are shown and the last row of Table 3 shows the values for the design variables.

The error at convergence is  $0.2025 \text{ mm}^2$  and the time necessary to achieve the convergence was 212 seconds, which is a very reasonable computational cost in this kind of problem.

Since it is stochastic, the results differ each time the algorithm runs. In order to evaluate the robustness, the algorithm was run 30 times and the sample mean error obtained at convergence was  $0.309 \text{ mm}^2$  with a sample standard deviation of  $0.211 \text{ mm}^2$ . The sample mean CPU time to achieve the convergence was 215.05 seconds with a standard deviation of 19.031 seconds.

## 5.2. Application to a Mechanism for Injection Machine

In this example the methodology has been applied to the design of a mechanism for die-cast injection machine. Figure 8(a) shows the scheme of such a machine, where the system for the injection of zamak alloys is shown at the top. The mould is located below the injection system (not shown in the figure). The system for the displacement of the mould is shown on the left-hand side of the figure. Figure 8(b) shows the detail of the injection system where it is possible to see the linkage used for this purpose. The mechanism selected for this application is a combination of a four-bar linkage together with a slider-crank mechanism connected by the coupler link. The motion of the slider follows a straight line pushing the zamak alloy through the entrance to fill the mould. This motion must be controlled in order to fill the mould adequately. To obtain good quality in the manufacturing process a rapid, motion of the slider is necessary initially, then a slower motion, and finally a fast backward motion when the mould has been filled. This motion of the slider is coordinated with the input link which is driven by an electric motor with constant velocity (see Figure 8(b)). The precision points are set every 18 deg of the motor rotation, or in other words, 20 precision points are selected for a full rotation of the motor. The coordinates of the precision points are shown in Table 4 and the desired motion is dotted in Figure 9.



**Figure 7:** (a) Solution with the evolutionary algorithm and (b) with the hybrid algorithm.

**Table 3:** Design variables.

Design variables	$x_0$ (mm)	$x_0$ (mm)	$\theta_1$ (rad)	$\theta_{20}$ (rad)	$L_1$ (mm)	$L_2$ (mm)	$L_3$ (mm)	$L_4$ (mm)	$L_5$ (mm)	$\alpha$ (rad)
EA solution	-10.47	8.34	0.31	1.68	24.30	14.81	24.48	31.70	4.02	-0.68
Hybrid solution	-8.01	4.91	0.28	5.50	28.16	16.94	23.05	32.68	7.13	-1.63

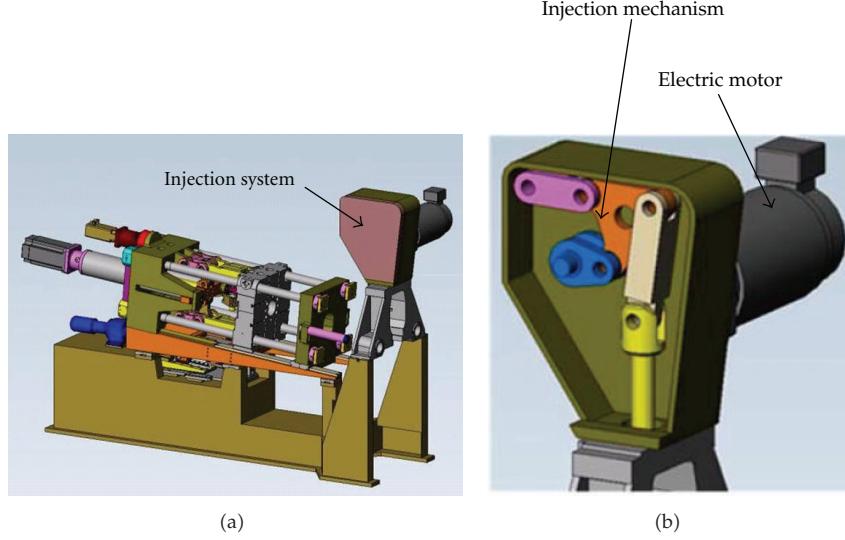
The scheme of the mechanism to be synthesized is shown in Figure 10 together with the twelve design variables. Figure 9 compares the results for the evolutionary algorithm and the hybrid optimization approach and Table 4 gives the values of the coordinates generated in all cases. Finally, Table 5 shows the values of the design variables at convergence for the evolutionary algorithm and the hybrid algorithm.

Despite of the difficulty of the problem, the graphical results in Figure 9 show that the evolutionary algorithm provides good accuracy in general; however, in the central part of the curve the accuracy is lower. The hybrid algorithm enhances the accuracy in this zone and provides a very good solution.

The sample mean error obtained by the hybrid algorithm is  $357.70 \text{ mm}^2$  with a standard deviation of  $14.07 \text{ mm}^2$ . The mean CPU time to achieve the convergence is 623.17 seconds with a standard deviation of 46.40 seconds.

## **6. Concluding Remarks**

In this paper a hybrid optimization approach has been presented with application to the optimal dimensional synthesis of planar mechanisms. The objective function is selected using a new error estimator defined by means of the precision points. This error estimator enables the evaluation of the fitness of the function without influence of translation, rotation, and scaling effects. The error estimation is done using a local optimization procedure

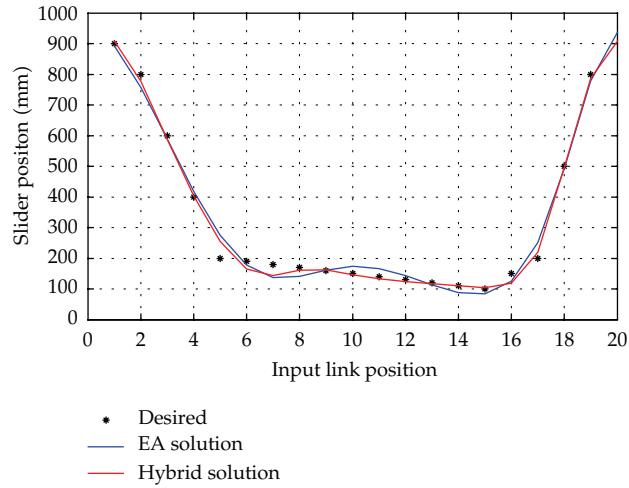


**Figure 8:** (a) Injection moulding machine and (b) detail of the injection system.

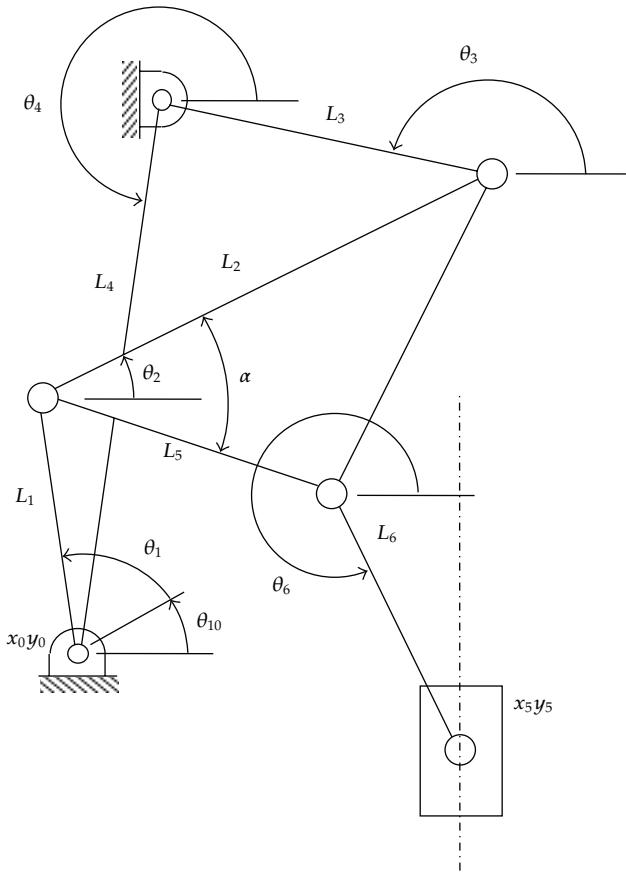
**Table 4:** Desired path and the path generated at convergence with the proposed algorithm.

	Desired $y_d$ (mm)	EA solution $y_g$ (mm)	Hybrid solution $y_s$ (mm)
1	900	892.33	908.6
2	800	757.53	776.8
3	600	588.30	587.4
4	400	417.67	402.7
5	200	274.19	255.0
6	190	178.37	165.4
7	180	137.35	143.2
8	170	140.40	161.5
9	160	161.43	161.6
10	150	173.98	146.8
11	140	166.79	133.0
12	130	143.37	123.9
13	120	113.29	117.4
14	110	88.23	110.3
15	100	83.79	104.2
16	150	125.08	118.4
17	200	250.88	219.1
18	500	491.63	496.4
19	800	777.75	785.5
20	900	937.63	908.4

providing a very efficient hybrid algorithm. The hybrid algorithm combines the advantages of both stochastic and deterministic approaches to improve the robustness and accuracy. Two examples have been presented in the paper to demonstrate the capacity of the method. The examples show that the proposed method not only achieves the convergence but also demonstrates how the accuracy is improved by the combination of the two procedures.



**Figure 9:** Desired motion, EA, and hybrid solution.



**Figure 10:** Scheme of the mechanism for injection.

**Table 5:** Design variables.

Design variables	$x_0$ (mm)	$y_0$ (mm)	$\theta_{10}$ (rad)	$\theta_4$ (rad)	$L_1$ (mm)	$L_2$ (mm)	$L_3$ (mm)	$L_4$ (mm)	$L_5$ (mm)	$L_6$ (mm)	$\alpha$ (rad)	$x_5$ (mm)
EA solution	810.92	311.51	2.745	-1.23	517.03	992.82	991.66	1227.4	1161.7	722.41	-0.286	100
Hybrid solution	537.39	393.31	2.785	-1.4	416.52	603.75	554.85	724.41	611.08	372.27	0.2437	100

To do this, the examples depict the solution for the case of the evolutionary algorithm working alone, and then the solution improved by the hybrid algorithm. This shows how the evolutionary algorithm provides an approximation to the solution and then the local optimization improves the accuracy. In both examples the solution provides good designs and the generated curves fit very well with the desired ones. In summary, the hybrid algorithm is a valuable tool for the design of mechanisms when highly demanding requirements are imposed. Thus, the conclusion we draw is that the appropriate combination of stochastic and deterministic algorithms has an enormous potential in the more effective solution of optimization problems in the design of mechanisms. This work will be further developed for the solution of other mechanism design problems by adapting the algorithm. Furthermore, another future task in this field aims to improve the efficiency of the hybrid optimizer by using the most recent developments in metaheuristic approaches such as Particle Swarm Optimization and Differential Evolution.

## Acknowledgment

This paper has been developed in the framework of the Project DPI2010-18316 funded by the Spanish Ministry of Economy and Competitiveness.

## References

- [1] F. Freudenstein, "An analytical approach to the design of four-link mechanisms," *Transactions of the ASME*, vol. 76, pp. 483–492, 1954.
- [2] G. N. Sandor, *A general complex number method for plane kinematics synthesis with applications [Ph.D. thesis]*, Columbia University, New York, NY, USA, 1959.
- [3] A. G. Erdman, "Three and four precision point kinematic synthesis of planar linkages," *Mechanism and Machine Theory*, vol. 16, no. 3, pp. 227–245, 1981.
- [4] S. Krishnamurty and D. A. Turcic, "Optimal synthesis of mechanisms using nonlinear goal programming techniques," *Mechanism and Machine Theory*, vol. 27, no. 5, pp. 599–612, 1992.
- [5] J. Mariappan and S. Krishnamurty, "A generalized exact gradient method for mechanism synthesis," *Mechanism and Machine Theory*, vol. 31, no. 4, pp. 413–421, 1996.
- [6] R. Sancibrian, P. Garcia, F. Viadero, and A. Fernández, "A general procedure based on exact gradient determination in dimensional synthesis of planar mechanisms," *Mechanism and Machine Theory*, vol. 41, no. 2, pp. 212–229, 2006.
- [7] A. Kunjur and S. Krishnamurty, "Genetic algorithms in mechanical synthesis," *Journal of Applied Mechanism and Robotics*, vol. 4, no. 2, pp. 18–24, 1997.
- [8] J. A. Cabrera, A. Simon, and M. Prado, "Optimal synthesis of mechanisms with genetic algorithms," *Mechanism and Machine Theory*, vol. 37, no. 10, pp. 1165–1177, 2002.
- [9] J. A. Cabrera, F. Nadal, J. P. Muñoz, and A. Simon, "Multiobjective constrained optimal synthesis of planar mechanisms using a new evolutionary algorithm," *Mechanism and Machine Theory*, vol. 42, no. 7, pp. 791–806, 2007.
- [10] J. H. Holland, *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, Mich, USA, 1975, An introductory analysis with applications to biology, control, and artificial

intelligence.

- [11] K. A. DeJong, *An analysis of the behaviour of a class of genetic adaptive system [Ph.D. thesis]*, University of Michigan, Ann Arbor, Mich, USA, 1975.
- [12] N. Nariman-Zadeh, M. Felezi, A. Jamali, and M. Ganji, "Pareto optimal synthesis of four-bar mechanisms for path generation," *Mechanism and Machine Theory*, vol. 44, no. 1, pp. 180–191, 2009.
- [13] W. Y. Lin, "A GA-DE hybrid evolutionary algorithm for path synthesis of four-bar linkage," *Mechanism and Machine Theory*, vol. 45, no. 8, pp. 1096–1107, 2010.
- [14] M. Khorshidi, M. Soheilypour, M. Peyro, A. Atai, and M. S. Panahi, "Optimal design of four-bar mechanisms using a hybrid multi-objective GA with adaptive local search," *Mechanism and Machine Theory*, vol. 46, no. 10, pp. 1453–1465, 2011.
- [15] I. Ullah and S. Kota, "Optimal synthesis of mechanisms for path generation using fourier descriptors and global search methods," *Journal of Mechanical Design*, vol. 119, no. 4, pp. 504–510, 1997.
- [16] I. Fernández-Bustos, J. Aguirrebeitia, R. Avilés, and C. Angulo, "Kinematical synthesis of 1-dof mechanisms using finite elements and genetic algorithms," *Finite Elements in Analysis and Design*, vol. 41, no. 15, pp. 1441–1463, 2005.

*Research Article*

## **Goal-Programming-Driven Genetic Algorithm Model for Wireless Access Point Deployment Optimization**

**Chen-Shu Wang<sup>1</sup> and Ching-Ter Chang<sup>2</sup>**

<sup>1</sup> Graduate Institute of Information and Logistics Management, National Taipei University of Technology, Taipei 10608, Taiwan

<sup>2</sup> Department of Information Management, Chang Gung University, Tao Yuan 333, Taiwan

Correspondence should be addressed to Chen-Shu Wang, wangcs@ntut.edu.tw

Received 24 February 2012; Accepted 8 May 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 C.-S. Wang and C.-T. Chang. 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.

Appropriate wireless access point deployment (APD) is essential for ensuring seamless user communication. Optimal APD enables good telecommunication quality, balanced capacity loading, and optimal deployment costs. APD is a typical NP-complex problem because improving wireless networking infrastructure has multiple objectives (MOs). This paper proposes a method that integrates a goal-programming-driven model (PM) and a genetic algorithm (GA) to resolve the MO-APD problem. The PM identifies the target deployment subject of four constraints: budget, coverage, capacity, and interference. The PM also calculates dynamic capacity requirements to replicate real wireless communication. Three experiments validate the feasibility of the PM. The results demonstrate the utility and stability of the proposed method. Decision makers can easily refer to the PM-identified target deployment before allocating APs.

### **1. Introduction**

Appropriate wireless access point deployment (APD) is essential for ensuring seamless user communication. Optimal APD enables good telecommunication quality, balanced capacity loading, and optimal deployment costs. APD is a typical NP-complex problem [1] because it involves multiple decision objectives, such as budget [2–4], coverage [2, 5–8], interference [3, 4, 7], and dynamic capacity [1, 4, 6–9]. Furthermore, these objectives usually contradict each other [7]. For example, the number of APs is usually positively related to the wireless signal coverage rate and telecommunication reliability [1]. However, more APs increase deployment costs. These conflicting criteria should be considered simultaneously when solving APD problems [9, 10].

**Table 1:** A comparison of the three wireless AP types.

AP Type	Cost (NT\$)	Range (m)	Speed (Mbit/s)	Manufacturers
Type 1 802.11 b/g	1365	30 to 89	54	ASUS, D-Link, Lantech, SMC
Type 2 802.11 b/g	1710	33 to 101	108	Corega, D-Link, PCI, SMC
Type 3 802.11 b/g/n	3138	51 to 163	300	Apple, ASUS, Buffalo, Corega, D-Link

In the last decade, many studies have attempted to solve APD optimally by considering multiple objectives (MOs). There are four main objectives: budget, coverage rate, capacity, and interference. Studies have attempted to identify maximal coverage. For example, Huang et al. developed a growth-planning algorithm to establish the maximal coverage range [11]. Zhao et al. used a point-to-point signal strength strategy to implement indoor AP location optimization for maximal coverage [12]. For the capacity objective, the capacity requirements of wireless networks compared to wired networks are particularly difficult to evaluate because users are dynamic and can move from place to place. This makes APD a dynamic and complex problem. The dynamic capacity requirement must be addressed to resolve APD [13] because users can access particular APs to balance loads [9, 14]. Finally, for the interference objective, too many APs of the same type and placed too close together may cause AP malfunction because of frequency interference. To avoid communication interference, some studies [6, 15] have suggested that APs should be arranged on different communication channels, but this leads to other communication channel assignment problems.

This paper applies a goal-programming-driven model (PM) to the MO-APD problem. It uses goal programming (GP) to infer and model the PM and a genetic algorithm (GA) to search for near optimal solutions. These methods are easily applied to MO-APD problems to reflect real situations. The remainder of this paper is organized as follows: Section 2 defines the problem; Section 3 details the PM; Section 4 presents a discussion on the PM solution process using a GA; Section 5 provides the results of numerical experiments which are given in this section; lastly, Section 6 offers a conclusion and suggestions for future research.

## 2. Description of the APD Problem

This research resolves the MO-APD problem according to four decision constraints: budget, coverage, capacity requirements, and interference. The PM identifies a feasible target deployment ( $T$ ), which consists of three types of wireless APs, as shown in Table 1. This study conducted experiments and surveys that indicate that the coverage range and communication speed of a Type 3 AP are wider and faster, respectively, than AP Types 1 and 2. However, Type 1 AP equipment is cheaper than AP Types 2 and 3. Two APs may interfere with each other if they are the same type and are too near. Therefore, the PM must balance the four decision constraints and allocate three AP types in the target deployment of the APD problem. Table 2 lists the variables used in the proposed models.

**Table 2:** Variables used in this paper.

Variable	Definition
$B_{ij}$	The base station for the potential AP deployment area, where $i$ is a row and $j$ is a column. $i = 1, 2, \dots, n; j = 1, 2, \dots, m$ .
$x_q$	Represents the type $q$ AP allocated to $B_{ij}$ , where $q \in [1, 2, 3]$ in Table 1. If $x_q = 0$ , no AP has been allocated to $B_{ij}$ .
$c_{x_q}$	Represents the AP cost for each AP type, $x_q \in [1, 2, 3]$ .
$\text{req}_{ij}$	The networking capacity requirement between $B_i$ and $B_j$ .

### 2.1. The Budget Constraint ( $\theta(T)$ )

Budget is the most important APD-MO constraint that directly affects the feasibility of  $T$ . The cost function in (2.1) evaluates the total cost of  $T$ . Equation (2.2) evaluates the budget constraint. In (2.2), bgt represents the given budget constraint for the AP allocation for  $T$ :

$$\text{CST}(T) = \sum_i \sum_j C_{x_q} B_{i,j}, \quad \forall X_q \neq 0, \quad i = 1, 2, \dots, n, \quad j = 1, 2, \dots, m, \quad (2.1)$$

$$\begin{aligned} & \text{if } \text{CST}(T) < \text{bgt}, \quad \theta(T) = 1, \\ & \text{else } \theta(T) = \frac{\text{bgt}}{\text{CST}(T)}. \end{aligned} \quad (2.2)$$

### 2.2. The Coverage Constraint ( $\Phi(T)$ )

Figure 1 shows that to enable seamless user communication, two APs are allocated, but two capacity requirements ( $\text{req}_{2,1}$  and  $\text{req}_{2,3}$ ) have no signal coverage. The coverage function ( $\text{CVG}(T)$ ) evaluates the signal coverage area of  $T$ . Equation (2.3) evaluates the coverage fulfillment rate:

$$\Phi(T) = \frac{\text{CVG}(T)}{\text{target area}}. \quad (2.3)$$

### 2.3. The Capacity Constraint ( $\Psi(T)$ )

Figure 2 shows a dynamic capacity scenario. The target area allocates two Type 1 APs (AP<sub>1</sub> and AP<sub>2</sub>), and two APs simultaneously cover the capacity requirements ( $\text{req}_{22}$ ). For time slot 1 ( $T_1$ ), the capacity requirements of  $\text{req}_{12}$ ,  $\text{req}_{13}$ , and  $\text{req}_{22}$  access AP<sub>1</sub>, and the capacity requirements of  $\text{req}_{31}$  and  $\text{req}_{32}$  access AP<sub>2</sub>. Therefore, for time slot  $T_1$ , AP<sub>1</sub> and AP<sub>2</sub> must provide 55 and 25 mbit/s capacity, respectively. In time slot  $T_2$ ,  $\text{req}_{22}$  shifts connection from AP<sub>1</sub> to AP<sub>2</sub> for balance loading. Therefore, for time slot  $T_2$ , AP<sub>1</sub> and AP<sub>2</sub> must provide 35 and 45 mbit/s, respectively. Actual capacity requirements are difficult to evaluate accurately.

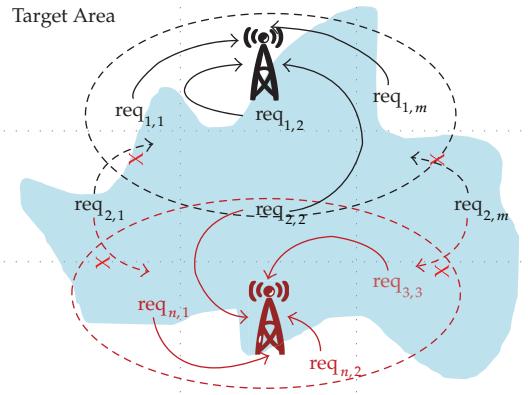


Figure 1: The signal coverage illustration.

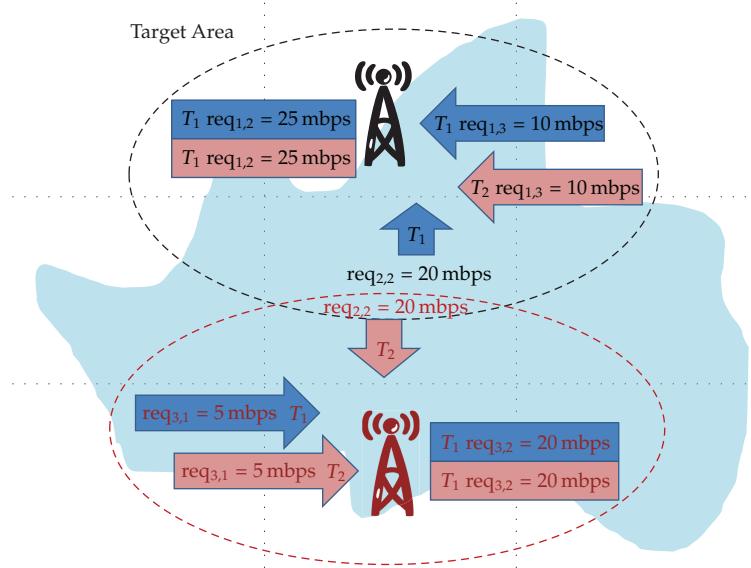
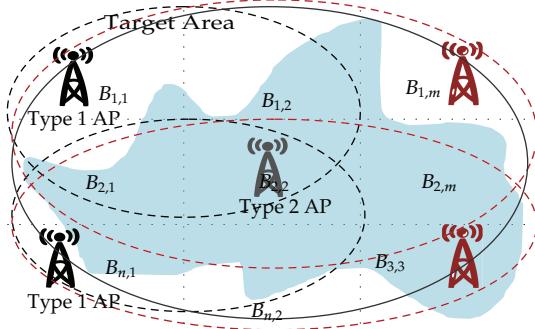


Figure 2: The dynamic capacity requirements of two APs. The blue area represents time slot 1 ( $T_1$ ) and the pink area represents time slot 2 ( $T_2$ ).

A Monte Carlo simulation algorithm—that simulates the capacity of  $T$ —implements the  $\text{DCP}_{ij}(T)$  function. Equation (2.4) evaluates the capacity fulfillment rate:

$$\begin{aligned}
 & \text{if } \text{DCP}_{ij}(T) > \text{req}_{ij}, \quad \Psi(T) = 1, \quad i = 1, 2, \dots, n; \quad j = 1, 2, \dots, m, \\
 & \text{else } \Psi(T) = \frac{\text{DCP}_{ij}(T)}{\text{req}_{ij}}.
 \end{aligned} \tag{2.4}$$



**Figure 3:** The frequency interference illustration.

#### 2.4. The Interference Constraint ( $\omega(T)$ )

Figure 3 shows that many Type 1 APs are used to maximize coverage and capacity of  $T$  because of budget constraints. However, too many APs of the same type and allocated too near to each other may lead to AP malfunction because of AP frequency interference. For example, Figure 3 shows a perfect coverage design. However, it also shows an increased interference rate. The interference function (IFT( $T$ )) evaluates the interference area of  $T$ . Equation (2.5) evaluates the interference fulfillment rate:

$$\omega(T) = 1 - \frac{\text{IFT}(T)}{\text{target area}}. \quad (2.5)$$

### 3. Proposed Model to Solve MO-APD

Two main approaches can be used to formulate MO-APD. One approach is the cost-oriented approach, which aims to minimize total cost subject to MO performance constraints. This study formulated MO-APD using the cost-oriented approach as shown in *Proposal 1* (P1). GAL-CVG, GAL-CP, and GAL-IFT are the given constraints for coverage rate, capacity fulfillment rate, and interference fulfillment rate, respectively.

(P1)

$$\text{Min CST}(T) \leq \text{bgt}, \quad (3.1)$$

subject to

$$\Phi(T) \geq \text{GAL} - \text{CVG}, \quad (3.2)$$

$$\Psi(T) \geq \text{GAL} - \text{CP}, \quad (3.3)$$

$$\omega(T) \geq \text{GAL} - \text{IFT}. \quad (3.4)$$

Equations (3.2)–(3.4) are the coverage, capacity, and interference constraints. Equation (3.1) is the objective function, which minimizes the total cost subject to multiple decision constraints (3.2)–(3.4).

The second approach is performance oriented, and it maximizes the performance of target deployment subjects to real constraints (e.g., budget). This study reformulated the MO-APD using the performance-oriented method, shown in *Proposal 2* (P2).

(P2)

$$\text{Max } \Phi(T) + \Psi(T) + \omega(T), \quad (3.5)$$

subject to (3.1)–(3.4).

Equation (3.5) is the objective function in P2, which maximizes the coverage, capacity fulfillment, and interference fulfillment rates of  $T$  subject to budgetary (3.1) and other decision constraints (3.2)–(3.4). GP aids MO decision-making problem modeling. It was first introduced by Charnes and Cooper [16] and further developed by Tamiz et al. [17], Romero [18], and Chang [19]. Various types of GP approaches exist, such as lexicographic GP, weighted GP, MINMAX (Chebyshev) GP, and multichoice GP [19]. To enable decision makers to easily set the constraint weighting according to their preferences, this study used a weighted GP approach to translate (P2) into the (PM).  $w_{\text{cvg}}$ ,  $w_{\text{cp}}$ , and  $w_{\text{IFT}}$  are the important weights (between 0 and 1) for the GAL-CVG, GAL-CP, and GAL-IFT constraints, respectively.

(PM)

$$\text{Min } W_{\text{bgt}}(\text{bgt}^+) + W_{\text{cvg}}(\text{cvg}^-) + W_{\text{cp}}(\text{cp}^-) + W_{\text{IFT}}(\text{IFT}^-), \quad (3.6)$$

subject to

$$\text{CST}(T) - \text{bgt}^+ + \text{bgt}^- = \text{bgt}, \quad (3.7)$$

$$\Phi(T) - \text{cvg}^+ + \text{cvg}^- = \text{GAL} - \text{CVG}, \quad (3.8)$$

$$\Psi(T) - \text{cp}^+ + \text{cp}^- = \text{GAL} - \text{CP}, \quad (3.9)$$

$$\omega(T) - \text{IFT}^+ + \text{IFT}^- = \text{GAL} - \text{IFT}, \quad (3.10)$$

$$\text{bgt}^+, \text{bgt}^-, \text{cvg}^+, \text{cvg}^-, \text{cp}^+, \text{cp}^-, \text{IFT}^+, \text{IFT}^- \geq 0. \quad (3.11)$$

#### 4. Process for Solving the PM Using a GA

The GA is a stochastic searching method that uses the mechanics of natural selection to solve optimization problems. The GA was developed from the theory of natural selection [20]. Because the GA is a good stochastic technique for solving combinatorial optimization problems, this study uses the GA as the PM search tool, as shown in Figure 4.

An initial solution population is randomly created. The fitness of each individual in the population then determines whether it survives. Termination criteria (such as the generation size or the fitness value exceeding the threshold) determine the target deployment ( $T$ ) to be achieved. Finally, genetic operators such as selection, crossover, and mutation identify the

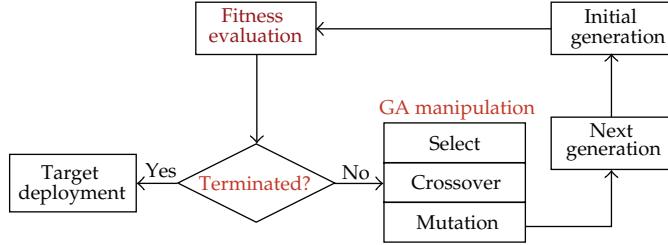


Figure 4: GA-based PM solution process.

next generation. After meeting a number of iterations or predefined criteria, a near optimal solution is found.

#### 4.1. Representation Structure: Encode/Decode

A graph represents a target deployment ( $T$ ) that can also be expressed as a two-dimensional matrix. In the graph, each potential base station ( $B_{i,j}$ ) has two states: AP ( $B_{i,j} = 1$ ) allocated and no AP ( $B_{i,j} = 0$ ) allocated. A base station with an allocated AP must have an AP type ( $x_q = [1, 2, 3]$ ).  $n \times m$  bit strings were used as chromosomes to represent  $T$ :

$$T = \begin{bmatrix} B_{1,1} & B_{1,2} & \cdots & B_{1,m} \\ B_{2,1} & B_{2,2} & \cdots & B_{2,m} \\ \vdots & \vdots & \vdots & \vdots \\ B_{n,1} & B_{n,2} & \cdots & B_{n,m} \end{bmatrix}. \quad (4.1)$$

#### 4.2. Evaluation Function

The PM objective function ( $F(i)$ ) was used as a GA evaluation function in (4.2). All variables in (4.2) are defined as in the PM:

$$F(i) = W_{\text{bgt}}(\text{bgt}^+) + W_{\text{cvg}}(\text{cvg}^-) + W_{\text{cp}}(\text{cp}^-) + W_{\text{IFT}}(\text{IFT}^-). \quad (4.2)$$

#### 4.3. GA Manipulations

- (1) *Selection*: roulette wheel selection ensures that highly fit chromosomes produce more offspring. This method selects a candidate network according to its survival probability, which is equal to its fitness relative to the whole population, as shown in (4.3):

$$\left[ \frac{F(i)}{\sum F(i)} \right]. \quad (4.3)$$

- (2) *Crossover*: the crossover method randomly selects two chromosomes from the mating pool for mating. Crossover site  $C$  is randomly selected in the interval

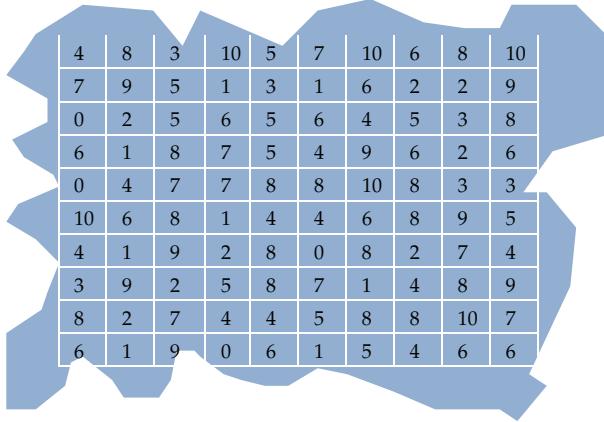


Figure 5: Capacity requirements for Experiment 1.

$[1, n \times m]$ . Two new chromosomes, called offspring, are then obtained by swapping all characters between position C and  $n \times m$ .

- (3) *Mutation:* the combined reproduction and crossover methods occasionally lose potentially useful chromosome information. Mutation is introduced to overcome this. It is implemented by randomly complementing a bit (0 to 1 and vice versa). This ensures that good chromosomes are not permanently lost.

## 5. Experiment Validation and Analysis

To validate the efficiency and feasibility of the PM at resolving APD problems, three experiments were designed and implemented. Experiment 1 included four subtests to validate parameter combination types consisting of different decision variables. Experiment 2 included two subtests to confirm the ability of the PM to solve dynamic capacity problems. Experiment 3 ensured that the PM is suitable for large-scale problems and tested the GA parameter effects.

### 5.1. Experiment 1: Decision Variable Combination Validation

Four subtests consisting of different decision variables and important weights validated the ability of the PM to solve APD problems. The target area in Figure 5 is a  $90 \text{ km}^2$  irregularly shaped area. The capacity requirements in Figure 5 were identical for all subtests. All  $\text{req}_{ij}$  could move around the target area, where signal coverage was present. For comparative purposes, the GA parameters of the four subtests—population size (600), terminated generation (500), crossover rate (0.4), and mutation rate (0.1)—were fixed. Table 3 lists the other decision variables.

Table 4 shows the four subtests formulated as Model I and Model II according to the PM. To avoid the randomizing effect of the GA, all subtests were run three times with the same parameters on the same machine. The result averages are reported. Table 5 shows the analysis of the experiment results. The E1.a and E1.b results show that the important budget weight is less in E1.b than in E1.a. Therefore, only 15 APs (on average) are deployed for E1.b,

**Table 3:** Decision variables and import weights for four subtests in Experiment 1.

Decision variable	E1.a	E1.b	E1.c	E1.d
Budget	30000 ( $W_{\text{bgt}} = .25$ )	30000 ( $W_{\text{bgt}} = .40$ )	35000 ( $W_{\text{bgt}} = .25$ )	
Coverage rate	85% ( $W_{\text{cvg}} = .25$ )	85% ( $W_{\text{cvg}} = .20$ )	85% ( $W_{\text{cvg}} = .25$ )	
Capacity fulfillment rate	80% ( $W_{\text{cp}} = .25$ )	80% ( $W_{\text{cp}} = .20$ )	80% ( $W_{\text{cp}} = .25$ )	
Interference fulfillment rate	80% ( $W_{\text{IFT}} = .25$ )	80% ( $W_{\text{IFT}} = .20$ )	80% ( $W_{\text{IFT}} = .25$ )	90% ( $W_{\text{IFT}} = .25$ )

Note that important weights are marked in parentheses.

**Table 4:** Model I and Model II for resolving the four subtests.

Model I (for subtests E1.a and E1.b)	Model II (for subtests E1.c and E1.d)
$\text{Min } 0.25^a(0.4)^b \text{bgt}^+ + 0.25^a(0.2)^b \text{cvg}^- + 0.25^a(0.2)^b \text{cp}^- + 0.25^a(0.2)^b \text{IFT}^-$	$\text{Min } 0.25 \text{bgt}^+ + 0.25 \text{cvg}^- + 0.25 \text{cp}^- + 0.25 \text{IFT}^-$
Subject to	Subject to
$\text{CST}(T) - \text{bgt}^+ + \text{bgt}^- = 30000$	$\text{CST}(T) - \text{bgt}^+ + \text{bgt}^- = 30000$
$\Phi(T) - \text{cvg}^+ + \text{cvg}^- = 0.85$	$\Phi(T) - \text{cvg}^+ + \text{cvg}^- = 0.85$
$\Psi(T) - \text{cp}^+ + \text{cp}^- = 0.8$	$\Psi(T) - \text{cp}^+ + \text{cp}^- = 0.8$
$\omega(T) - \text{IFT}^+ + \text{IFT}^- = 0.8$	$\omega(T) - \text{IFT}^+ + \text{IFT}^- = 0.8^c(0.9)^d$
$\text{bgt}^+, \text{bgt}^-, \text{cvg}^+, \text{cvg}^-, \text{cp}^+, \text{cp}^-, \text{IFT}^+, \text{IFT}^- \geq 0$	$\text{bgt}^+, \text{bgt}^-, \text{cvg}^+, \text{cvg}^-, \text{cp}^+, \text{cp}^-, \text{IFT}^+, \text{IFT}^- \geq 0$

Note that <sup>a</sup>is for E1(a), <sup>b</sup>for E1(b), <sup>c</sup>for E1(c), and <sup>d</sup>for E1(d).

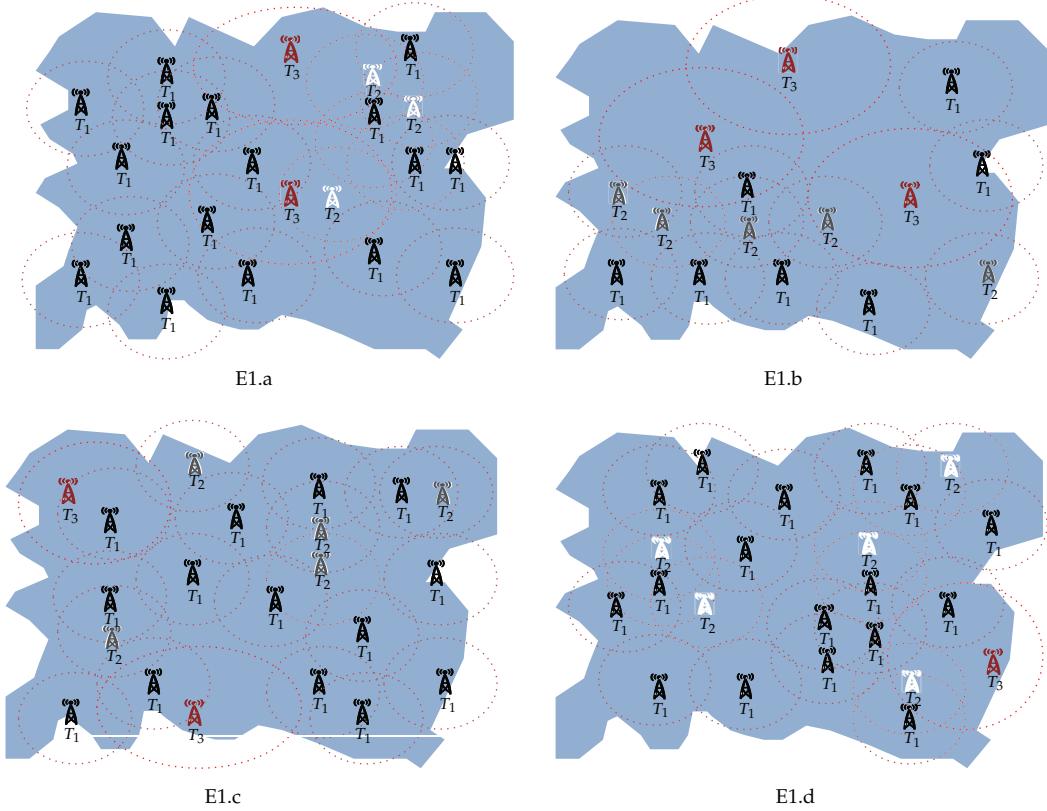
**Table 5:** Analysis of the results of Experiment 1.

Indicator	E1.a	E1.b	E1.c	E1.d
Fitness	.9534	.9420	.9568	.9567
Time (s)	160.0993	160.106	159.0593	170.4233
Cost	36954	29360	34647	34694
Coverage rate	.85	.6967	.8167	.8233
Capacity fulfillment rate	.8490	.6981	.8189	.8269
Interference fulfillment rate	.94	.9867	.94	.99
Number of APs	22	15	21	23

as shown in Figure 6. The results also show that the decision maker must either increase the budget or adjust the other decision objectives. For example, E1.b shows that the coverage and capacity fulfillment rates can only reach 0.7 at the current budget. As the budget increases from 30000 (in E1.b) to 35000 (in E1.c and E1.d), the number of APs deployed increases to 22 (on average). The coverage and capacity rates increase from 0.7 in E1.b to 0.82 in E1.c and E1.d. E1.d deployed more APs (23) at a lower cost than E1.a (22 APs). Figure 6 shows that the APs in E1.c and E1.d are spread evenly in the target area to avoid interference. Figure 7 shows the convergence trends for all subtests.  $T$  emerges after 100–150 iterations.

## 5.2. Experiment 2: Dynamic Capacity Requirement Validation

Experiment 2 consisted of two subtests to validate the ability of the PM to resolve dynamic capacity requirements. Figure 6(a) shows that in subtest E2.a, most capacity requirements are in the central area of the target ( $32 \text{ km}^2$ ). Figure 6(b) shows that in subtest E2.b, the capacity



**Figure 6:** AP deployment results for the four subtests in Experiment 1.

**Table 6:** Decision variable and important weights for Experiment 2.

Decision variable	Variable value
Budget	15000 ( $W_{\text{bgt}} = .3$ )
Coverage rate	85% ( $W_{\text{cvg}} = .15$ )
Interference fulfill rate	85% ( $W_{\text{IFT}} = .15$ )
Capacity fulfill rate	95% ( $W_{\text{cp}} = .4$ )

Note that the important weights are marked in parentheses.

requirements are scattered in the corners of the target area. Table 6 shows that the capacity requirements and all default decision variables are identical for both tests. The GA parameters—population size (600), terminated generation (500), crossover rate (0.4), and mutation rate (0.1)—were fixed to enable result comparison. To avoid random GA effects, all subtests were run three times with the same parameters on the same machine. The result averages are reported.

Table 7 shows the experiment results analysis. As expected, APD follows the capacity requirements, as shown in Figures 8(a) and 8(b). Figures 8(a) and 8(b) also show that APs are more central in E2.a than in E2.b to fulfill the capacity requirements. Although the capacity requirements are the same in both experiments, E2.a requires nine APs, which is more than

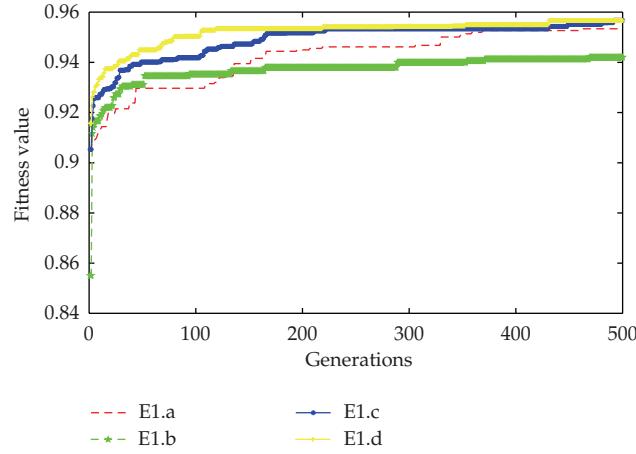


Figure 7: The convergence trends for the four subtests in Experiment 1.

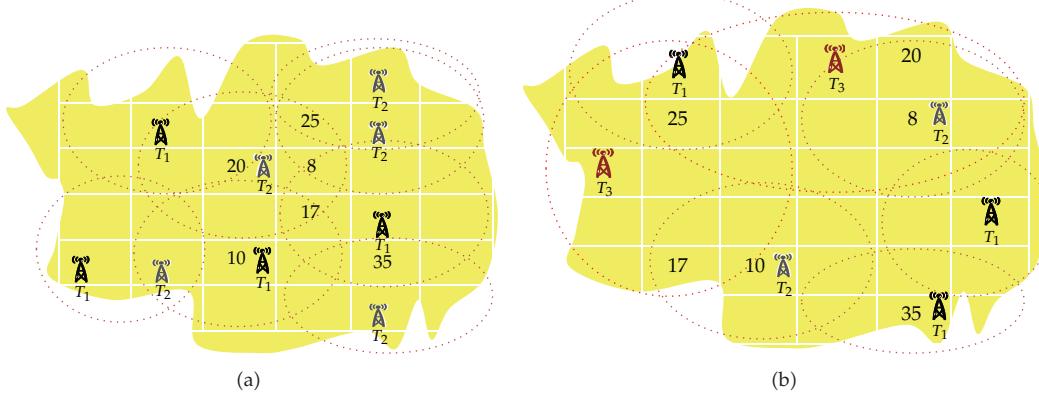


Figure 8: The capacity requirement and results of Experiment 2 for (a) subtest E2.a and (b) subtest E2.b. The numbers represent the capacity requirements.

E2.b (seven APs). Therefore, capacity requirements are dynamic, and  $T$  requires more APs to manage the capacity requirement increase in E2.a.

### 5.3. Experiment 3: The Effect of Large-Scale Problems and GA Parameters on Validation

Two subtests of Experiment 3 were designed as large-scale problems. Table 8 lists the decision variables and the important weights. Experiment 3 also tested GA parameter combinations, including crossover and mutation rates.

The results analysis in Table 9 shows that the PM is more sensitive to crossover rate. Generation—as an evaluation indicator for subtests E3.a and E3.b—shows that feasible deployment can be reached in the following order of crossover rates: 0.6 (converged by 22 iterations), 0.4 (converged by 70 iterations), and 0.2 (converged by 95 iterations). However, no such pattern exists for mutation rate in either test. Therefore, a crossover rate of 0.4 and

**Table 7:** Analysis of the results of Experiment 2.

Indicator	E2.a	E2.b
Fitness	1	1
Time (s)	5.1883	4.7797
Cost	14387	14068
Coverage rate	.8518	.8518
Capacity fulfillment rate	1	1
Interference fulfillment rate	.9815	.9259
Number of APs	9 APs	7 APs

**Table 8:** Decision variables and important weights for Experiment 3.

Objective	E3.a	E3.b
Target area	2250 km <sup>2</sup>	90000 km <sup>2</sup>
Budget ( $W_{\text{bgt}} = .55$ )	1350 000	5650000
Coverage rate ( $W_{\text{cvg}} = .15$ )	85%	85%
Capacity and interference fulfillment rates ( $W_{\text{cp}} = W_{\text{IFT}} = 0.15$ )	80%	80%

Note that the capacity requirements are shown in the Appendix and the important weights are marked in parentheses.

**Table 9:** Analysis of the results of Experiment 3.

Indicator	Fixed mutation rate ( $m = 0.1$ )			Fixed crossover rate ( $c = 0.4$ )	
	$C = 0.2$	$C = 0.4$	$C = 0.6$	$m = 0.01$	$M = 0.2$
Fitness	1 (.9869)	1 (1)	1 (1)	1 (1)	1 (1)
Generation	96 (93)	53 (86)	18 (26)	9 (94)	22 (62)
Time	489.2550 (3795.2)	526.2310 (5080.5)	552.5640 (3880.6)	496.0540 (3334.8)	541.5390 (4866.5)
Cost	1,349,670 (5,736,480)	1,326,549 (5,593,797)	1,347,933 (5,576,205)	1,348,896 (5,580,897)	1,324,479 (5,566,053)
Capacity fulfillment rate	.8485 (.8641)	.8384 (.8536)	.8495 (.8486)	.8450 (.8502)	.8531 (.8560)
Coverage fulfillment rate	.8500 (.8536)	.8496 (.8634)	.8484 (.8506)	.8496 (.8510)	.8496 (.8559)
Interference	.8996 (.8929)	.9100 (.8914)	.8996 (.8891)	.9048 (.8987)	.9012 (.8952)

The experiment results for E3(b) are in parentheses.

a mutation rate of 0.1 are recommended for experiment implementation to avoid GA parameter effects. The results in Table 9 show that large-scale problems (E3.b) can be resolved within an acceptable time (4191.52 s, approximately 1 h).

## 6. Conclusion

Optimal wireless LAN (WLAN) design is important to ensure seamless user communication. Appropriately locating wireless APs for WLANs is important. Optimal APD enables high telecommunication quality, balanced capacity loading, and optimal deployment costs. This study proposes a GP-driven model integrated with a GA to solve MO-APD subject to four constraints: budget, capacity, interference, and coverage. The experiment results show that

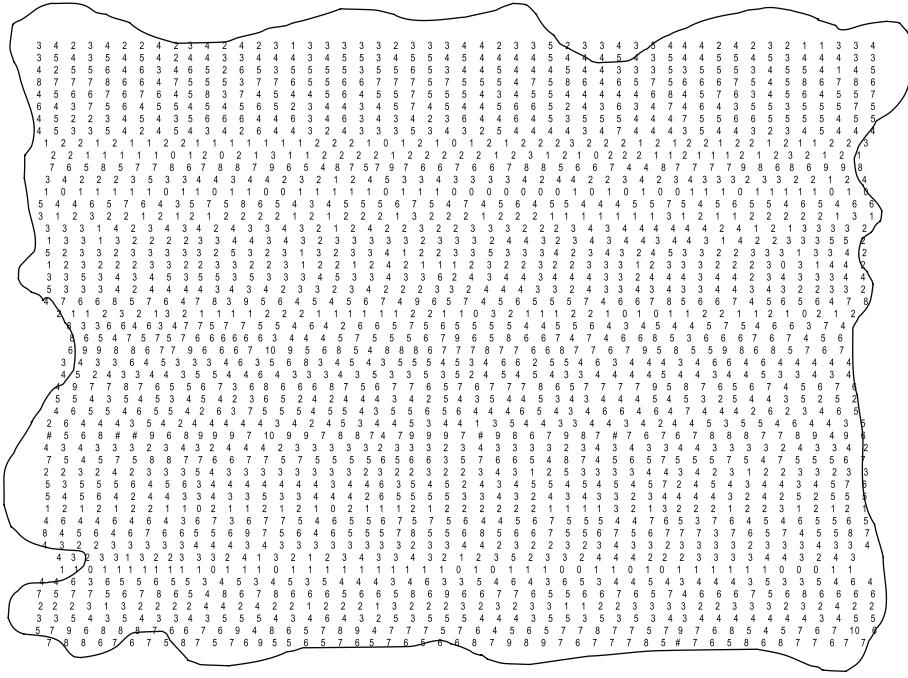


Figure 9: Capacity requirements for Experiment 3.a.

the PM resolves many APD problems and achieves dynamic capacity replication. Results confirm the ability of the PM to solve large-scale APD problems. Future research should focus on other applications and further verification of PM.

## Appendix

Figure 9 shows the capacity requirement for the E3.a subtest.

## References

- [1] N. Weicker, G. Szabo, K. Weicker, and P. Widmayer, "Evolutionary multiobjective optimization for base station transmitter placement with frequency assignment," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 189–203, 2003.
- [2] A. Mc Gibney, M. Klepal, and D. Pesch, "Agent-based optimization for large scale WLAN design," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 4, pp. 470–486, 2011.
- [3] L. Liao, W. Chen, C. Zhang, L. Zhang, D. Xuan, and W. Jia, "Two birds with one stone: wireless access point deployment for both coverage and localization," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 5, pp. 2239–2252, 2011.
- [4] K. P. Scheibe and C. T. Ragsdale, "A model for the capacitated, hop-constrained, per-packet wireless mesh network design problem," *European Journal of Operational Research*, vol. 197, no. 2, pp. 773–784, 2009.
- [5] K. Collins, S. Mangold, and G. M. Muntean, "Supporting mobile devices with wireless LAN/MAN in large controlled environments," *IEEE Communications Magazine*, vol. 48, no. 12, pp. 36–43, 2010.
- [6] A. Hills, "Large-scale wireless LAN design," *IEEE Communications Magazine*, vol. 39, no. 11, pp. 98–104, 2001.

- [7] K. Jaffr  s-Runser, J. M. Gorce, and S. Ub  da, "Mono- and multiobjective formulations for the indoor wireless LAN planning problem," *Computers and Operations Research*, vol. 35, no. 12, pp. 3885–3901, 2008.
- [8] R. Whitaker and S. Hurely, "Evolution of planning for wireless communication systems," in *Proceedings of the 36th Hawaii International Conference on System Sciences*, p. 10, January 2003.
- [9] J. H. Lee, B. J. Han, H. J. Lim, Y. D. Kim, N. Saxena, and T. M. Chung, "Optimizing access point allocation using genetic algorithmic approach for smart home environments," *Computer Journal*, vol. 52, no. 8, pp. 938–949, 2009.
- [10] I. E. Liao and K. F. Kao, "Enhancing the accuracy of WLAN-based location determination systems using predicted orientation information," *Information Sciences*, vol. 178, no. 4, pp. 1049–1068, 2008.
- [11] X. Huang, U. Behr, and W. Wiesbeck, "Automatic cell planning for a low-cost and spectrum efficient wireless network," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '00)*, vol. 1, pp. 282–276, December 2000.
- [12] Y. Zhao, H. Zhou, and M. Li, "Indoor access points location optimization using Differential Evolution," in *Proceedings of the International Conference on Computer Science and Software Engineering (CSSE '08)*, pp. 382–385, December 2008.
- [13] J. H. Lee, B. J. Han, H. K. Bang, and T. M. Chung, "An optimal access points allocation scheme based on genetic algorithm," in *Proceedings of the International Conference on Future Generation Communication and Networking (FGCN '07)*, pp. 55–59, December 2007.
- [14] F. Guo and T. C. Chiueh, "Scalable and robust WLAN connectivity using access point array," in *Proceedings of the International Conference on Dependable Systems and Networks*, pp. 288–297, July 2005.
- [15] X. Ling and K. L. Yeung, "Joint access point placement and channel assignment for 802.11 wireless LANs," *IEEE Transactions on Wireless Communications*, vol. 5, no. 10, pp. 2705–2711, 2006.
- [16] A. Charnes and W. W. Cooper, *Management Model and Industrial Application of Linear Programming*, vol. 1, Wiley, New York, NY, USA, 1961.
- [17] M. Tamiz, D. Jones, and C. Romero, "Goal programming for decision making: an overview of the current state-of-the-art," *European Journal of Operational Research*, vol. 111, no. 3, pp. 569–581, 1998.
- [18] C. Romero, "Extended lexicographic goal programming: a unifying approach," *Omega*, vol. 29, no. 1, pp. 63–71, 2001.
- [19] C. T. Chang, "Revised multi-choice goal programming," *Applied Mathematical Modelling*, vol. 32, no. 12, pp. 2587–2595, 2008.
- [20] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.

*Research Article*

## Multithreshold Segmentation Based on Artificial Immune Systems

**Erik Cuevas,<sup>1</sup> Valentin Osuna-Enciso,<sup>2</sup> Daniel Zaldivar,<sup>1</sup>  
Marco Pérez-Cisneros,<sup>1</sup> and Humberto Sossa<sup>2</sup>**

<sup>1</sup> Departamento de Ciencias Computacionales, CUCEI, Universidad de Guadalajara,  
Avenida Revolución 1500, 44430 Guadalajara, JAL, Mexico

<sup>2</sup> Centro de Investigación en Computación-IPN, Avenida Juan de Dios Bátiz s/n,  
Colonia Nueva Industrial Vallejo, 07738 Mexico, DF, Mexico

Correspondence should be addressed to Erik Cuevas, erik.cuevas@ceui.udg.mx

Received 19 February 2012; Accepted 18 April 2012

Academic Editor: Yi-Chung Hu

Copyright © 2012 Erik Cuevas 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.

Bio-inspired computing has lately demonstrated its usefulness with remarkable contributions to shape detection, optimization, and classification in pattern recognition. Similarly, multithreshold selection has become a critical step for image analysis and computer vision sparking considerable efforts to design an optimal multi-threshold estimator. This paper presents an algorithm for multi-threshold segmentation which is based on the artificial immune systems(AIS) technique, also known as theclonal selection algorithm (CSA). It follows the clonal selection principle (CSP) from the human immune system which basically generates a response according to the relationship between antigens (Ag), that is, patterns to be recognized and antibodies (Ab), that is, possible solutions. In our approach, the 1D histogram of one image is approximated through a Gaussian mixture model whose parameters are calculated through CSA. Each Gaussian function represents a pixel class and therefore a thresholding point. Unlike the expectation-maximization (EM) algorithm, the CSA-based method shows a fast convergence and a low sensitivity to initial conditions. Remarkably, it also improves complex time-consuming computations commonly required by gradient-based methods. Experimental evidence demonstrates a successful automatic multi-threshold selection based on CSA, comparing its performance to the aforementioned well-known algorithms.

### 1. Introduction

Several image-processing applications aim to detect and classify relevant features which may be later analyzed to perform several high-level tasks. In particular, image segmentation seeks to group pixels within meaningful regions. Commonly, gray levels belonging to the object are

substantially different from those featuring the background. Thresholding is thus a simple but effective tool to isolate objects of interest; its applications include several classics such as document image analysis, whose goal is to extract printed characters [1, 2], logos, graphical content, or musical scores; also it is used for map processing which aims to locate lines, legends, and characters [3]. Moreover, it is employed for scene processing, seeking for object detection, marking [4], and for quality inspection of materials [5, 6].

Thresholding selection techniques can be classified into two categories: bilevel and multilevel. In the former, one limit value is chosen to segment an image into two classes: one representing the object and the other one segmenting the background. When distinct objects are depicted within a given scene, multiple threshold values have to be selected for proper segmentation, which is commonly called multilevel thresholding.

A variety of thresholding approaches have been proposed for image segmentation, including conventional methods [7–10] and intelligent techniques [11, 12]. Extending the segmentation algorithms to a multilevel approach may cause some inconveniences: (i) they may have no systematic or analytic solution when the number of classes to be detected increases, and (ii) they may also show a slow convergence and/or high computational cost [11].

In this work, the segmentation algorithm is based on a parametric model holding a probability density function of gray levels which groups a mixture of several Gaussian density functions (Gaussian mixture). Mixtures represent a flexible method of statistical modelling as they are employed in a wide variety of contexts [13]. Gaussian mixture has received considerable attention in the development of segmentation algorithms despite its performance is influenced by the shape of the image histogram and the accuracy of the estimated model parameters [14]. The associated parameters can be calculated considering the expectation-maximization (EM) algorithm [15, 16] or gradient-based methods such as Levenberg-Marquardt, LM [17]. However, EM algorithms are very sensitive to the choice of the initial values [18]; meanwhile, gradient-based methods are computationally expensive and may easily get stuck within local minima [14]. Therefore, some researchers have attempted to develop methods based on modern global optimization algorithms such as the learning automata (LA) [19] and differential evolution algorithm [20]. In this paper, an alternative approach using a bio-inspired optimization algorithm for determining the parameters of a Gaussian mixture is presented.

On the other hand, biological inspired methods can successfully be transferred into novel computational paradigms as shown by the successful development of artificial neural networks, evolutionary algorithms, swarming algorithms, and so on. The human immune system (HIS) is a highly evolved, parallel, and distributed adaptive system [21] that exhibits remarkable abilities that can be imported into important aspects in the field of computation. This emerging field is known as artificial immune system (AIS) [22] which is a computational system fully inspired by the immunology theory and its functions, including principles and models. AISs have recently reached considerable research interest from different communities [23], focusing on several aspects of optimization, pattern recognition, abnormality detection, data analysis, and machine learning. Artificial immune optimization has been successfully applied to tackle numerous challenging optimization problems with remarkable performance in comparison to other classical techniques [24].

Clonal selection algorithm (CSA) [25] is one of the most widely employed AIS approaches. The CSA is a relatively novel evolutionary optimization algorithm which has been built on the basis of the clonal selection principle (CSP) [26] of HIS. The CSP explains the immune response when an antigenic pattern is recognized by a given antibody.

In the clonal selection algorithm, the antigen (Ag) represents the problem to be optimized and its constraints, while the antibodies (Abs) are the candidate solutions of the problem. The antibody-antigen affinity indicates as well the matching between the solution and the problem. The algorithm performs the selection of antibodies based on affinity either by matching against an antigen pattern or by evaluating the pattern via an objective function. In mathematical grounds, CSA has the ability of getting out of local minima while simultaneously operating over a pool of points within the search space. It does not use the derivatives or any of its related information as it employs probabilistic transition rules instead of deterministic ones. Despite its simple and straightforward implementation, it has been extensively employed in the literature for solving several kinds of challenging engineering problems [27–29].

In this paper, the segmentation process is considered as an optimization problem approximating the 1D histogram of a given image by means of a Gaussian mixture model. The operation parameters are calculated through the CSA. Each Gaussian contained within the histogram represents a pixel class and therefore belongs to the thresholding points. In order to compare the segmentation results with other optimization methods, the number of elements in the Gaussian mixture (classes) is considered already known or given by the user. The experimental results, presented in this work, demonstrate that CSA exhibits fast convergence, relative low computational cost, and no sensitivity to initial conditions by keeping an acceptable segmentation of the image, that is, a better mixture approximation in comparison to the EM- or gradient-based algorithms.

This paper organizes as follows. Section 2 presents the method following the Gaussian approximation of the histogram. Section 3 provides information about the CSA while Section 4 demonstrates the automatic threshold determination. Section 5 discusses some implementation details. Experimental results for the proposed approach are presented in Section 6, followed by the discussion summarized in Section 7.

## 2. Gaussian Approximation

Let consider an image holding  $L$  gray levels  $[0, \dots, L - 1]$  whose distribution is displayed within a histogram  $h(g)$ . In order to simplify the description, the histogram is normalized just as a probability distribution function, yielding

$$\begin{aligned} h(g) &= \frac{n_g}{N}, \quad h(g) > 0, \\ N &= \sum_{g=0}^{L-1} n_g, \quad \sum_{g=0}^{L-1} h(g) = 1, \end{aligned} \tag{2.1}$$

where  $n_g$  denotes the number of pixels with gray level  $g$  and  $N$  being the total number of pixels in the image.

The histogram function can thus be contained into a mix of Gaussian probability functions of the form

$$p(x) = \sum_{i=1}^K P_i \cdot p_i(x) = \sum_{i=1}^K \frac{P_i}{\sqrt{2\pi\sigma_i^2}} \exp\left[-\frac{(x - \mu_i)^2}{2\sigma_i^2}\right], \tag{2.2}$$

with  $P_i$  being the probability of class  $i$ ,  $p_i(x)$  being the probability distribution function of gray-level random variable  $x$  in class  $i$ ,  $\mu_i$  and  $\sigma_i$  being the mean and standard deviation of

the  $i$ th probability distribution function, and  $K$  being the number of classes within the image. In addition, the constraint  $\sum_{i=1}^K P_i = 1$  must be satisfied.

The mean square error is used to estimate the  $3K$  parameters  $P_i$ ,  $\mu_i$ , and  $\sigma_i$ ;  $i = 1, \dots, K$ . For instance, the mean square error between the Gaussian mixture  $p(x_i)$  and the experimental histogram function  $h(x_i)$  is now defined as follows:

$$J = \frac{1}{n} \sum_{j=1}^n [p(x_j) - h(x_j)]^2 + \omega \cdot \left| \left( \sum_{i=1}^K P_i \right) - 1 \right|. \quad (2.3)$$

Assuming an  $n$ -point histogram as in [13] and  $\omega$  being the penalty associated with the constrain  $\sum_{i=1}^K P_i = 1$ . In general, the parameter estimation that minimizes the square error produced by the Gaussian mixture is not a simple problem. A straightforward method is to consider the partial derivatives of the error function to zero by obtaining a set of simultaneous transcendental equations [13]. However, an analytical solution is not always available considering the nonlinear nature of the equation which in turn yields the use of iterative approaches such as gradient-based or maximum likelihood estimation. Unfortunately, such methods may also get easily stuck within local minima.

In the case of other algorithms such as the EM algorithm and the gradient-based methods, the new parameter point lies within a neighbourhood distance of the previous parameter point. However, this is not the case for the CSA which is based on stochastic principles. New operating points are thus determined by a parameter probability function that may yield points lying far away from previous operating points, providing the algorithm with a higher ability to locate and pursue a global minimum.

This paper aims to compare its segmentation results to other optimization methods that have been applied to similar segmentation tasks. Therefore, the number of elements in the Gaussian mixture (classes) is considered as already known or provided by the user. The number of classes, in most cases, represents the number of objects which are contained in the image. However, if the selected number is lower than the object number, it can be assumed that results actually favour the classification of bigger objects yet bearing the expense of ignoring small subjects.

### 3. Clonal Selection Algorithm

In natural immune systems, only the antibodies (Abs) which are able to recognize the intrusive antigens (nonself cells) are to be selected to proliferate by cloning [21]. Therefore, the fundament of the clonal optimization method is that only capable Abs will proliferate. Particularly, the underlying principles of the CSA are borrowed from the CSP as follows:

- (i) maintenance of memory cells which are functionally disconnected from repertoire,
- (ii) selection and cloning of most stimulated Abs,
- (iii) suppression of nonstimulated cells,
- (iv) affinity maturation and reselection of clones showing the highest affinities,
- (v) mutation rate proportional to Abs affinities.

From immunology concepts, an antigen is any substance that forces the immune system to produce antibodies against it. Regarding the CSA systems, the antigen concept

refers to the pending optimization problem which focuses on circle detection. In CSA, B cells, T cells, and antigen-specific lymphocytes are generally called antibodies. An antibody is a representation of a candidate solution for an antigen, for example, the prototype circle in this work. A selective mechanism guarantees that those antibodies (solutions) that better recognize the antigen and therefore may elicit the response are to be selected holding long life spans. Therefore, such cells are to be named memory cells ( $\mathbf{M}$ ).

### 3.1. Definitions

In order to describe the CSA, the notation includes boldfaced capital letters indicating matrices and boldfaced small letters indicating vectors. Some relevant concepts are also revisited below:

- (i) antigen: the problem to be optimized and its constraints (circle detection),
- (ii) antibody: the candidate solutions of the problem (circle candidates),
- (iii) affinity: the objective function measurement for an antibody (circle matching),

The limited-length character string  $\mathbf{d}$  is the coding of variable vector  $\mathbf{x}$  as  $\mathbf{d} = \text{encode}(\mathbf{x})$ ; and  $\mathbf{x}$  is called the decoding of antibody  $\mathbf{d}$  following  $\mathbf{x} = \text{decode}(\mathbf{d})$ .

Set  $\mathbf{I}$  is called the antibody space; namely,  $\mathbf{d} \in \mathbf{I}$ . The antibody population space is thus defined as

$$\mathbf{I}^m = \{\mathbf{D} : \mathbf{D} = (\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_m), \mathbf{d}_k \in \mathbf{I}, 1 \leq k \leq m\}, \quad (3.1)$$

where the positive integer  $m$  is the size of antibody population  $\mathbf{D} = \{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_m\}$  which is an  $m$ -dimensional group of antibody  $\mathbf{d}$ , being a spot within the antibody space  $\mathbf{I}$ .

### 3.2. CSA Operators

Based on [30], the CSA implements three different operators: the clonal proliferation operator ( $T_P^C$ ), the affinity maturation operator ( $T_M^A$ ), and the clonal selection operator ( $T_S^C$ ).  $\mathbf{A}(k)$  is the antibody population at time  $k$  that represents the set of antibodies  $\mathbf{a}$ , such as  $\mathbf{A}(k) = \{\mathbf{a}_1(k), \mathbf{a}_2(k), \dots, \mathbf{a}_n(k)\}$ . The evolution process of CSA can be described as follows:

$$\mathbf{A}(k) \xrightarrow{T_P^C} \mathbf{Y}(k) \xrightarrow{T_M^A} \mathbf{Z}(k) \cup \mathbf{A}(k) \xrightarrow{T_S^C} \mathbf{A}(k+1). \quad (3.2)$$

#### 3.2.1. Clonal Proliferation Operator ( $T_P^C$ )

Define

$$\mathbf{Y}(k) = T_P^C(\mathbf{A}(k)) = \left[ T_P^C(\mathbf{a}_1(k)), T_P^C(\mathbf{a}_2(k)), \dots, T_P^C(\mathbf{a}_n(k)) \right], \quad (3.3)$$

where  $(k) = T_P^C(\mathbf{A}(k)) = \mathbf{e}_i \cdot \mathbf{a}_i(k) i = 1, 2, \dots, n$ , and  $\mathbf{e}_i$  is a  $q_i$ -dimensional identity column

vector. Function  $\text{round}(x)$  gets  $x$  to the least integer bigger than  $x$ . There are various methods for calculating  $q_i$ . In this work, it is calculated as follows:

$$q_i(k) = \text{round} \left[ N_c \cdot \frac{F(\mathbf{a}_i(k))}{\sum_{j=1}^n F(\mathbf{a}_j(k))} \right] \quad i = 1, 2, \dots, n, \quad (3.4)$$

where  $N_c$  is called the clonal size. The value of  $q_i(k)$  is proportional to the value of  $F(\mathbf{a}_i(k))$ . After clonal proliferation, the population becomes

$$\mathbf{Y}(k) = \{\mathbf{Y}_1(k), \mathbf{Y}_2(k), \dots, \mathbf{Y}_n(k)\}, \quad (3.5)$$

where

$$\begin{aligned} \mathbf{Y}_i(k) &= \{\mathbf{y}_{ij}(k)\} = \{\mathbf{y}_{i1}(k), \mathbf{y}_{i2}(k), \dots, \mathbf{y}_{iq_i}(k)\}, \\ \mathbf{y}_{ij}(k) &= \mathbf{a}_1(k), \quad j = 1, 2, \dots, q_i, \quad i = 1, 2, \dots, n. \end{aligned} \quad (3.6)$$

### 3.2.2. Affinity Maturation Operator ( $T_M^A$ )

The affinity maturation operation is performed by hypermutation. Random changes are introduced into the antibodies just like it happens in the immune system. Such changes may lead to increase in the affinity. The hypermutation is performed by the operator  $T_M^A$  which is applied to the population  $\mathbf{Y}(k)$  as it is obtained by clonal proliferation  $\mathbf{Z}(k) = T_M^C(\mathbf{Y}(k))$ .

The mutation rate is calculated using the following equation [25]:

$$\alpha = e^{(-\rho \cdot F(ab))}, \quad (3.7)$$

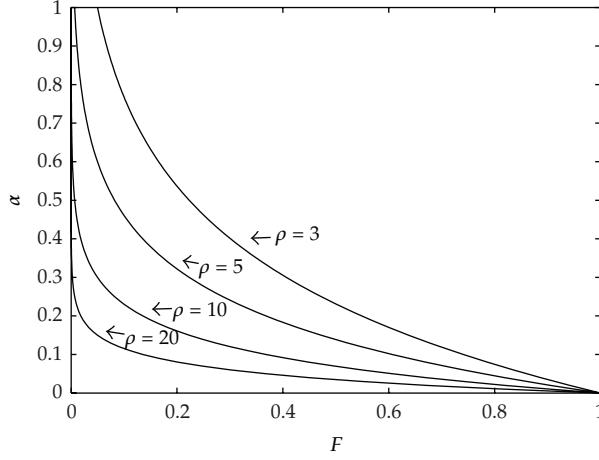
$\alpha$  being the mutation rate,  $F$  being the objective function value of the antibody ( $ab$ ) as it is normalized between  $[0,1]$ , and  $\rho$  being a fixed step. In [31], it is demonstrated the importance of including the factor  $\rho$  into (3.7) to improve the algorithm performance. The way  $\rho$  modifies the shape of the mutation rate is shown by Figure 1.

The number of mutations held by a clone with objective function value  $F$  is equal to  $L \cdot \alpha$ , considering  $L$  as the length of the antibody—22 bits are used in this paper. For the binary encoding, mutation operation can be done as follows: each gene within an antibody may be replaced by its opposite number (i.e., 0-1 or 1-0).

Following the affinity maturation operation, the population becomes

$$\begin{aligned} \mathbf{Z}(k) &= \{\mathbf{Z}_1(k), \mathbf{Z}_2(k), \dots, \mathbf{Z}_n(k)\}, \\ \mathbf{Z}_i(k) &= \{\mathbf{z}_{ij}(k)\} = \{\mathbf{z}_{i1}(k), \mathbf{z}_{i2}(k), \dots, \mathbf{z}_{iq_i}(k)\}, \\ \mathbf{z}_{ij}(k) &= T_M^A(\mathbf{y}_{ij}(k)), \quad j = 1, 2, \dots, q_i, \quad i = 1, 2, \dots, n, \end{aligned} \quad (3.8)$$

where  $T_M^A$  is the operator as it is defined by (3.7) and applied onto the antibody  $\mathbf{y}_{ij}$ .



**Figure 1:** Hypermutation rate versus fitness, considering some size steps.

### 3.2.3. Clonal Selection Operator ( $T_S^C$ )

Define, for all  $i = 1, 2, \dots, n$ ,  $\mathbf{b}_i(k) \in \mathbf{Z}_i(k)$  as the antibody with the highest affinity in  $\mathbf{Z}_i(k)$ , then  $\mathbf{a}_i(k+1) = T_S^C(\mathbf{Z}_i(k) \cup \mathbf{a}_i(k))$ , where  $T_S^C$  is defined as

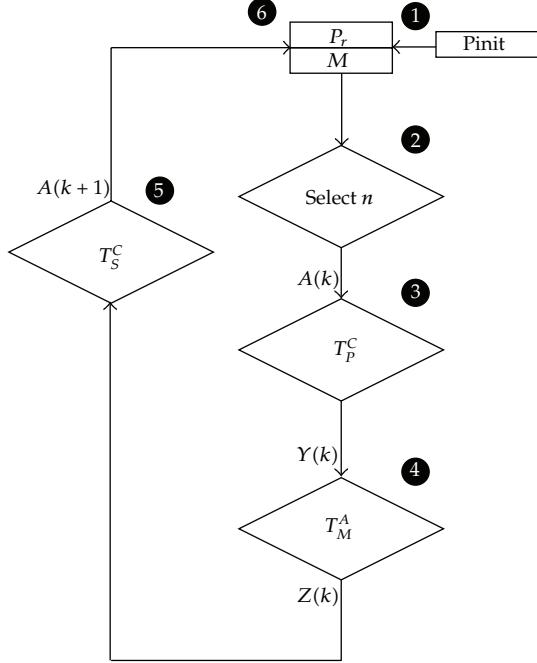
$$T_S^C(\mathbf{Z}_i(k) \cup \mathbf{a}_i(k)) = \begin{cases} \mathbf{b}_i(k) & \text{if } F(\mathbf{a}_i(k)) < F(\mathbf{b}_i(k)) \\ \mathbf{a}_i(k) & \text{if } F(\mathbf{a}_i(k)) \geq F(\mathbf{b}_i(k)), \end{cases} \quad (3.9)$$

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

Each step of the CSA may be defined as follows.

- (1) Initialize randomly a population ( $P_{init}$ ), a set  $h = P_r + n$  of candidate solutions of subsets of memory cells ( $\mathbf{M}$ ) which is added to the remaining population ( $P_r$ ), with the total population being  $\mathbf{P}_T = P_r + M$ , with  $\mathbf{M}$  holding  $n$  memory cells.
- (2) Select the  $n$  best individuals of the population  $\mathbf{P}_T$  to build  $\mathbf{A}(k)$ , according to the affinity measure (objective function).
- (3) Reproduce ( $T_P^C$ ) population  $\mathbf{A}(k)$  proportionally to their affinity with the antigen and generate a temporary population of clones  $\mathbf{Y}(k)$ . The clone number is an increasing function of the affinity with the antigen (3.1).
- (4) Mutate ( $T_M^A$ ) the population  $\mathbf{Y}(k)$  of clones according to the affinity of the antibody to the antigen (3.4). A matured antibody population  $\mathbf{Z}(k)$  is thus generated.
- (5) Reselect ( $T_S^C$ ) the best individuals from  $\mathbf{Z}(k)$  and  $\mathbf{A}(k)$  to compose a new memory set  $\mathbf{M} = \mathbf{A}(k+1)$ .
- (6) Add random  $P_r$  novel antibodies (diversity introduction) to the new memory cells  $\mathbf{M}$  to build  $\mathbf{P}_T$ .
- (7) Stop if any criteria are reached; otherwise return to (2.2).

Figure 2 shows the full draw of the CSA. The clone number in Step 3 is defined according to (3.1). Although a unique mutation operator is used in Step 5, the mutated values



**Figure 2:** Basic flow diagram of clonal selection algorithm (CSA).

of individuals are inversely proportional to their fitness by means of (3.7); that is, the more Ab shows a better fitness, the less it may change.

The similarity property [32] within the Abs can also affect the convergence speed of the CSA. The idea of the antibody addition based on the immune network theory is introduced for providing diversity to the newly generated Abs in  $M$ , which may be similar to those already in the old memory  $M$ . Holding such a diverse Ab pool, the CSA can avoid being trapped into local minima [30], contrasting to well-known genetic algorithms (GAs) which usually tend to bias the whole population of chromosomes towards only the best candidate solution. Therefore, it can effectively handle challenging multimodal optimization tasks [33–36].

The management of population includes a simple and direct searching algorithm for globally optimal multimodal functions. This is also another clear difference in comparison to other evolutionary algorithms, like GA, because it does not require crossover but only cloning and hypermutation of individuals in order to use affinity as selection mechanism. The CSA is adopted in this work to find the parameters  $P$ ,  $\sigma$ , and  $\mu$  of Gaussian functions and their corresponding threshold values for the image.

#### 4. Determination of Thresholding Values

The next step is to determine the optimal threshold values. Considering that the data classes are organized such that  $\mu_1 < \mu_2 < \dots < \mu_K$ , the threshold values are obtained by computing the overall probability error of two adjacent Gaussian functions, yielding

$$E(T_h) = P_{h+1} \cdot E_1(T_h) + P_i \cdot E_2(T_h), \quad h = 1, 2, \dots, K - 1, \quad (4.1)$$

considering

$$\begin{aligned} E_1(T_h) &= \int_{-\infty}^{T_h} p_{h+1}(x)dx, \\ E_2(T_h) &= \int_{T_h}^{\infty} p_h(x)dx. \end{aligned} \quad (4.2)$$

$E_1(T_h)$  is the probability of mistakenly classifying the pixels in the  $(h + 1)$ th class belonging to the  $h$ th class, while  $E_2(T_h)$  is the probability of erroneously classifying the pixels in the  $h$ th class belonging to the  $(h + 1)$ th class.  $P_j$ 's are the a priori probabilities within the combined probability density function, and  $T_h$  is the threshold value between the  $h$ th and the  $(h + 1)$ th classes. One  $T_h$  value is chosen such as the error  $E(T_h)$  is minimized. By differentiating  $E(T_h)$  with respect to  $T_h$  and equating the result to zero, it is possible to use the following equation to define the optimum threshold value  $T_h$ :

$$AT_h^2 + BT_h + C = 0, \quad (4.3)$$

considering

$$\begin{aligned} A &= \sigma_h^2 - \sigma_{h+1}^2, \\ B &= 2 \cdot (\mu_h \sigma_{h+1}^2 - \mu_{h+1} \sigma_h^2), \\ C &= (\sigma_h \mu_{h+1})^2 - (\sigma_{h+1} \mu_h)^2 + 2 \cdot (\sigma_h \sigma_{h+1})^2 \cdot \ln\left(\frac{\sigma_{h+1} P_h}{\sigma_h P_{h+1}}\right). \end{aligned} \quad (4.4)$$

Although the above quadratic equation has two possible solutions, only one of them is feasible; that is, a positive value falling within the interval.

## 5. Implementation Details

In this work, either an antibody or an antigen will be represented (in binary form) by a bit chain of the form

$$c = \langle c_1, c_2, \dots, c_L \rangle, \quad (5.1)$$

with  $c$  representing a point in an  $L$ -dimensional space:

$$c \in S^L. \quad (5.2)$$

The clonal selection algorithm can be stated as follows.

- (1) An original population of  $N$  individuals (antibodies) is generated, considering the size of 22 bits.
- (2) The  $n$  best individuals based on the affinity measure are selected. They will represent the memory set.

- (3) Such  $n$  best individuals are cloned  $m$  times.
- (4) Performing a hypermutation of the cloned individuals which follows the proportion inside the affinity between antibodies and antigens and generating one improved antibody population.
- (5) From the hypermutated population, the individuals with the higher affinity are to be reselected.
- (6) As for the original population, the individuals with the highest affinity are replaced, improving the overall cells set.

Once the above steps are completed, the process is started again, until one individual showing the highest affinity is found, that is, finding the minimum stated in (2.3). At this work, the algorithm considers 3 Gaussians to represent the same number of classes. However, it can be easily expanded to more classes. Each single Gaussian has the variables  $P_i, \mu_i, \sigma_i$  (with  $i = 1, 2, 3$ ) representing the Hamming shape-space by means of an 22 bits word over the following ranges:

$$\begin{aligned} P_i &: [1, \max(h)] \\ \mu_i &: [\min(g), \max(g)] \\ \sigma_i &: [1, \max(g)^*0.5], \end{aligned} \quad (5.3)$$

with  $g$  representing the grey level and  $h$  is the histogram of the grey level image. Hence, the first step is to generate the initial individual population of the antibody population by means of

$$AB = 2 \cdot r, (N, S_p) - 1; \quad (5.4)$$

with  $S_p$  representing the bit string assigned to each of the initial individuals  $N$ . Later, in order to perform the mapping from binary string to real value, we use

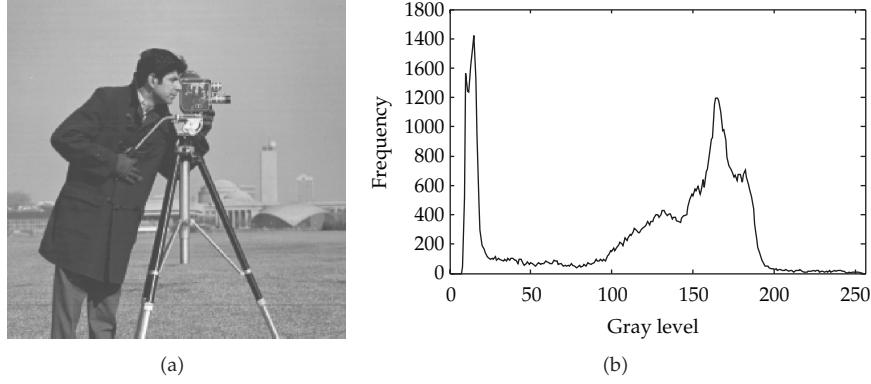
$$(\langle c_L, \dots, c_2, c_1 \rangle)_2 = \left( \sum_{i=0}^{21} c_i \cdot 2^i \right)_{10} = r'. \quad (5.5)$$

As to find the corresponding real value for  $r$ ,

$$r = r' \cdot \frac{r_{\max}}{2^{22} - 1}, \quad (5.6)$$

with  $r_{\max}$  representing  $\max(h), \max(g), \max(g)/2$ .

The population is set to 100 individuals ( $N$ ), including the best 20 individuals ( $n$ ). The 20 selected individuals are cloned 10 times ( $m$ ). The corresponding mutation probability is proportional to the resulting error by (2.3). The algorithm is thus executed until the minimum possible value of (2.3) is reached.



**Figure 3:** (a) Original “The Cameraman” image and (b) its correspondent histogram.

## 6. Experimental Results

### 6.1. Presentation of Results

In this section, two experiments are tested to evaluate the performance of the proposed algorithm. The first one considers the well-known image of the “The Cameraman” shown in Figure 3(a) as its corresponding histogram is presented by Figure 3(b). The goal is to segment the image with 3 different pixel classes. During learning, the CSA algorithm adjusts 9 parameters in this test. In this experiment, a population of 100 ( $N$ ) individuals is considered. Each candidate holds 9 dimensions, yielding

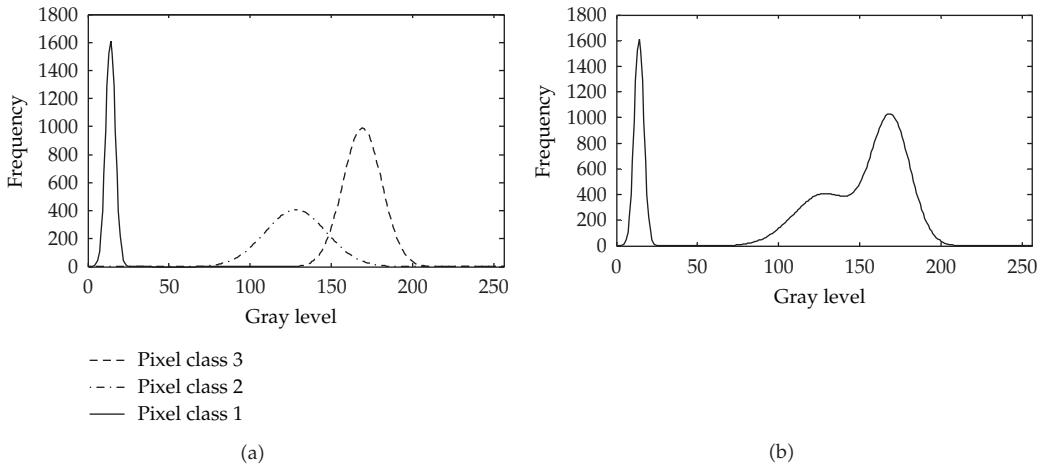
$$I_N = \left\{ \sigma_1^N, \sigma_2^N, \sigma_3^N, P_1^N, P_2^N, P_3^N, \mu_1^N, \mu_2^N, \mu_3^N \right\}, \quad (6.1)$$

with  $N$  representing the individual’s number, in this case, 100. The parameters  $(P, \sigma, \mu)$  are randomly initialized.

The experiments suggest that, after 130 iterations, the CSA algorithm has converged to the global minimum. Figure 4(a) shows the obtained Gaussian functions (pixel classes), while Figure 4(b) shows the combined graph. The layout in Figure 4(b) suggests an easy combination of the Gaussian functions to approximate the shape of the graph shown in Figure 3(b), representing the histogram of the original image. Figure 5 shows the segmented image whose threshold values are calculated according to (3.4) and (3.5).

In order to test the performance, the algorithm gets to analyze the image shown in Figure 6 whose histogram exhibits a remarkable problem (a set of peaks) regarding class approximation. Such image, due to its complexity, is considered as a benchmark image for other algorithms, including classical approaches as in [7, 10] or intelligent algorithms as in [11, 12]. For this particular image, after 190 generations, the algorithm was capable to achieve the minimum approximation value  $J$  (see (2.3)), considering three different classes. Figure 7 shows the approximation quality; meanwhile, Figure 8 presents the obtained segmented image.

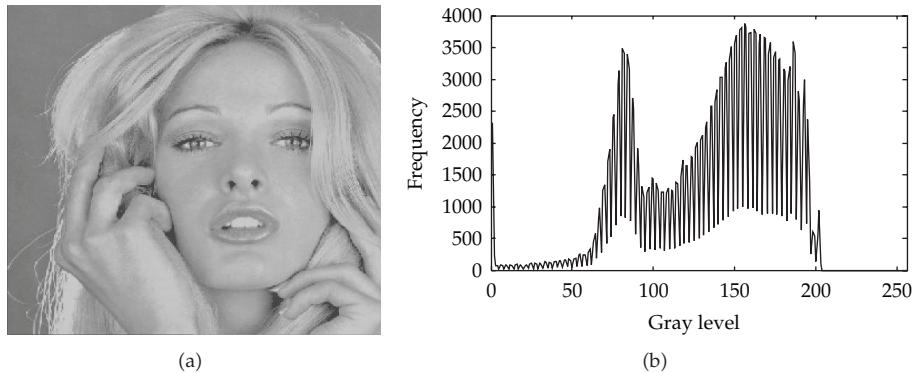
The third experiment considers a new image known as “The scene” shown by Figure 9(a). The histogram is presented in Figure 9(b). Now, the algorithm considers 4 pixel classes. The optimization is performed by the CSA algorithm resulting in the Gaussian



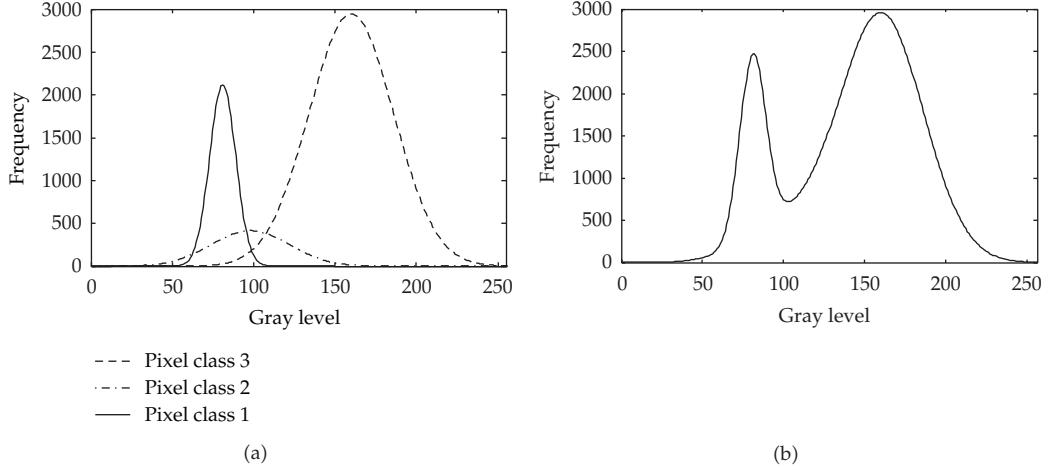
**Figure 4:** Applying the CSA algorithm for 3 classes and its results: (a) Gaussian functions for each class, (b) mixed Gaussian functions (approaching the original histogram).



**Figure 5:** The image after the segmentation is applied, considering three classes.



**Figure 6:** Benchmark image and its histogram.



**Figure 7:** CSA segmentation for 3 classes over the benchmark image: (a) Gaussian functions for each class. (b) Mixed Gaussian functions approaching the look of the original histogram.



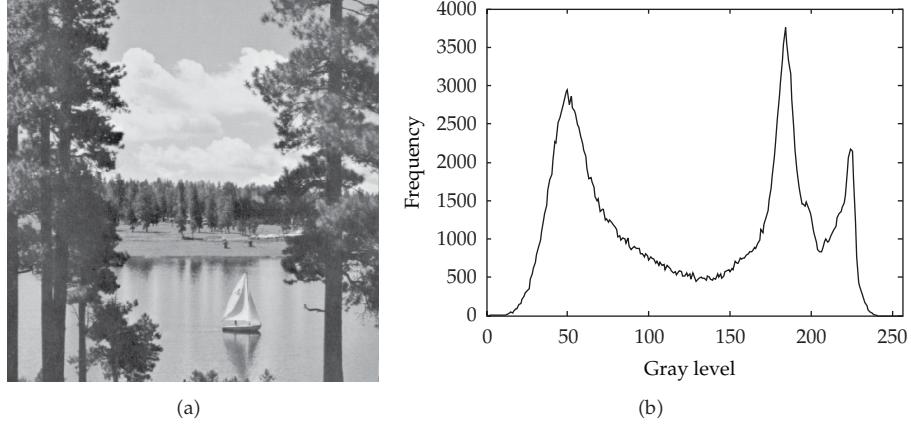
**Figure 8:** The benchmark image after segmentation considering all three classes.

functions shown by Figure 10(a). Figure 10(b) presents the combined graph from the addition of such Gaussian functions.

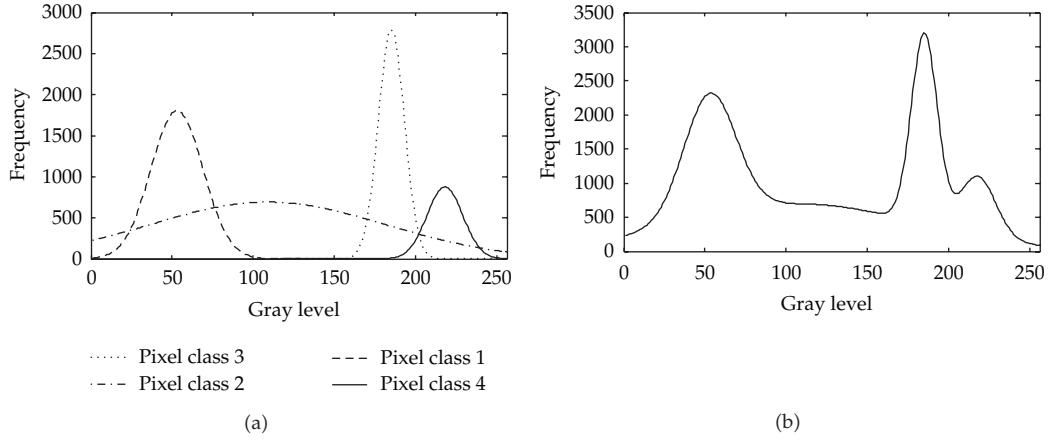
After the optimization by the CSA algorithm, the added layout including all 4 Gaussian functions is obtained as shown by Figure 11(a). It is also evident that the resulting function approaches the original histogram as Figure 11(b) shows the resulting image after applying the segmentation algorithm.

## 6.2. Comparing the CSA versus the EM and LM Methods

In order to enhance the algorithm's analysis, the proposed approach is compared to the EM algorithm and the Levenberg-Marquardt (LM) method which are commonly employed for



**Figure 9:** Third experiment data: (a) the original image “The scene” and (b) its corresponding histogram.

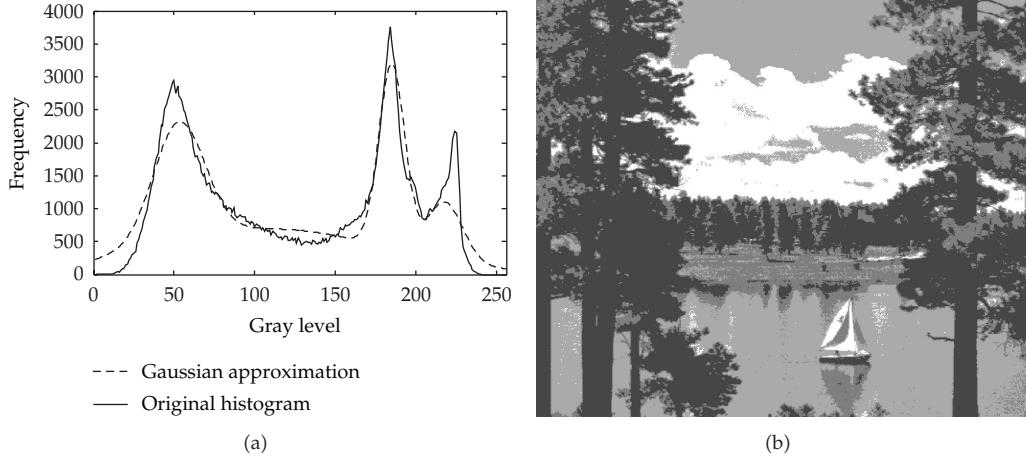


**Figure 10:** Applying the CSA algorithm for 4 classes: (a) Gaussian functions at each class: (b) adding all four Gaussian functions, it approaches the original histogram.

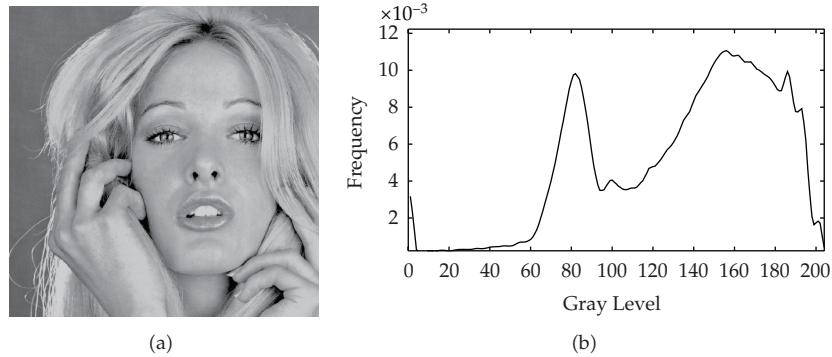
computing Gaussian mixture parameters. The comparison focuses on the following issues: sensitivity to initial conditions, singularities, convergence, and computational costs.

#### 6.2.1. Sensitivity to the Initial Conditions

In this experiment, initial values of the mixture model for all methods are randomly set while the same histogram is taken in account for the approximation task. Final parameters representing the Gaussian mixture (considering four different classes) after convergence are reported. All algorithms (EM, LM, and CSA) are executed until no change in parameter values is detected. Figure 12(a) shows the image used in this comparison while Figure 12(b) pictures its histogram. All experiments are conducted several times in order to assure consistency. Table 1 exhibits the parameter values ( $\mu_q, \sigma_q, P_q, q \in 1, \dots, 4$ ) of the obtained Gaussian mixtures, considering the two initial conditions in which the highest contrasts were found. Figure 13 shows the segmented images obtained under such initial conditions. Further



**Figure 11:** Segmentation of “The Scene” considering four classes for the CSA algorithm. (a) Comparison between the original histogram and the Gaussian approach. (b) The image after the segmentation process.

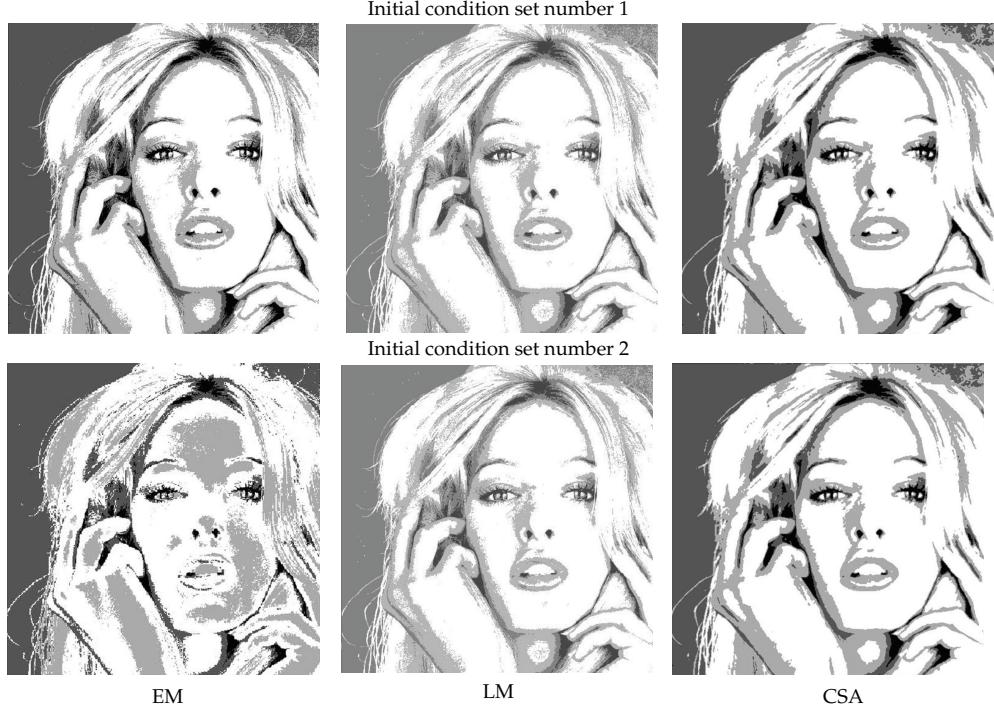


**Figure 12:** (a) Original image used for the comparison on initial conditions and (b) its corresponding histogram.

analysis on Table 1 shows the acute sensitivity of the EM algorithm to initial conditions. By such sensitivity, it is observed in Figure 13 where a clear pixel misclassification appears in some sections of the image. In case of the LM method, although it does not present a considerable difference in comparison to optimal values, its deviation shows that it is prone to get trapped into a local minimum. On the other hand, the CSA algorithm exhibits the best performance as its parameter values fall the nearest to the optimal approximation performance.

#### 6.2.2. Convergence and Computational Cost

The experiment aims to measure the number of iterations and the computing time spent by the EM, the LM, and the CSA required to calculate the parameters of the Gaussian mixture after considering three different benchmark images. Figure 14 shows the images used in the experiment. Such images are selected, since they are employed in the standard

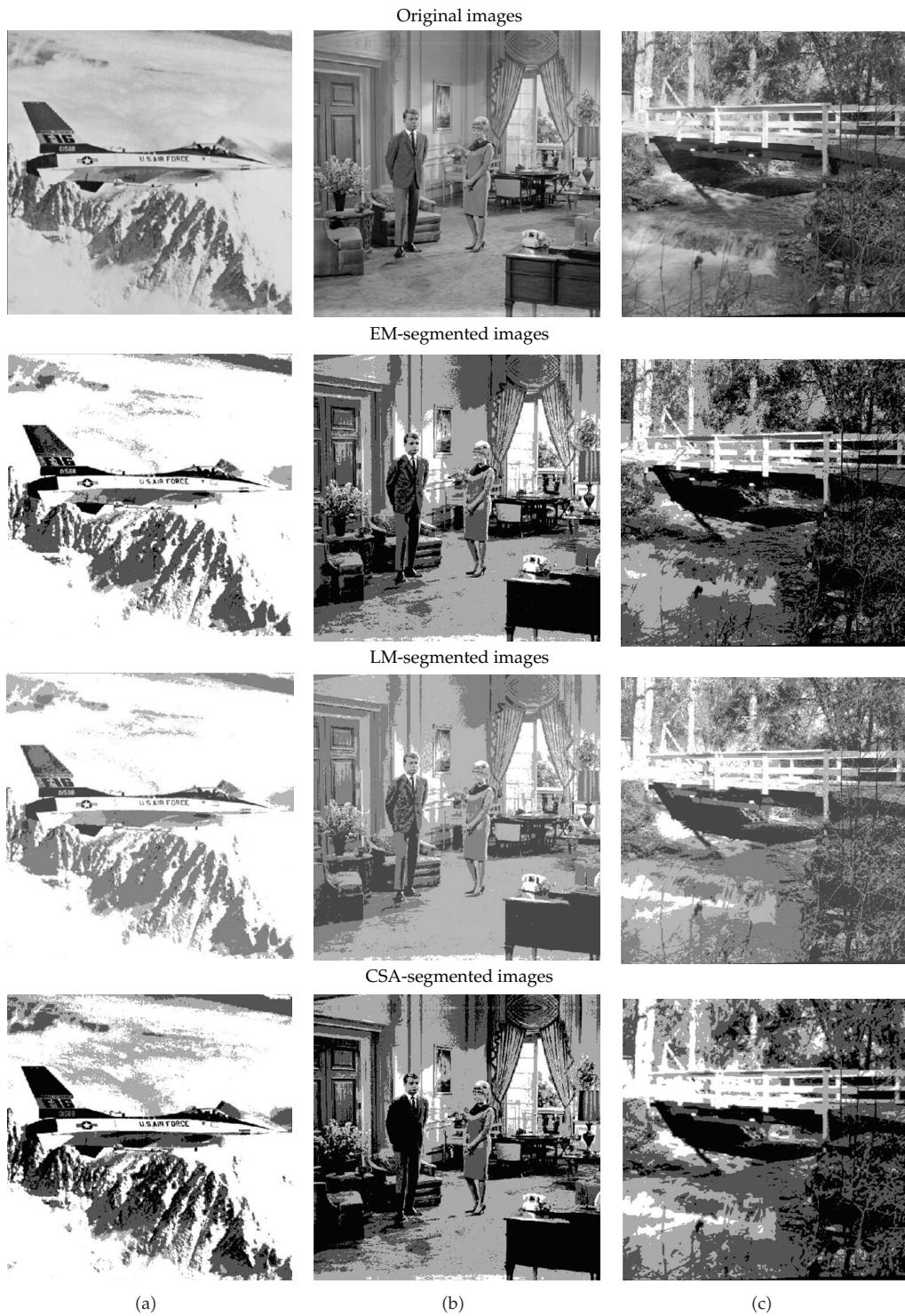


**Figure 13:** Segmented images after applying the EM, the LM, and the CSA algorithm with different initial conditions.

**Table 1:** Comparison between the EM, the LM, and the CSA algorithm, considering two different initial conditions.

Parameters	Initial condition 1	EM	LM	CSA	Initial condition 2	EM	LM	CSA
$\mu_1$	40.6	33.13	32.12	32.23	10	20.90	31.80	32.25
$\mu_2$	81.2	81.02	82.05	81.55	100	82.78	80.85	82.00
$\mu_3$	121.8	127.52	127	126.89	138	146.67	128	127.11
$\mu_4$	162.4	167.58	166.80	167.00	200	180.72	165.90	166.50
$\sigma_1$	15	25.90	25.50	25.30	10	18.52	20.10	25.01
$\sigma_2$	15	9.78	9.70	9.86	5	12.52	9.81	9.45
$\sigma_3$	15	17.72	17.05	17.12	8	20.5	15.15	17.23
$\sigma_4$	15	17.03	17.52	17.45	22	10.09	18.00	17.22
$P_1$	0.25	0.0313	0.0310	0.317	0.20	0.0225	0.0312	0.317
$P_2$	0.25	0.2078	0.2081	0.198	0.30	0.2446	0.2079	0.214
$P_3$	0.25	0.2508	0.2500	0.249	0.20	0.5232	0.2502	0.245
$P_4$	0.25	0.5102	0.5110	0.501	0.30	0.2098	0.5108	0.498

segmentation literature. All the experiments consider four classes. Table 2 shows the averaged measurements as they are obtained from 20 experiments. It is evident that the EM is the slowest to converge (iterations), and the LM shows the highest computational cost (time elapsed) because it requires complex Hessian approximations. On the other hand, the CSA shows an acceptable tradeoff between its convergence time and its computational



**Figure 14:** Original benchmark images ((a)–(c)) and segmented images obtained by the EM, the LM, and the CSA algorithms.

**Table 2:** Iterations and time requirements of the EM, the LM, and the CSA algorithm as they are applied to segment benchmark images (see Figure 14).

Iterations Time elapsed	(a)	(b)	(c)
EM	1855	1833	1870
	2.72 s	2.70 s	2.73 s
LM	985	988	958
	4.03 s	4.04 s	4.98 s
CSA	201	188	282
	0.21 s	0.18 s	0.25 s

cost. Finally, Figure 14 below shows the segmented images as they are generated by each algorithm. By analyzing the images (a)–(c) in Figure 14, it is clear that the CSA approach presents better results when the segmented images are compared with the original ones. In case of the EM and LM algorithms, several stains are identified in regions where the intensity level must be considered homogenous.

## 7. Conclusions

In this paper, an automatic image multi-threshold approach based on the clonal selection algorithm (CSA) is proposed. The segmentation process is considered to be similar to an optimization problem. The algorithm approximates the 1-D histogram of a given image using a Gaussian mixture model whose parameters are calculated through the CSA. Each Gaussian function approximating the histogram represents a pixel class and therefore one threshold point.

Experimental evidence shows that the CSA has a compromise between its convergence time and its computational cost when it is compared to the expectation-maximization (EM) method and the Levenberg-Marquardt (LM) algorithm. Additionally, the CSA also exhibits a better performance under certain circumstances (initial conditions) on which it is well reported in the literature [14, 18] that the EM has underperformed. Finally, the results have shown that the stochastic search accomplished by the CSA method shows a consistent performance with no regard of the initial value and still showing a greater chance to reach the global minimum.

Although Table 2 indicates that the CSA method can yield better results in comparison to the EM and gradient-based methods, notice that the aim of our paper is not intended to beat all segmentation algorithms which have been proposed earlier but to show that the artificial immune systems can effectively serve as an attractive alternative to evolutionary algorithms which have been employed before to successfully segment images.

## References

- [1] T. Abak, U. Baris, and B. Sankur, “The performance evaluation of thresholding algorithms for optical character recognition,” in *Proceedings of the 4th International Conference on Document Analysis and Recognition*, pp. 697–700, August 1997.
- [2] M. Kamel and A. Zhao, “Extraction of binary character/graphics images from grayscale document images,” *Graphical Models and Image Processing*, vol. 55, no. 3, pp. 203–217, 1993.

- [3] O. D. Trier and A. K. Jain, "Goal-directed evaluation of binarization methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 12, pp. 1191–1201, 1995.
- [4] B. Bhanu, "Automatic target recognition: state of the art survey," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 22, no. 4, pp. 364–379, 1986.
- [5] M. Sezgin and B. Sankur, "Selection of thresholding methods for non-destructive testing applications," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '01)*, pp. 764–767, October 2001.
- [6] M. Sezgin and R. Taşaltın, "A new dichotomization technique to multilevel thresholding devoted to inspection applications," *Pattern Recognition Letters*, vol. 21, no. 2, pp. 151–161, 2000.
- [7] R. Guo and S. M. Pandit, "Automatic threshold selection based on histogram modes and a discriminant criterion," *Machine Vision and Applications*, vol. 10, no. 5-6, pp. 331–338, 1998.
- [8] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognition*, vol. 26, no. 9, pp. 1277–1294, 1993.
- [9] P. K. Sahoo, S. Soltani, A. K. C. Wong, and Y. C. Chen, "A survey of thresholding techniques," *Computer Vision, Graphics and Image Processing*, vol. 41, no. 2, pp. 233–260, 1998.
- [10] W. Snyder, G. Bilbro, A. Logenthiran, and S. Rajala, "Optimal thresholding-a new approach," *Pattern Recognition Letters*, vol. 11, no. 12, pp. 803–810, 1990.
- [11] S. Chen and M. Wang, "Seeking multi-thresholds directly from support vectors for image segmentation," *Neurocomputing*, vol. 67, no. 4, pp. 335–344, 2005.
- [12] L. Chih-Chin, "A novel image segmentation approach based on particle swarm optimization," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 89, no. 1, pp. 324–327, 2006.
- [13] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Addison Wesley, Boston, Mass, USA, 1992.
- [14] L. Gupta and T. Sortrakul, "A gaussian-mixture-based image segmentation algorithm," *Pattern Recognition*, vol. 31, no. 3, pp. 315–325, 1998.
- [15] A. P. Dempster, A. P. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society B*, vol. 39, no. 1, pp. 1–38, 1997.
- [16] Z. Zhang, C. Chen, J. Sun, and L. Chan, "EM algorithms for gaussian mixtures with split-and-merge operation," *Pattern Recognition*, vol. 36, no. 9, pp. 1973–1983, 2003.
- [17] H. Park, S. Amari, and K. Fukumizu, "Adaptive natural gradient learning algorithms for various stochastic models," *Neural Networks*, vol. 13, no. 7, pp. 755–764, 2000.
- [18] H. Park and T. Ozeki, "Singularity and slow convergence of the em algorithm for gaussian mixtures," *Neural Processing Letters*, vol. 29, no. 1, pp. 45–59, 2009.
- [19] E. Cuevas, D. Zaldivar, and M. Pérez-Cisneros, "Seeking multi-thresholds for image segmentation with learning automata," *Machine Vision and Applications*, vol. 22, no. 5, pp. 805–818, 2011.
- [20] E. Cuevas, D. Zaldivar, and M. Pérez-Cisneros, "A novel multi-threshold segmentation approach based on differential evolution optimization," *Expert Systems with Applications*, vol. 37, no. 7, pp. 5265–5271, 2010.
- [21] G. A. Goldsby, T. J. Kindt, J. Kuby, and B. A. Osborne, *Immunology*, W. H. Freeman, New York, NY, USA, 5th edition, 2003.
- [22] L. N. de Castro and J. Timmis, *Artificial Immune Systems: A New Computational Intelligence Approach*, Springer, London, UK, 2002.
- [23] D. Dasgupta, "Advances in artificial immune systems," *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 40–49, 2006.
- [24] X. Wang, X. Z. Gao, and S. J. Ovaska, "Artificial immune optimization methods and applications—a survey," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '04)*, pp. 3415–3420, The Hague, The Netherlands, October 2004.
- [25] 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.
- [26] G. L. Ada and G. Nossal, "The clonal-selection theory," *Scientific American*, vol. 257, no. 2, pp. 50–57, 1987.
- [27] C. A. C. Coello and N. C. Cortés, "Solving multiobjective optimization problems using an artificial immune system," *Genetic Programming and Evolvable Machines*, vol. 6, no. 2, pp. 163–190, 2005.
- [28] F. Campelo, F. G. Guimarães, H. Igarashi, and J. A. Ramírez, "A clonal selection algorithm for optimization in electromagnetics," *IEEE Transactions on Magnetics*, vol. 41, no. 5, pp. 1736–1739, 2005.

- [29] W. Dong, G. Shi, and L. Zhang, "Immune memory clonal selection algorithms for designing stack filters," *Neurocomputing*, vol. 70, no. 4–6, pp. 777–784, 2007.
- [30] M. Gong, L. Jiao, L. Zhang, and H. Du, "Immune secondary response and clonal selection inspired optimizers," *Progress in Natural Science*, vol. 19, no. 2, pp. 237–253, 2009.
- [31] V. Cutello, G. Narzisi, G. Nicosia, and M. Pavone, "Clonal selection algorithms: a comparative case study using effective mutation potentials," in *Proceedings of the 4th International Conference on Artificial Immune Systems, (ICARIS '05)*, C. Jacob et al., Ed., pp. 13–28, August 2005.
- [32] M. Gong, L. Jiao, and X. Zhang, "A population-based artificial immune system for numerical optimization," *Neurocomputing*, vol. 72, no. 1–3, pp. 149–161, 2008.
- [33] J. Yoo and P. Hajela, "Immune network simulations in multicriterion design," *Structural and Multidisciplinary Optimization*, vol. 18, no. 2, pp. 85–94, 1999.
- [34] X. Wang, X. Z. Gao, and S. J. Ovaska, "A hybrid optimization algorithm in power filter design," in *Proceedings of the 31st Annual Conference of IEEE Industrial Electronics Society (IECON '05)*, pp. 1335–1340, Raleigh, NC, USA, November 2005.
- [35] X. Xu and J. Zhang, "An improved immune evolutionary algorithm for multimodal function optimization," in *Proceedings of the 3rd International Conference on Natural Computation (ICNC '07)*, pp. 641–646, Haikou, China, August 2007.
- [36] T. Tang and J. Qiu, "An improved multimodal artificial immune algorithm and its convergence analysis," in *Proceedings of the 6th World Congress on Intelligent Control and Automation (WCICA '06)*, pp. 3335–3339, Dalian, China, June 2006.

*Research Article*

## **Quality Improvement and Robust Design Methods to a Pharmaceutical Research and Development**

**Byung Rae Cho<sup>1</sup> and Sangmun Shin<sup>2</sup>**

<sup>1</sup> Department of Industrial Engineering, Clemson University, Clemson, SC 29634, USA

<sup>2</sup> Department of Systems Management & Engineering, Inje University,  
Gyeongnam Gimhae 621-749, Republic of Korea

Correspondence should be addressed to Sangmun Shin, sshin@inje.ac.kr

Received 19 November 2011; Accepted 18 April 2012

Academic Editor: Dongdong Ge

Copyright © 2012 B. R. Cho and S. Shin. 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.

Researchers often identify robust design, based on the concept of building quality into products or processes, as one of the most important systems engineering design concepts for quality improvement and process optimization. Traditional robust design principles have often been applied to situations in which the quality characteristics of interest are typically time-insensitive. In pharmaceutical manufacturing processes, time-oriented quality characteristics, such as the degradation of a drug, are often of interest. As a result, current robust design models for quality improvement which have been studied in the literature may not be effective in finding robust design solutions. In this paper, we show how the robust design concepts can be applied to the pharmaceutical production research and development by proposing experimental and optimization models which should be able to handle the time-oriented characteristics. This is perhaps the first attempt in the robust design field. An example is given, and comparative studies are discussed for model verification.

### **1. Introduction**

Continuous quality improvement has become widely recognized by many industries as a critical concept in maintaining a competitive advantage in the marketplace. It is also recognized that quality improvement activities are efficient and cost-effective when implemented during the design stage. Based on this awareness, Taguchi [1] introduced a systematic method for applying experimental design, which has become known as robust design which is often referred to as robust parameter design. The primary goal of this method is to determine the best design factor settings by minimizing performance variability and product bias, that is, the deviation from the target value of a product. Because of the practicability in reducing

the inherent uncertainty associated with system performance, the widespread application of robust design techniques has resulted in significant improvements in product quality, manufacturability, and reliability at low cost. Although the main robust design principles have been implemented in a number of different industrial settings, our literature study indicates that robust design has been rarely addressed in the pharmaceutical design process.

In the pharmaceutical industry, the development of a new drug is a lengthy process involving laboratory experiments. When a new drug is discovered, it is important to design an appropriate pharmaceutical dosage or formulation for the drug so that it can be delivered efficiently to the site of action in the body for the optimal therapeutic effect on the intended patient population. The Food and Drug Administration (FDA) requires that an appropriate assay methodology for the active ingredients of the designed formulation be developed and validated before it can be applied to animal or human subjects. Given this fact, one of the main challenges faced by many researchers during the past decades is the optimal design of pharmaceutical formulations to identify better approaches to various unmet clinical needs. Consequently, the pharmaceutical industry's large investment in the research and development (R&D) of new drugs provides a great opportunity for research in the areas of experimentation and design of pharmaceutical formulations. By definition, pharmaceutical formulation studies are mixture problems. These types of problems take into account the proportions within the mixture, not the amount of the ingredient; thus, the ingredients in such formulations are inherently dependent upon one another, and consequently experimental design methodologies commonly used in many manufacturing settings may not be effective. Instead, for mixture problems, a special kind of experimental design, referred to as a mixture experiment, is needed. In mixture experiments, typical factors in question are the ingredients of a mixture, and the quality characteristic of interest is often based on the proportionality of each of those ingredients. Hence, the quality of the pharmaceutical product is influenced by such designs when they are applied in the early stages of drug development.

In this paper, we propose a new robust design model in the context of pharmaceutical production R&D. The main contribution of this paper is twofold. First, traditional experimental design methods have often applied to situations in which the quality characteristics of interest are typically time-insensitive. In pharmaceutical manufacturing processes, time-oriented quality characteristics, such as the degradation of a drug, are often of interest, and these time-oriented data often follow a Weibull distribution. Since it may take a long time to observe the degradation of a drug product, the concept of censored samples can be integrated in designing optimal pharmaceutical formulations. In this paper, we develop a censored sample-based experimental design model for optimal pharmaceutical formulations by integrating the main robust design principles. Second, we then show how the response surface methodology, which is a well-established statistical tool, can be integrated with the proposed censored sample-based robust design model. Finally, we show how the maximum likelihood method is implemented in estimating mean and variance of censored Weibull data. A numerical example is given, and comparison studies for the two estimation methods are discussed for model verification. This paper is organized as follows. In the next section, we present a literature review on mixture design and robust design. In Section 3, we describe our proposed censored robust design model for the optimal design of pharmaceutical formulations in detail. The maximum likelihood method is then studied, and optimization models are proposed. In Section 4, we demonstrate our proposed methods using a numerical example and compare the results under the two different optimization models. In the last section, we conclude the paper with a discussion of our findings.

## 2. Literature Study

In this section, the literature of robust design and mixture designs is discussed.

### 2.1. Robust Design

Because product performance is directly related to product quality, Taguchi's techniques [1, 2] of robust design (RD) have become increasingly popular in industry since the mid 1980s. RD is a powerful and cost-effective quality improvement methodology for products and processes, which results in higher customer satisfaction and operational performance. There is little disagreement among researchers and practitioners about Taguchi's basic philosophy. Steinberg and Bursztyn [3] provided a comprehensive discussion on Taguchi's off-line quality control and showed that the use of noise factors can significantly increase the capability for detecting factors with dispersion effects, when noise factors are explicitly modeled in the analysis. However, the ad hoc robust design methods suggested by Taguchi remain controversial due to various mathematical flaws. The controversy surrounding Taguchi's assumptions, experimental design, and statistical analysis has been well addressed by Leon et al. [4], Box [5], Box et al. [6], Nair [7], and Tsui [8]. Consequently, researchers have closely examined alternative approaches using well-established statistical and optimization tools. Vining and Myers [9] introduced the dual response approach based on response surface methodology (RSM) as an alternative for modeling process relationships by separately estimating the response functions of process mean and variance, thereby achieving the primary goal of robust design by minimizing the process variance while adjusting the process mean at the target. Del Castillo and Montgomery [10] and Copeland and Nelson [11] showed that the optimization technique used by Vining and Myers [9] does not always guarantee optimal robust design solutions and proposed standard nonlinear programming techniques, such as the generalized reduced gradient method and the Nelder-Mead simplex method, which can provide better robust design solutions. The modified dual response approaches using fuzzy theory were further developed by Khattree [12] and Kim and Lin [13]. However, Lin and Tu [14], pointing out that the robust design solutions obtained from the dual response model may not necessarily be optimal since this model forces the process mean to be located at the target value, proposed the mean-squared error model, relaxing the zero-bias assumption. While allowing some process bias, the resulting process variance is less than or at most equal to the variance obtained from the Vining and Myers model [9]; hence, the mean-squared error model may provide better (or at least equal) robust design solutions unless the zero-bias assumption must be met. Further modifications to the mean-squared error model have been discussed by Jayaram and Ibrahim [15], Cho et al. [16], Kim and Cho [17, 18], Yue [19], Park and Cho [20], Miro-Quesada and Del Castillo [21], Cho and Park [22], Govindaluri and Cho [23], Shin and Cho [24, 25], and Lee et al. [26]. Along this line, Myers et al. [27], Park and Cho [22], and Robinson et al. [28] developed modified dual-response models using generalized linear model, a robust design model using the weighted-least-square method for unbalanced data, and a robust design model using a generalized linear mixed model for nonnormal quality characteristics, respectively. As for an experimental strategy, Kovach and Cho [29–31] and Kovach et al. [32] studied D-optimal robust design problems by minimizing the variance of the regression coefficients. Ginsburg and Ben-Gal [33] then developed a new optimality criterion, called the Vs optimality, which minimizes the variance of the optimal solution by prioritizing the estimation of various model coefficients, thereby, estimating coefficients more accurately at each experimental stage. It is well known

that estimated empirical models are often subject to random error. In order to obtain a more precise robust design solution in the presence of the error, Xu and Albin [34] developed a model which can be resistant to the error by considering all points in the confidence intervals associated with the estimated model. When multiple quality characteristics are considered, those characteristics are often correlated. Govindaluri and Cho [23], investigated the effect of correlations of quality characteristics on robust design solutions, while Egorov et al. [35] and Kovach et al. [32] studied optimal robust design solutions using the indirect optimization algorithm and physical programming, respectively. Finally, Shin and Cho [25] studied trade-off studies on minimizing variance and achieving the predetermined target value.

## 2.2. Mixture Designs

Scheffe [36] first introduced his theory on the prediction of the responses of mixture experiments based on their proportions. The theory defines  $x_i$  as the proportion of ingredient  $i$  in the mixture. Furthermore, the proportionality idea of this theory provides the experiment with a property in which the proportions of the  $k$  ingredients within the mixture must equal 100 percent, as illustrated by the equation  $\sum_{i=1}^k x_i = x_1 + x_2 + \dots + x_k = 1$ , where  $x_i \geq 0$  for all  $i = 1, 2, \dots, k$ . Scheffe [36] employed a simplex lattice design to represent the design points of the feasible experimental region of the ingredients. The simplex lattice design is defined by the notation  $\{k, m\}$ , where  $m + 1$  defines the number of equally spaced proportion values from 0 to 1 for each experiment and those proportions are determined by the equation  $x_i = 0, 1/m, 2/m, \dots, 1$ . All possible combinations of the proportions are used to determine the design points within the simplex lattice design. In general, the number of design points in a  $\{k, m\}$  simplex lattice design is defined  $n = (k + m - 1)! / m!(k - 1)!$  Scheffe [37] also modified this simplex lattice design to introduce the simplex centroid design for experiments that include the overall centroid of the region at the coordinate  $(1/k, 1/k, \dots, 1/k)$ .

Augmented designs of both the simplex lattice and simplex centroid designs exist. Cornell [38] analyzed both an augmented simplex lattice design and an augmented simplex centroid design with ten design points each. Applications of mixture experiments revealed other design possibilities. The most natural obstacle is the limitation on the proportion of a certain ingredient within a mixture. The limitation could be found in the form of lower, upper, and both lower and upper bounds or constraints. This led researchers to develop other ways to obtain design points that are within the feasible region given the constraints. An example of such models is the extreme vertices design for mixture experiments. First introduced by Mclean and Anderson [39], extreme vertices designs for mixture problems consider the extreme points of the irregular polyhedron, formed by constraints in experimental runs, in addition to the centroids of each facet. The major disadvantage with this design is the possible large number of design points that can be obtained with the given constraints, specifically as the number of ingredients increases and the feasible region becomes more complex. Snee and Marquart [40] presented an algorithm to determine the appropriate subset of design points when the vertices of the feasible region are too many to handle. They compare the efficiency of their approach to G- and D-optimal designs, both of which are common techniques used for determining the appropriate points at which to take observations. Bayesian D-optimal designs shown by DuMouchel and Jones [41] are a modification of D-optimal designs, which reduces the dependency of the design on the assumed model. Using such models as a leverage point, Andere-Rendon et al. [42] investigated the Bayesian D-optimal design specifically for mixture experiments which include both potential and primary model terms

in order to form the Bayesian slack variable model. The results favored the Bayesian D-optimal design with smaller bias errors and better-fitted models. Along the same lines, Goos and Donev [43] extended the work of Donev [44] with the implementation of D-optimal designs for blocked mixture experiments. Unlike other research that used orthogonally blocked experiments (see [45, 46]), they employed mixture designs that are not orthogonally blocked and used an algorithm that provided a simpler methodology to construct blocked mixture designs.

The simplified polynomials, also referred to as canonical polynomials, are widely used throughout the literature and are embedded in software packages for mixture experiments. However, these designs have been scrutinized, especially because of their lack of squared terms. Piepel et al. [47] proposed a partial quadratic mixture model that includes the linear Scheffé terms but augments them with the appropriate squared or quadratic cross product terms. Extending from alternative models proposed by Snee and Rayner [48], Cornell and Gorman [49] explained how highly constrained regions, such as those in mixture experiments having components with considerably smaller ranges than others, result in skewed responses, thus creating fitted models that have inherent collinearity. Both models attempt to modify the scale on the feasible region that results in the experiment's constraints in an effort to eliminate the collinearity between components. For other collinearity research and discussions, refer to the publications of Sengupta et al. [50] and Prescott et al. [51]. Other research publications have employed robust design methodologies for mixture experiments. Steiner and Hamada [52] modeled a mixture experiment to include the coefficient and terms that account for the interactions of mixture and controllable process variables and the interactions of the mixture and noise variables. This model utilizes the Taguchi loss function [1, 2] to reduce the noise variables. The practicality of the design may be questionable as it does not account for constraints within the system, which opens the opportunity for further research. A different approach to dealing with noise factors can be studied in Goldfarb et al. [53]. Continuing in similar research, Goldfarb et al. [54] introduced a three-dimensional variance dispersion graph with the purpose of comparing competing mixture experimental designs based on their prediction variance properties. Goldfarb et al. [55] proposed an interesting addition to this research by implementing genetic algorithms within an objective function to minimize the maximum scaled prediction variance in the mixture design region. This investigation showed that with a few runs of the genetic algorithm, the scaled prediction variance can be significantly reduced, allowing the experimenter to control noise variables inherent in the experiment.

### 3. Proposed Censored Robust Design Model

In this section, we describe the proposed model in three phases—experimental phase, estimation phase, and optimization phase.

#### 3.1. Notations

Notations associated with parameters and variables used in paper are defined as follows:

$\mathbf{x} = [x_1, x_2, \dots, x_k]$  vector of  $k$  control factors,

$\mathbf{y}$  vector of output observations,

$y_i$ :  $i$  output observations,

$\mathbf{T}$  censored observations,

**Table 1:** General layout of the proposed methodology.

Design points	Control factors	Experimental observations	Estimate	
	$x_1, x_2, \dots, x_f$		Mean	Variance
1		$y_1, \dots, y_{n_1}, T_1, \dots, T_{m_1}$	$\hat{\mu}_1$	$\hat{\sigma}_1$
2		$y_1, \dots, y_{n_2}, T_1, \dots, T_{m_2}$	$\hat{\mu}_2$	$\hat{\sigma}_2$
$\vdots$	Experimental design	$\vdots$	$\vdots$	$\vdots$
$i$		$y_1, \dots, y_{n_i}, T_1, \dots, T_{m_i}$	$\hat{\mu}_i$	$\hat{\sigma}_i$
$\vdots$		$\vdots$	$\vdots$	$\vdots$
$d$		$y_1, \dots, y_{n_d}, T_1, \dots, T_{m_d}$	$\hat{\mu}_d$	$\hat{\sigma}_d$

$T_i$   $i$  censored observations,

$\boldsymbol{\theta}$  vector of parameter  $\theta$ ,

$\theta_k$  the component of vector parameter  $\boldsymbol{\theta}$ ,

$F(\mathbf{y}; \boldsymbol{\theta})$  cumulative distribution function associated with parameters  $\mathbf{y}$  and  $\boldsymbol{\theta}$ ,

$f(\mathbf{y}; \boldsymbol{\theta})$  probability density function associated with parameters  $\mathbf{y}$  and  $\boldsymbol{\theta}$ ,

$L(\mathbf{y}, \mathbf{T}; \boldsymbol{\theta})$  maximum likelihood function associated with parameters  $\mathbf{y}$ ,  $\mathbf{T}$ , and  $\boldsymbol{\theta}$ ,

$l(\mathbf{y}, \mathbf{T}; \boldsymbol{\theta})$  loglikelihood function of  $L(\mathbf{y}, \mathbf{T}; \boldsymbol{\theta})$ ,

$\Gamma(x)$  gamma function,

$\hat{\mu}(\mathbf{x})$  estimated function of process mean,

$\hat{\sigma}(\mathbf{x})$  estimated function of process variance.

### 3.2. Experimental Phase

Table 1 displays the general framework of the RD methodology, where the  $y_i$ 's are the observed pharmaceutical quality characteristic values and the  $T_i$ 's are the censored times. The mean and variance for each design point are estimated using the observations.

### 3.3. Estimation Phase

Observations are of two kinds—actual and censored observations. Assume that the observations follow a distribution with underlying cumulative distribution function  $F(\mathbf{y}; \boldsymbol{\theta})$  and probability density function  $f(\mathbf{y}; \boldsymbol{\theta})$ , where  $\boldsymbol{\theta}$  is a vector of parameters and  $\mathbf{y}$  is a vector of observations. Suppose that for each design point, we have  $n$  actual observations  $y_1, y_2, \dots, y_n$  and  $m$  censored observations  $T_1, T_2, \dots, T_m$ . Then the maximum likelihood function is

$$L(\mathbf{y}, \mathbf{T}; \boldsymbol{\theta}) = \prod_{i=1}^n f(y_i; \boldsymbol{\theta}) \prod_{j=1}^m [1 - F(T_j; \boldsymbol{\theta})], \quad (3.1)$$

and the loglikelihood function is

$$l(\mathbf{y}, \mathbf{T}; \boldsymbol{\theta}) = \sum_{i=1}^n \ln f(y_i; \boldsymbol{\theta}) + \sum_{j=1}^m \ln[1 - F(T_j; \boldsymbol{\theta})]. \quad (3.2)$$

If the censoring time is fixed at  $T$ , say, then  $\prod_{j=1}^m [1 - F(T_j; \boldsymbol{\theta})] = [1 - F(T; \boldsymbol{\theta})]^m$ . In that case, the loglikelihood function is

$$l(\mathbf{y}, \mathbf{T}; \boldsymbol{\theta}) = \sum_{i=1}^n \ln f(y_i; \boldsymbol{\theta}) + m \ln[1 - F(T; \boldsymbol{\theta})]. \quad (3.3)$$

The values of the components of  $\boldsymbol{\theta}$  that maximize the loglikelihood function will constitute the maximum likelihood estimates. These are computed as the solutions to the system of equations:

$$\frac{\partial}{\partial \theta_k} l(\mathbf{y}, \mathbf{T}; \boldsymbol{\theta}) = 0, \quad (3.4)$$

where the  $\theta_k$ 's are the components of the vector of parameters  $\boldsymbol{\theta}$ . If the underlying distribution follows a Weibull distribution with parameters  $\alpha$  and  $\beta$ , then  $\boldsymbol{\theta} = [\alpha, \beta]$ , and for the  $n$  actual observations  $y_1, y_2, \dots, y_n$ , and  $m$  censored observations  $T_1, T_2, \dots, T_m$ , the loglikelihood function is

$$l(\mathbf{y}, \mathbf{T}; \mu; \sigma) = n(\ln \alpha + \ln \beta) + \sum_{i=1}^n (\beta - 1) \ln y_i - \alpha \sum_{i=1}^n y_i^\beta - \alpha \sum_{j=1}^m T_j^\beta. \quad (3.5)$$

For a fixed censoring time  $T$ ,

$$l(\mathbf{y}, \mathbf{T}; \mu; \sigma) = n(\ln \alpha + \ln \beta) + \sum_{i=1}^n (\beta - 1) \ln y_i - \alpha \sum_{i=1}^n y_i^\beta - \alpha m T. \quad (3.6)$$

The maximum likelihood estimates of  $\alpha$  and  $\beta$  are the solutions to the system of equations:

$$\begin{aligned} \frac{n}{\alpha} - \sum_{i=1}^n y_i^\beta - m T^\beta &= 0, \\ \frac{n}{\beta} + \sum_{i=1}^n \ln y_i - \alpha \sum_{i=1}^n y_i^\beta \ln y_i - \alpha m T^\beta \ln T &= 0. \end{aligned} \quad (3.7)$$

Equation (3.7) is obtained by taking the derivatives of the loglikelihood function with respect to  $\alpha$  and  $\beta$ . Similarly, a system of equations can be derived using (3.6) for varying censoring

times. The solutions to (3.7), namely,  $\hat{\alpha}$  and  $\hat{\beta}$ , are the maximum likelihood estimates of  $\alpha$  and  $\beta$ , and we use them in estimating the mean and standard deviation as follows:

$$\begin{aligned}\hat{\mu} &= \hat{\alpha}^{1/\hat{\beta}} \Gamma\left(1 + \frac{1}{\hat{\beta}}\right), \\ \hat{\sigma} &= \sqrt{\hat{\alpha}^{2/\hat{\beta}} \Gamma\left(1 + \frac{2}{\hat{\beta}}\right) - \hat{\mu}^2}.\end{aligned}\tag{3.8}$$

### 3.4. Optimization Phase

The main objective of the proposed robust design is to obtain the optimal pharmaceutical formulation settings by maximizing mean response while minimizing variability. Thus, in order to meet this goal, we seek to maximize the mean response while minimizing the variability. To achieve this objective, we propose the following optimization model (Model 1):

$$\begin{aligned}\min_{x \in \Omega} \quad & \left\{ -\mu^2(x) + \sigma^2(x) \right\} \\ \text{s.t.} \quad & \sum_{i=1}^k x_i = 1.\end{aligned}\tag{3.9}$$

It is noted that the sum of pharmaceutical component proportion is one. By considering the usual approximation of the Taguchi's loss function [1], we can also find the solution to the following optimization model (Model 2):

$$\begin{aligned}\min_{x \in \Omega} \quad & \left\{ \frac{1}{\mu^2(x)} \left( 1 + 3 \frac{\sigma^2(x)}{\mu^2(x)} \right) \right\} \\ \text{s.t.} \quad & \sum_{i=1}^k x_i = 1.\end{aligned}\tag{3.10}$$

By inspection, we notice that this function decreases as  $\mu(x)$  increases and as  $\sigma(x)$  decreases. Also, a part of the feasibility requirements for these proposed objective functions is that the mean response is nonzero, that is,  $\mu(x) \neq 0$ , which is the case for censored samples where the objective is to get  $\mu(x)$  as large as possible. We will demonstrate that both proposed optimization models yield optimal solutions in the next section.

## 4. Numerical Example and Comparison Study

Consider an experiment on the degradation of a drug where the factors of concern are corn starch ( $x_1$ ), saccharin sodium ( $x_2$ ), and dextrose ( $x_3$ ). Thus, the vector of the control factors is  $x = [x_1, x_2, x_3]$ . The objective of the experiment is to determine the settings of the control factors,  $x^* = [x_1^*, x_2^*, x_3^*]$ , that give the longest possible degradation and the smallest possible variability. The chosen design is a mixture design for three control factors. Suppose

**Table 2:** Mixture experiments and data.

$x_1$	$x_2$	$x_3$	Observations
1	0	0	30+, 30+, 27.3130, 30+, 25.3841, 18.4713, 26.3209, 25.1345, 32.6499, 30+
0	1	0	30+, 20.5247, 9.9338, 22.1908, 24.0196, 16.9328, 18.3113, 21.5325, 21.4749, 17.7806
0	0	1	11.6089, 24.6178, 18.2190, 21.9588, 27.6887, 21.6017, 26.7790, 13.8795, 24.6055, 30+
1/2	1/2	0	24.8139, 27.3573, 30+, 23.8765, 30+, 15.8716, 30+, 30+, 25.4216, 30+
1/2	0	1/2	29.3682, 30+, 34.0853, 28.2447, 21.2239, 19.7356, 30+, 23.1654, 20.4046
0	1/2	1/2	23.8148, 14.5317, 25.6167, 23.4480, 30+, 18.7073, 30+, 41.6825, 24.0814, 14.2608
1/6	1/6	2/3	30+, 27.6718, 30+, 27.1105, 18.4573, 28.0881, 30+, 28.7328, 21.7883, 23.6375
1/6	2/3	1/6	14.8984, 19.5179, 27.8163, 24.6057, 30+, 26.3950, 28.0956, 18.4866, 18.7682, 27.7717
2/3	1/6	1/6	23.0132, 25.5782, 30+, 30+, 28.1479, 30+, 30+, 19.7557, 30+
2/3	1/3	1/3	30+, 28.9475, 19.3873, 30+, 27.4342, 28.5594, 30+, 30+, 17.7312, 21.8007

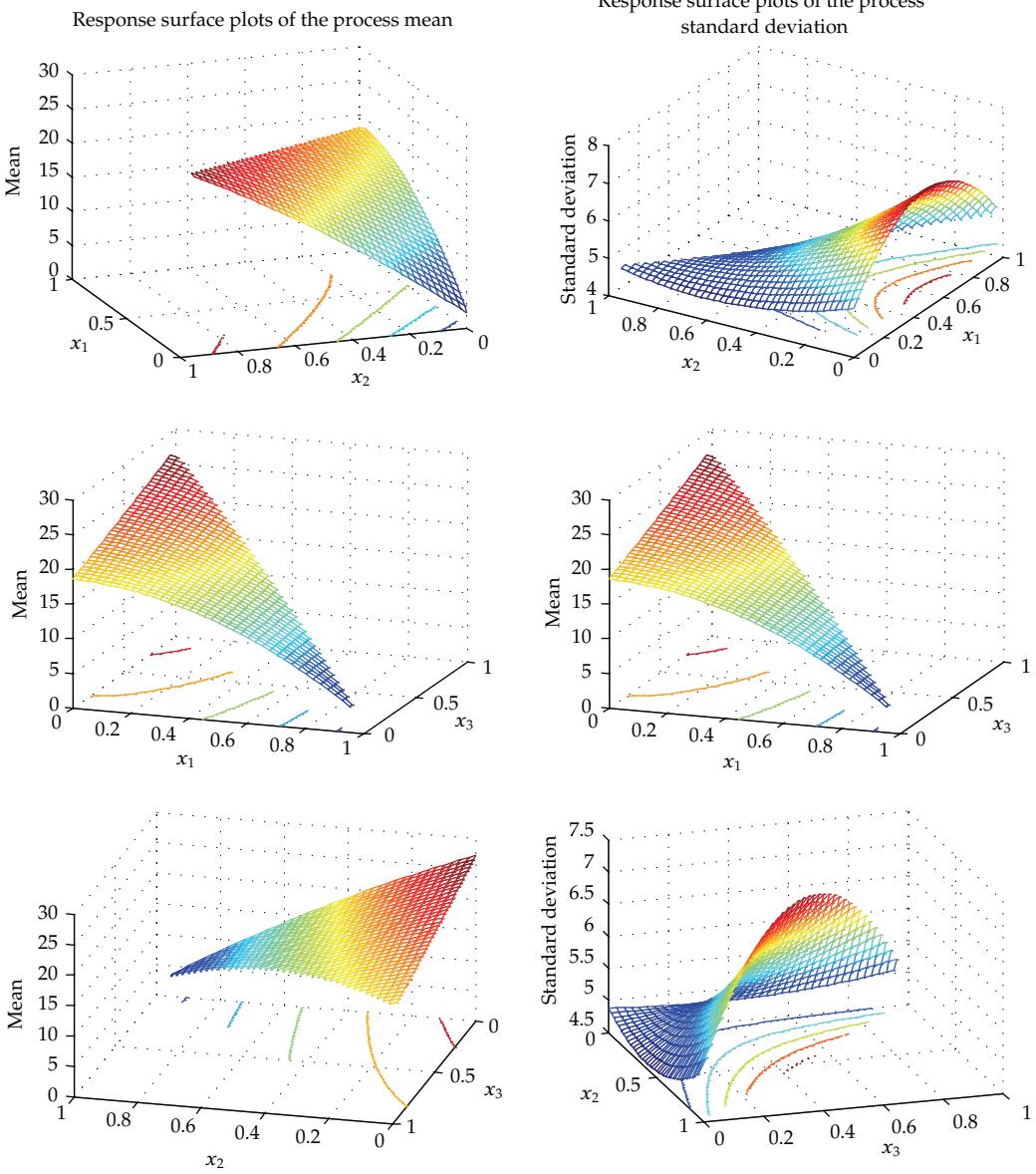
**Table 3:** Parameter estimates.

Design point	$x_1$	$x_2$	$x_3$	Estimate	
				$\hat{\mu}$	$\hat{\sigma}$
1	1	0	0	25.6885	4.5176
2	0	1	0	18.2861	5.0843
3	0	0	1	20.8556	5.5625
4	1/2	1/2	0	22.3393	5.4712
5	1/2	0	1/2	25.1136	5.4931
6	0	1/2	1/2	23.3288	8.4396
7	1/6	1/6	2/3	24.5420	4.1651
8	1/6	2/3	1/6	22.6876	4.9773
9	2/3	1/6	1/6	22.8861	5.1334
10	2/3	1/3	1/3	23.3731	5.3724

10 samples are subject to each design point, and the experiment is run for 30 hours. The manufacturer believes that the degradation times follow a Weibull distribution. Notice that this is a censoring problem because the experiment is terminated after 30 hours, and for the rest of the samples denoted by 30+, we only know that their degradation times exceed 30 hours but we do not know their actual times. Using the functions in the methods of Sections 3.2 and 3.3, estimates of the means and standard deviations are obtained for each design point. Table 2 shows the experimental design together with the observed responses that have failed. Note that the number of observations per design point is not the same throughout the experiment.

In Table 3, we show the estimates of the mean and standard deviation obtained by our proposed algorithm and by the method of ML estimation. Using the MLE algorithm, the response surfaces for the mean and standard deviation are, respectively, found as

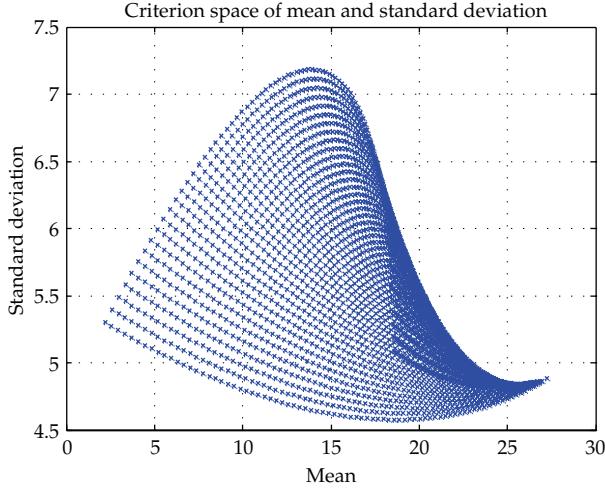
$$\begin{aligned}\hat{\mu}(\mathbf{x}) &= 27.222x_1 + 18.601x_2 + 2.190x_3 - 0.318x_1x_2 + 5.680x_1x_3 + 14.908x_2x_3, \\ \hat{\sigma}(\mathbf{x}) &= 4.884x_1 + 5.088x_2 + 5.300x_3 - 0.568x_1x_2 - 1.917x_1x_3 + 7.957x_2x_3.\end{aligned}\tag{4.1}$$



**Figure 1:** Response surface plots of the mean and standard deviation.

Using (4.1), Figure 1 illustrates surface plots of the process mean and standard deviation associated with input control factors ( $x_1, x_2, x_3$ ). In order to represent tradeoffs between the mean and standard deviation, a criterion space is demonstrated in Figure 2.

Table 4 shows the results obtained using the two different optimization models we studied earlier. For this particular example, Model 1 is more effective in increasing the mean value, while Model 2 turns out to be more effective in decreasing standard deviation. The basic goal of both optimization models is to obtain the optimal factor level settings which maximize the mean and minimize the standard deviation. We recommend practitioners use



**Figure 2:** Criterion space for the mean and standard deviation.

**Table 4:** Optimal settings under two models.

Design	Opt settings $x^* = [x_1^*, x_2^*, x_3^*]$	Opt mean $\hat{\mu}(x^*)$	Opt Std Dev $\hat{\sigma}(x^*)$
Model 1 based on (11)	[0.0000, 0.6587, 0.3413]	27.1974	6.9492
Model 2 based on (12)	[0.3332, 0.3336, 0.3333]	26.1309	5.6933

both optimization models and pick one of them, depending on their priority of mean and standard deviation.

## 5. Conclusions

Robust design has been demonstrated to be a successful process improvement methodology for diverse system engineering problems. However, its ability to provide sound solutions for time-oriented data is currently limited since the traditional robust design method is mainly applicable to time-insensitive data, and an appropriate robust design method for time-oriented data has not been reported. This paper has proposed new robust design experimental and optimization models for time-oriented data by developing a set of methods in three separate phases, namely, modeling, estimation, and optimization phases. More specifically, this paper has proposed a censored sample-based robust design model by developing a censored experimental design framework, a censored maximum likelihood estimation method in the context of robust design, and optimization models for the censored robust design. For verification, a proposed optimization model has been compared with the traditional Taguchi optimization model. The proposed experimental methodology is particularly useful for experiments with time-oriented pharmaceutical characteristics, such as the degradation of a drug, with an unequal number of observations at each design point. With suitable modifications, our proposed methodology could be extended to the case in which multiple quality characteristics are often correlated under study. Another area for

future study includes the development of optimal designs, known as computerized designs, for the case in which physical experimental constraints are imposed.

## Acknowledgment

This work was supported by the 2011 Inje University research grant.

## References

- [1] G. Taguchi, *Introduction to Quality Engineering*, Asian Productivity Organization, UNIPUB/Kraus International, White Plains, NY, USA, 1986.
- [2] G. Taguchi, *Systems of Experimental Design: Engineering Methods to Optimize Quality and Minimize Cost*, UNIPBUK/Kraus International, White Plains, NY, USA, 1987.
- [3] D. M. Steinberg and D. Bursztyn, "Noise factors, dispersion effects, and robust design," *Statistica Sinica*, vol. 8, no. 1, pp. 67–85, 1998.
- [4] R. V. León, A. C. Shoemaker, and R. N. Kacker, "Performance measures independent of adjustment. An explanation and extension of Taguchi's signal-to-noise ratios," *Technometrics*, vol. 29, no. 3, pp. 253–285, 1987.
- [5] G. E. P. Box, "Signal-to-noise ratios, performance criteria, and transformations," *Technometrics*, vol. 30, no. 1, pp. 1–17, 1988.
- [6] G. E. P. Box, S. Bisgaard, and C. Fung, "An explanation and critique of Taguchi's contributions to quality engineering," *Quality and Reliability Engineering International*, vol. 4, no. 2, pp. 123–131, 1988.
- [7] V. N. Nair, "Taguchi's parameter design: a panel discussion," *Technometrics*, vol. 34, no. 2, pp. 127–161, 1992.
- [8] K. L. Tsui, "An overview of Taguchi method and newly developed statistical methods for robust design," *IIE Transactions*, vol. 24, pp. 44–57, 1992.
- [9] G. G. Vining and R. H. Myers, "Combining Taguchi and response surface philosophies: a dual response approach," *Journal of Quality Technology*, vol. 22, pp. 38–45, 1990.
- [10] E. Del Castillo and D. C. Montgomery, "A nonlinear programming solution to the dual response problem," *Journal of Quality Technology*, vol. 25, no. 3, pp. 199–204, 1993.
- [11] K. A. F. Copeland and P. R. Nelson, "Dual response optimization via direct function minimization," *Journal of Quality Technology*, vol. 28, no. 3, pp. 331–336, 1996.
- [12] R. Khattree, "Robust parameter design: a response surface approach," *Journal of Quality Technology*, vol. 28, no. 2, pp. 187–198, 1996.
- [13] K. J. Kim and D. K. J. Lin, "Dual response surface optimization: a fuzzy modeling approach," *Journal of Quality Technology*, vol. 30, no. 1, pp. 1–10, 1998.
- [14] D. K. J. Lin and W. Tu, "Dual response surface optimization," *Journal of Quality Technology*, vol. 27, no. 1, pp. 34–39, 1995.
- [15] J. S. R. Jayaram and Y. Ibrahim, "Multiple response robust design and yield maximization," *International Journal of Quality and Reliability Management*, vol. 6, pp. 826–837, 1999.
- [16] B. R. Cho, Y. J. Kim, D. L. Kimbler, and M. D. Phillips, "An integrated joint optimization procedure for robust and tolerance design," *International Journal of Production Research*, vol. 38, no. 10, pp. 2309–2325, 2000.
- [17] Y. J. Kim and B. R. Cho, "Economic integration of design optimization," *Quality Engineering*, vol. 12, no. 4, pp. 561–567, 2000.
- [18] Y. J. Kim and B. R. Cho, "Determining the optimum process mean for a skewed process," *International Journal of Industrial Engineering*, vol. 10, no. 4, pp. 555–561, 2003.
- [19] R.-X. Yue, "Model-robust designs in multiresponse situations," *Statistics & Probability Letters*, vol. 58, no. 4, pp. 369–379, 2002.
- [20] C. Park and B. R. Cho, "Development of robust design under contaminated and non-normal data," *Quality Engineering*, vol. 15, no. 3, pp. 463–469, 2003.
- [21] G. Miro-Quesada and E. Del Castillo, "A dual response approach to the multivariate robust parameter design problem," *Technometrics*, vol. 46, no. 2, pp. 176–187, 2004.
- [22] B. R. Cho and C. Park, "Robust design modeling and optimization with unbalanced data," *Computers and Industrial Engineering*, vol. 48, no. 2, pp. 173–180, 2005.

- [23] S. M. Govindaluri and B. R. Cho, "Integration of customer and designer preferences in robust design," *International Journal of Six Sigma and Competitive Advantage*, vol. 1, no. 3, pp. 276–294, 2005.
- [24] S. Shin and B. R. Cho, "Bias-specified robust design optimization and its analytical solutions," *Computers and Industrial Engineering*, vol. 48, no. 1, pp. 129–140, 2005.
- [25] S. Shin and B. R. Cho, "Studies on a biobjective robust design optimization problem," *IIE Transactions*, vol. 41, no. 11, pp. 957–968, 2009.
- [26] S. B. Lee, C. Park, and B. R. Cho, "Development of a highly efficient and resistant robust design," *International Journal of Production Research*, vol. 45, no. 1, pp. 157–167, 2007.
- [27] W. R. Myers, W. A. Brenneman, and R. H. Myers, "A dual-response approach to robust parameter design for a generalized linear model," *Journal of Quality Technology*, vol. 37, no. 2, pp. 130–138, 2005.
- [28] T. J. Robinson, S. S. Wulff, D. C. Montgomery, and A. I. Khuri, "Robust parameter design using generalized linear mixed models," *Journal of Quality Technology*, vol. 38, no. 1, pp. 65–75, 2006.
- [29] J. Kovach and B. R. Cho, "A D-optimal design approach to robust design under constraints: a new Design for Six Sigma tool," *International Journal of Six Sigma and Competitive Advantage*, vol. 2, no. 4, pp. 389–403, 2006.
- [30] J. Kovach and B. R. Cho, "Development of a D-optimal robust design model for restricted experiments," *International Journal of Industrial Engineering*, vol. 14, pp. 117–128, 2007.
- [31] J. Kovach and B. R. Cho, "Solving multiresponse optimization problems using quality function-based robust design," *Quality Engineering*, vol. 20, no. 3, pp. 346–360, 2008.
- [32] J. Kovach, B. R. Cho, and J. Antony, "Development of an experiment-based robust design paradigm for multiple quality characteristics using physical programming," *International Journal of Advanced Manufacturing Technology*, vol. 35, no. 11-12, pp. 1100–1112, 2008.
- [33] H. Ginsburg and I. Ben-Gal, "Designing experiments for robust-optimization problems: the Vs-optimality criterion," *IIE Transactions*, vol. 38, no. 6, pp. 445–461, 2006.
- [34] D. Xu and S. L. Albin, "Robust optimization of experimentally derived objective functions," *IIE Transactions*, vol. 35, no. 9, pp. 793–802, 2003.
- [35] I. N. Egorov, G. V. Kretinin, I. A. Leshchenko, and S. V. Kuptsov, "Multi-objective approach for robust design optimization problems," *Inverse Problems in Science and Engineering*, vol. 15, no. 1, pp. 47–59, 2007.
- [36] H. Scheffé, "Experiments with mixtures," *Journal of the Royal Statistical Society. Series B*, vol. 20, pp. 344–360, 1958.
- [37] H. Scheffé, "The simplex-centroid design for experiments with mixtures," *Journal of the Royal Statistical Society. Series B*, vol. 25, pp. 235–263, 1963.
- [38] J. A. Cornell, *Experiments with Mixtures: Designs, Models, and the Analysis of Mixture Data*, John Wiley & Sons Inc., New York, NY, USA, 1981.
- [39] R. A. McLean and V. L. Anderson, "Linear estimates of parameters in the extreme value distribution," *Technometrics*, vol. 8, pp. 454–477, 1966.
- [40] D. D. Snee and D. W. Marquardt, "Extreme vertices designs for linear mixture models," *Technometrics*, vol. 16, no. 3, pp. 399–408, 1974.
- [41] W. DuMouchel and B. Jones, "A simple bayesian modification of d-optimal designs to reduce dependence on an assumed model," *Technometrics*, vol. 36, pp. 37–47, 1994.
- [42] J. Andere-Rendon, D. C. Montgomery, and D. A. Rollier, "Design of mixture experiments using bayesian D-optimality," *Journal of Quality Technology*, vol. 29, no. 4, pp. 451–463, 1997.
- [43] P. Goos and A. N. Donev, "The D-optimal design of blocked experiments with mixture components," *Journal of Quality Technology*, vol. 38, no. 4, pp. 319–332, 2006.
- [44] A. N. Donev, "Design of experiments with both mixture and qualitative factors," *Journal of the Royal Statistical Society. Series B*, vol. 51, no. 2, pp. 297–302, 1989.
- [45] A. K. Nigam, "Correction to: "Block designs for mixture experiments"," *The Annals of Statistics*, vol. 4, no. 6, pp. 1294–1295, 1976.
- [46] P. W. M. John, "Experiments with mixtures involving process variables," Tech. Rep. 8, Center for Statistical Sciences, University of Texas, Austin, Tex, USA, 1984.
- [47] G. F. Piepel, J. M. Szychowski, and J. L. Loeppky, "Augmenting Scheffé linear mixture models with squared and/or crossproduct terms," *Journal of Quality Technology*, vol. 34, no. 3, pp. 297–314, 2002.
- [48] R. D. Snee and A. A. Rayner, "Assessing the accuracy of mixture model regression calculations," *Journal of Quality Technology*, vol. 14, no. 2, pp. 67–79, 1982.

- [49] J. A. Cornell and J. W. Gorman, "Fractional design plans for process variables in mixture experiments," *Journal of Quality Technology*, vol. 16, no. 1, pp. 20–38, 1984.
- [50] T. K. Sengupta, R. K. Nandy, S. Mukhopadhyay et al., "Mixture models based on homogeneous polynomials," *Journal of Statistical Planning and Inference*, vol. 71, no. 1-2, pp. 303–311, 1998.
- [51] P. Prescott, N. R. Draper, A. M. Dean, and S. M. Lewis, "Mixture experiments: ILL-conditioning and quadratic model specification," *Technometrics*, vol. 44, no. 3, pp. 260–268, 2002.
- [52] S. H. Steiner and M. Hamada, "Making mixtures robust to noise and mixing measurement errors," *Journal of Quality Technology*, vol. 29, no. 4, pp. 441–450, 1997.
- [53] H. B. Goldfarb, C. M. Borror, and D. C. Montgomery, "Mixture-process variable experiments with noise variables," *Journal of Quality Technology*, vol. 35, no. 4, pp. 393–405, 2003.
- [54] H. B. Goldfarb, C. M. Borror, D. C. Montgomery, and C. M. Anderson-Cook, "Three-dimensional variance dispersion graphs for mixture-process experiments," *Journal of Quality Technology*, vol. 36, no. 1, pp. 109–124, 2004.
- [55] H. B. Goldfarb, C. M. Borror, D. C. Montgomery, and C. M. Anderson-Cook, "Using genetic algorithms to generate mixture-process experimental designs involving control and noise variables," *Journal of Quality Technology*, vol. 37, no. 1, pp. 60–74, 2005.

*Research Article*

# A Label Correcting Algorithm for Partial Disassembly Sequences in the Production Planning for End-of-Life Products

**Pei-Fang (Jennifer) Tsai**

*Department of Industrial Engineering and Management, National Taipei University of Technology,  
10608 Taipei, Taiwan*

Correspondence should be addressed to Pei-Fang (Jennifer) Tsai, ptsai@ntut.edu.tw

Received 8 February 2012; Accepted 8 April 2012

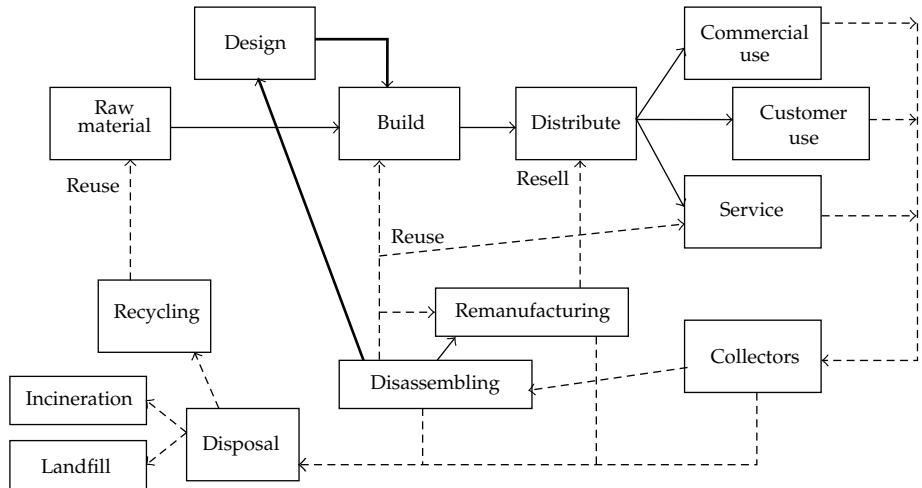
Academic Editor: Jung-Fa Tsai

Copyright © 2012 Pei-Fang (Jennifer) Tsai. 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.

Remanufacturing of used products has become a strategic issue for cost-sensitive businesses. Due to the nature of uncertain supply of end-of-life (EoL) products, the reverse logistic can only be sustainable with a dynamic production planning for disassembly process. This research investigates the sequencing of disassembly operations as a single-period partial disassembly optimization (SPPDO) problem to minimize total disassembly cost. AND/OR graph representation is used to include all disassembly sequences of a returned product. A label correcting algorithm is proposed to find an optimal partial disassembly plan if a specific reusable subpart is retrieved from the original return. Then, a heuristic procedure that utilizes this polynomial-time algorithm is presented to solve the SPPDO problem. Numerical examples are used to demonstrate the effectiveness of this solution procedure.

## 1. Introduction

Product recovery, or remanufacturing, has been considered as one of the most profitable options in dealing with the end-of-life (EoL) products. The benefit of product recovery is especially more attractive when a facility is capable of performing both manufacturing and remanufacturing processes, and the coordination of these two processes can be included in the production planning and scheduling. Griese et al. [1] discussed the economic challenges for reuse and the main technical obstacles in three product categories: medical equipment, automotive electronics, and computers. They argued that, for personal computers, reuse and repair appeared to have more potential than pure recycling materials. Similar benefit was confirmed by Grenchus et al. [2] from the practice at the IBM Endicott asset recovery center. They found that, with little disassembly effort, functional parts that were recoverable had more resale value than plain material recovery.



**Figure 1:** Integrated supply chain for original equipment manufacturers. Adopted from [4].

For any original equipment manufacturer (OEM) with the capacity in performing assembly and disassembly operations, the retrieved components from returned products can be integrated with their forward production. As shown in Figure 1, a complete product life cycle is defined by forward logistics (solid line) and reverse logistics (dotted line). Information (bold line) from disassembly operation can further assist in design and manufacturing new products. Retrieving spare parts from returned products has been one of the most prominent strategies. Fleischmann [3] observed that, when the returned used equipments were integrated into the spare part planning, a push policy in which returned equipments were tested and dismantled as soon as available can achieve higher service level in IBM's spare part management.

It remains as one of the biggest challenges to develop the techniques in production planning system for product recovery to be sustainable [5]. For regular assembly, demand can be determined in advance and hence the required resources can be planned and scheduled along the time horizon. However, for disassembly process planning, variation in quantity and quality of returned products is so huge that it fails to fit into any available planning scheme. Even when the demand for remanufactured products is known with a set of available returned products, it is still challenging to decide how these products should be dismantled to minimize the disassembly effort for those refurbished products. Kasmara et al. [6] used an integer programming model that included sales and returns in each period with the objective function as maximization of profits. Clegg et al. [7] presented a linear programming model of production planning for both new and remanufactured products. Some studies focused on the effect of average flowtime for both assembly and disassembly operations under different scenarios in planning mixes [8-13].

The ability to salvage the value of these returned products relies both on the disassembly capacity and the ability to find the most cost-effective disassembly sequences to retrieve valuable parts. This research is motivated to find disassembly sequences with minimum operation costs in the production planning for EoL products. A single period planning is considered due to the inherent fluctuation in the demand and supply of EoL products in different periods. Moreover, a partial disassembly policy is considered for better profit in product recovery.

## 2. Literature on Disassembly Planning

Disassembly is the process of dismantling a product through successive steps at the end of its life so as to recover useful components or subassemblies, for resale, reuse, or proper disposal. A product, as a whole, may not be repairable at the end of its conventional useful life, yet there might be components and subassemblies that are in good enough condition for use in a new product or in the remanufacture of an old one [14]. The efficient retrieval of these parts/subassemblies will not only cut down the production cost but will also reduce the associated disposal costs and, consequently, the environmental hazards.

Taleb and Gupta [15] addressed the scheduling problem of disassembly operations with commonality in the parts or materials. Then, a scheduling mechanism was provided to determine the total disassembly requirement as well as the time to release along the planning horizon. Langella [16] developed heuristics for the disassembly planning when a returned product was used in multiple purposes. Gungor and Gupta [17] used a heuristic algorithm to evaluate total costs among different disassembly strategies for returned products using the precedence relationship among subassemblies. Deterministic formulations originated from variations of travelling salesman problem (TSP) were used to find optimal disassembly sequences [18, 19]. Johnson and Wang [20] considered the disassembly precedence relationship according to a bill of material (BOM) of the product and formulated the problem as a two-commodity network formulation. Lambert [21] solved a similar formulation as in [20] with sequence-dependent setup time by using an iterative heuristic approach.

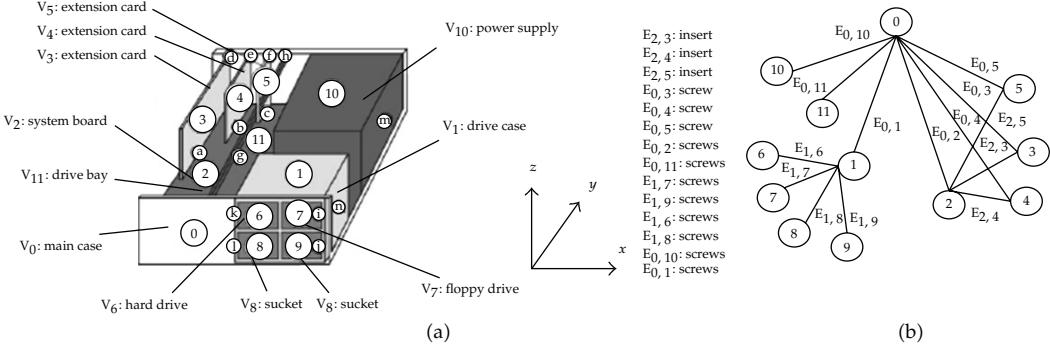
Graph-based representation is often used to find all feasible disassembly sequences. De Fazio and Whitney [22] proposed a network of liaison sequences, or precedence sequences, that are generated from the bill of material (BOM) of a product. Sarin et al. [19] incorporated the precedence constraints into a tree-structured network with additional nodes representing the connection and joints between subparts. Zhang and Kuo [23] and Kuo [24] proposed a component-fastener graph to represent the assembly relationship to find the most profitable dismantle sequence for returned products. Homem de Mello and Sanderson [25, 26] introduced an AND/OR graph representation of assembly plan which includes all possible sequences of operations. Lambert [27] utilized the AND/OR graph to generate sequences in retrieving subassemblies with the consideration of disassembly line balancing.

The objective of this research is to find an optimal partial disassembly sequence to retrieve reusable subassemblies or subparts from EoL products in single planning period or referred to as single period partial disassembly optimization (SPPDO) problem herein. A label correcting algorithm is proposed to solve the AND/OR graph as a shortest path problem. Furthermore, a heuristic procedure is developed to utilize this label correcting algorithm in solving the SPPDO problem. The remainder of the paper is organized as follows. In the next section, the representations of disassembly sequences and AND/OR graph are briefly discussed. The subsequent sections further introduce the mathematical formulation of the problem, the label correcting algorithm, and a heuristic procedure for solving SPPDO. This is followed by a section that illustrates how the heuristic algorithm works with numerical examples. Then, some concluding remarks are summarized in the last section.

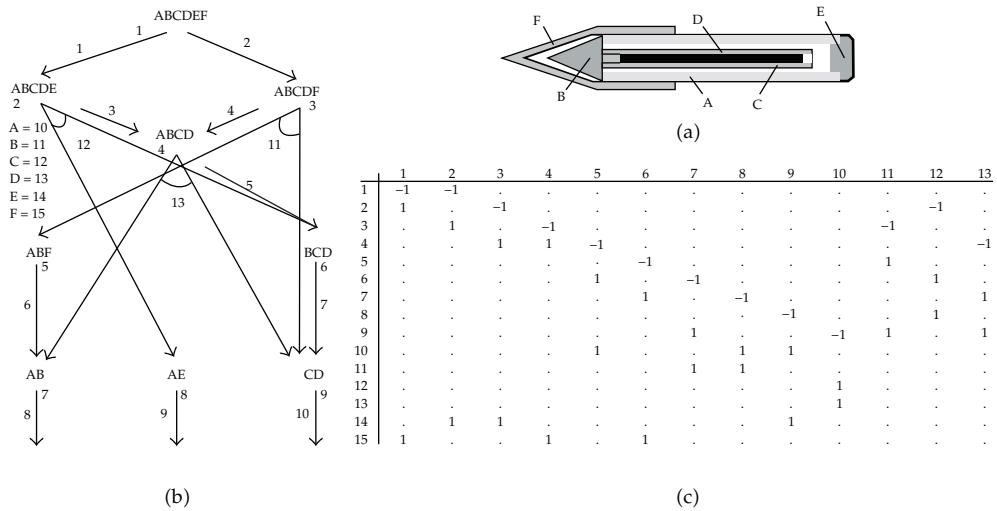
## 3. Formulation of Partial Disassembly Optimization Problem

### 3.1. Disassembly Sequence and AND/OR Graph

For a given return, the feasible disassembly sequence is restricted to the design of that product. After specifying the structure of a product, it is prerequisite to develop the relations



**Figure 2:** (a) The structure and (b) component-fastener graph for a partially disassembled PC [24].



**Figure 3:** (a) The structure, (b) AND/OR disassembly tree, and (c) transition matrix for a ballpoint [21].

between subassemblies and parts where separation operations are possible. A component-fastener graph can be used to represent the assembly relationship [23, 24]. As shown in Figure 2, if two components are attached or joined by fasteners, then these two components are connected by an undirected edge in the component-fastener graph.

A disassembly AND/OR graph is another useful representation for all possible disassembly sequences. An AND/OR graph is a directed hypergraph where subassemblies are represented as nodes. If more than one subassembly/part can be separated from a parent assembly in one disassembly operation, a hyperarc (AND arc) is used to indicate this operation and connects the parent node to all child nodes. Otherwise, a directed arcs (OR arcs) are used. Figure 3(a) illustrates the drawing of a ballpoint product with associated AND/OR graph in Figure 3(b). The assembly {ABCDE} yields two subassemblies {BCD} and {AE} through the disassembly operation {12}, and this operation is represented as an AND arc (“ $\cup$ ” arc). Moreover, this hypergraph is compact which requires a reduced number of nodes/arcs to enumerate all partial sequences of disassembly operations [25].

This AND/OR graph can also be represented completely via a transition matrix  $T$ . Suppose  $I$  be the set of subassemblies/subparts and  $J$  be the set of operations, the element  $T_{ij}$  has a value of 1 if a subassembly  $i \in I$  is released by some operation  $j \in J$  or -1 if subassembly

$i \in I$  is to be dismantled via operation  $j \in J$ . Figure 3(c) defines the associated transition matrix for the ballpoint product in Figure 3(a), where  $I = \{1 \dots 15\}$  and  $J = \{1 \dots 13\}$ . For example, the operation  $j = 12$  will dismantle one subassembly  $i = 2$  or assembly {ABCDE}, for two corresponding subassemblies  $i = 6$  and  $i = 8$  or assemblies {BCD}, and {AE}, respectively.

### 3.2. Mathematical Model

The partial disassembly optimization problem considered in this research is defined as a single-period partial disassembly optimization (SPPDO) problem. The problem assumed that the disassembly decision is to consider myopic optimal sequences given the quantity of returned products and the demands for reusable subparts in the period. For the product with  $I$  as the set of feasible subassemblies/subparts and  $J$  as the set of disassembly operations, two more sets are defined as follows:  $I_0$  as the set of original products,  $I_0 \subseteq I$ ;  $I_f$  as set of reusable subparts or target subparts,  $I_f \subseteq I$ .

Consider the AND/OR graph  $G$  for this returned product with  $I$  nodes and  $J$  arcs and the transition matrix  $T$  for the product. A feasible disassembly sequence to retrieve reusable parts from a returned product can be defined as a path from a source node in  $I_0$  to the corresponding target nodes in  $I_f$ . Let  $c_j$  be the cost of operation  $j$ , and  $c_j \geq 0$ , for all  $j \in J$ . In this model, the available quantity is defined as a negative number  $b_i \leq 0$ , for original returns  $i$ , for all  $i \in I_0$ . For each reusable subpart  $i$ , the demand is defined as  $b_i \geq 0$ , for all  $i \in I_f$ . Intermediate subassembly has zero demand; that is,  $b_i = 0$ , for all  $i \in I \setminus (I_0 \cup I_f)$ . Without loss of generality, we assume that this model has enough supply of original returns; that is,  $\sum_{i \in I} b_i \leq 0$ . Suppose  $y_j$  be the number of operation  $j$ ,  $j \in J$ , needed in this planning period, then an optimal partial disassembly sequence with minimum total disassembly cost can be obtained by solving the formulation proposed as follows:

SPPDO

$$\text{Minimize} \quad \sum_{j \in J} c_j \cdot y_j, \quad (3.1)$$

$$\text{Subject to} \quad \sum_{j \in J} T_{ij} \cdot y_j \geq b_i, \quad \forall i \in I_0 \cup I_f, \quad (3.2)$$

$$\sum_{j \in J} T_{ij} \cdot y_j \geq 0, \quad \forall i \in I \setminus (I_0 \cup I_f), \quad (3.3)$$

$$y_j \geq 0 \text{ and integer } \forall j \in J. \quad (3.4)$$

This SPPDO model is a generalized minimum cost flow problem where the arc in a graph include both hyperarcs (AND arcs) and regular directed arcs (OR arcs). The constraint set (3.2) is to ensure that the demands of reusable subparts are fulfilled or the required quantities for original returns are still sufficient. The constraint sets (3.3) and (3.4) are nonnegative constraints on the resulting quantity of intermediate subassemblies/parts and the number of operations needed. It is worth noting that this AND/OR graph is acyclic, and the sum of degrees from all nodes might not be zero. Further note that, since the transition matrix  $T$  does

not have the property of total unimodularity, this SPPDO formulation can only be solved as a pure integer programming (IP) problem.

In the literature, a searching algorithm for an AND/OR graph with different interpretation is available, but it is not applicable to solve the SPPDO problem directly. AND/OR graph is often used to represent a problem-solving process which transforms the original problem into several subproblems [28, 29]. Each node represents a distinct problem. The node represents the original problem is referred to as the starting node or root node. A terminal node, or leaf node, in this graph represents a problem whose solution is either known to exist or not to exist. A directed arc is linked from a node (problem) to its associated successive nodes (subproblems). For an OR arc, the problem is solved when the immediately successive subproblem is solved. If the problem is linked by an AND arc, the problem is only solved when all the successive subproblems are solved. Hence, the solution for the original problem is to search for a tree that connects the root node with terminal nodes only [28].

Zhang and Kuo [23] had extended this searching algorithm in finding a solution tree from an AND/OR graph to obtain the optimal assembly sequence toward the final product. Even with the assumption of reversible operation, the sequence obtained from the solution tree might not be directly used for generating disassembly sequence if partial disassembly is allowed. The main reason is that the disassembly level needs to be identified if full disassembly is not desirable. The selection in a proper set of terminal nodes can be combinatorial [25]. Considering the example as in Figures 4(a) and 4(b), the searching algorithm can be used to find the best trees for the assembly operations from leaf nodes to the root node as in Figure 5. But to retrieve a stick from the returned problem, the optimal sequences for partial disassembly operations ( $\{1\}, \{8\}$ ) or ( $\{4\}, \{7\}$ ) can only be obtained if and only if the set of terminal nodes selected are  $\{7, 9, 10\}$  or  $\{6, 10, 12\}$ , respectively.

### 3.3. Label Correcting Algorithm

Here we propose a label correcting algorithm to find an optimal disassembly sequence from an AND/OR graph. This algorithm maintains a label  $L_i = [d_i, p_i]$  for each node  $i$ ,  $i \in I$ , where  $d_i$  is the minimum disassembly cost to retrieve node  $i$  from some starting node in  $I_s$ , and  $p_i$  is the set of immediately predecessor nodes in the shortest path. Let  $f(j)$  and  $t(j)$  denote the from-node and to-node of some arc  $j$ ,  $j \in J$ . Further,  $F_i = \{j \in J : f(j) = i\}$  and  $R_i = \{j \in J : t(j) = i\}$  define the forward star and reverse star for each node  $i$ ,  $i \in I$ . The detailed steps of this algorithm are as follows.

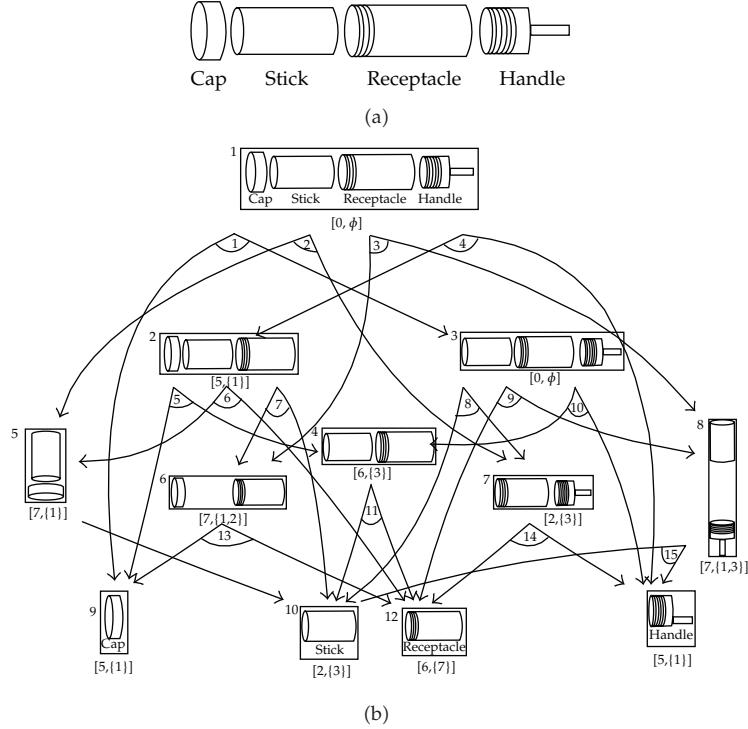
#### *Initialization*

For each source node  $k$  in  $k \in I_s$ , set the minimum cost  $d_k = 0$ , the predecessor set  $p_k = \emptyset$ , and update the set of labelled node  $L = L \cup \{k\}$ . For node  $k$  in  $I \setminus I_s$ , set the minimum cost  $d_k = \infty$ , the predecessor set  $p_k = \emptyset$ . Select the first labelled node  $k$  in  $L$ .

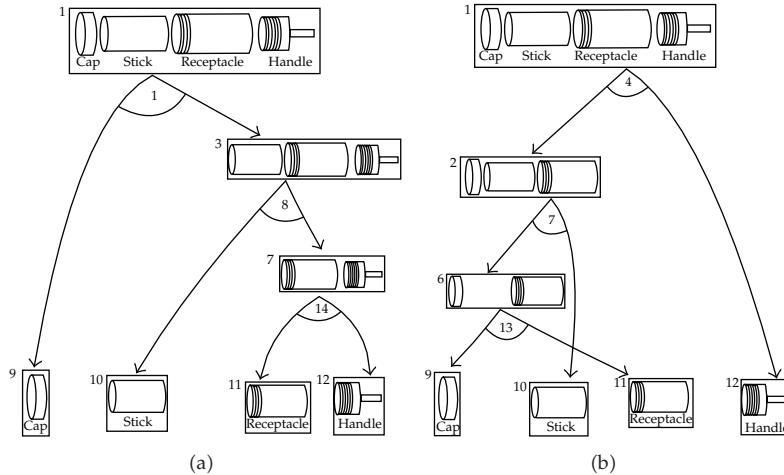
*Step 1.* Determine the set  $S_k$  for unlabelled nodes that are immediately connected from node  $k$ ; that is,  $S_k = \{n \mid n \in t(j), j \in F_k\} \setminus L$ . If  $S_k$  is not empty, then go to Step 2. Otherwise, go to Step 3.

*Step 2.* For each node  $i$  in  $S_k$ , do the following

- (1) For each arc  $j$  where  $j \in F_k \cap R_i$ , if  $d_k + c_j < d_i$ , update the minimum cost  $d_i = d_k + c_j$  and set  $p_i = \{k\}$ . Otherwise, if  $d_k + c_j = d_i$ , then update  $p_i = p_i \cup \{k\}$ .



**Figure 4:** (a) The structure and (b) the AND/OR disassembly tree for a simple product [25].



**Figure 5:** Alternated trees of disassembly sequence to remove the stick for the simple product in (a) [25].

- (2) Determine the set  $U_i$  for unlabelled nodes that are immediately connected toward node  $i$ ; that is,  $U_i = \{n | n \in f(j), j \in R_i\} \setminus L$ . If  $U_i$  is empty, make node  $i$  as labelled and update the set of labelled nodes  $L = L \cup \{i\}$ .

*Step 3.* Mark the current node  $k$  as solved. Move to the next unsolved node in  $L$  as new node  $k$ , go to Step 1. If all nodes are solved, stop.

If only original returns can be dismantle for reusable subparts, we have  $I_s = I_0$ . For any solved node  $i$ , the  $d_i$  from the label  $L_i = [d_i, p_i]$  represents the minimum disassembly cost to retrieve one subpart  $i$  directly from a starting assembly. Moreover, the complexity of this procedure is polynomial bound by the number of nodes; that is,  $O((|I|)^2)$  [30].

**Lemma 3.1.** *If only one target subpart has positive demand, the label correcting algorithm solves the SPPDO problem.*

*Proof.* In the SPPDO formulation, it is assumed that target subparts are retrieved from returned products directly. If only one target subpart  $i$  has positive demand, it is equivalent to find an optimal sequence of disassembly operations with the minimum total cost to reach the destination node  $i$  from the source node. This minimum cost is also defined as  $d_i$  in the label  $L_i = [d_i, p_i]$  for node  $i$ ,  $i \in I_f$ .  $\square$

**Lemma 3.2.** *If more than one target subpart has positive demands, then the corresponding  $d_i$  from the label correcting algorithm for associated nodes forms an upper bound for the optimal solution in the SPPDO problem. That is, suppose  $y_j^*, j \in J$ , be the optimal solution for the SPPDO problem, then one has  $\sum_{j \in J} c_j \cdot y_j^* \leq \sum_{i \in I_f} b_i \cdot d_i$ .*

*Proof.* Suppose two target subparts, say  $l$  and  $m$ ,  $l \neq m$ , have positive demands. Let  $P_l = \{j_{l1}, j_{l2}, \dots, j_{lp}\}$  and  $P_m = \{j_{m1}, j_{m2}, \dots, j_{mp}\}$  be the optimal disassembly sequences for node  $l$  and node  $m$ , respectively. Without loss of generality, we assume that  $b_l \geq b_m > 0$ . Suppose there exists some node  $k \in t_j$ ,  $j \in P_l$  and also  $k \in f_j$ ,  $j \in P_m$ . Then the shortest disassembly path from the source node to node  $k$  is overlapped in the optimal disassembly sequences from the source node to node  $l$  and node  $m$ . Hence, an upper bound for the optimal disassembly cost is  $\sum_{j \in J} c_j \cdot y_j^* \leq (b_l \cdot d_l + b_m \cdot d_m) - b_m \cdot d_k$ .  $\square$

Moreover, it can be concluded that  $\sum_{j \in J} c_j \cdot y_j^* = \sum_{i \in I_f} b_i \cdot d_i$  if and only if each original return can be retrieved for no more than one target subpart only.

### 3.4. A Heuristic Procedure for Solving Partial Disassembly Optimization Problem

Next, we presented heuristic procedure that utilizes the label correcting algorithm in the previous section. This procedure is to find a good solution for larger instances of SPPDO problem in a real-world setting within a reasonable computation effort. The detail of this iterative procedure is described as follows.

#### Initialization

Apply the label correcting algorithm with  $I_s = I_0$  to obtain the initial labels  $L_i = [d_i, p_i]$  for all subparts  $i$ ,  $i \in I$ . Set the variable  $y_j = 0$ ,  $j \in J$ . Let  $x_i$  be the quantity of subassembly  $i$  available for further dismantle, set  $x_i = 0$ ,  $i \in I$ , and  $x_i = |b_i|$ ,  $i \in I_0$ . Note that  $\sum_{i \in I_0} |b_i| \geq \sum_{i \in I_f} b_i$ .

*Phase 1* (Path construction). Select a target node  $i$  which has the maximum total potential cost in unfulfilled demand; that is,  $i = \arg_{k \in I_f} \max \{(b_k - x_k) \cdot d_k \mid b_k > x_k\}$ . Break ties arbitrarily.

Obtain the minimum cost disassembly sequences,  $P_i = \{j_{i1}, j_{i2}, \dots, j_{ip}\}$ , which forms a directed path from source node  $s$ ,  $s = f(j_{i1})$  toward the demand node  $i$ ;  $i \in t(j_{ip})$ . Break ties arbitrarily. Find the maximum flow for this path,  $\Delta = \min\{x_s, (b_i - x_i)\}$ .

For each arc  $j$ ,  $j \in P_i$ , start from the first arc, perform the following updates sequentially.

$$\begin{aligned} y_j &= y_j + \Delta, \\ x_k &= x_k - \Delta, \quad \text{if } k \in f(j), \\ x_k &= x_k + \Delta, \quad \text{if } k \in t(j). \end{aligned} \tag{3.5}$$

*Phase 2 (Termination test).* Update the total disassembly cost  $z = z + \Delta \cdot d_i$  and check for any unfulfilled demand; that is,  $x_k < b_k$ ,  $k \in I_f$ . If all demands are fulfilled, then stop. The current solution of  $y_j$ ,  $j \in J$ , is feasible for the SPPDO problem.

Otherwise, update the set of source nodes to include intermediate nodes with positive quantity; that is,  $I_s = I_0 \cup \{k\}$ , if  $x_k > 0$  for  $k \in I \setminus \{I_0 \cup I_f\}$ . Update labels  $L_i = [d_i, p_i]$  for all subparts  $i$ ,  $i \in I$  using the label correcting algorithm in Section 3.3 with the new set of source nodes  $I_s$ . Go to Phase 1.

**Lemma 3.3.** *The objective value of solution obtained from this heuristic procedure, referred to as  $z_H$ , is at least as good as that obtained from the label correcting algorithm, referred to as  $z_{LC}$ .*

*Proof.* In Phase 2, the labels for subparts are altered only when there exists some intermediate subassembly with positive quantity to be further dismantled for those target subparts with lower costs. Otherwise, the initial labels remain unchanged. So, the contribution of disassembly cost for a target subpart in  $z_H$  is no higher than that in  $z_{LC}$ .  $\square$

Furthermore, let  $z^* = \sum_{j \in J} c_j \cdot y_j$  be the optimal solution for the SPPDO problem and  $z_{LC} = \sum_{i \in I_f} b_i \cdot d_i$ . We have  $z^* \leq z_H \leq z_{LC}$  from Lemmas 3.2 and 3.3.

#### 4. Numerical Examples

In this section, the simple product in Figure 4(a) is used to demonstrate how the heuristic procedure in Section 3.4 works to generate partial disassembly sequences for the SPPDO problem. There are totally twelve different (sub)assemblies,  $I = \{1, \dots, 12\}$ . The original return is represented by node {1}, and reusable subparts are nodes {9}, {10}, {11}, {12} for cap, stick, receptacle, and handle, respectively, that is,  $I_0 = \{1\}$  and  $I_f = \{9, 10, 11, 12\}$ . A total of fifteen disassembly operations can be used to dismantle this product,  $J = \{1, \dots, 15\}$  with the associated costs  $C = \{5, 7, 7, 5, 6, 7, 2, 2, 7, 6, 2, 1, 4, 4, 1\}$ .

The construction of labels using the label correcting algorithm is shown in Figure 6. The minimum disassembly cost to retrieve node {10} (stick) is 7 with two alternative optimal disassembly sequences:  $P_{10} = \{4, 7\}$  or  $P_{10} = \{1, 8\}$ . This optimal solution is consistent with observation in Section 3.2.

Next, we demonstrate the use of heuristic algorithm with the consideration of the following demands for target subparts: three caps (node {9}) and one stick (node {11}); that is,  $b_9 = 3$ ,  $b_{11} = 1$ . It implies that the supply of original returns should be at least four; that is,  $b_1 = -4$ . In the initialization step, all variables  $x, y$  are set to zero except  $x_1 = 4$  and the initial labels are obtained from Figure 6. In Phase 1, since node {9} (cap) has a higher total unfulfilled cost than node {11} (stick), node {9} is selected along with the associated directed

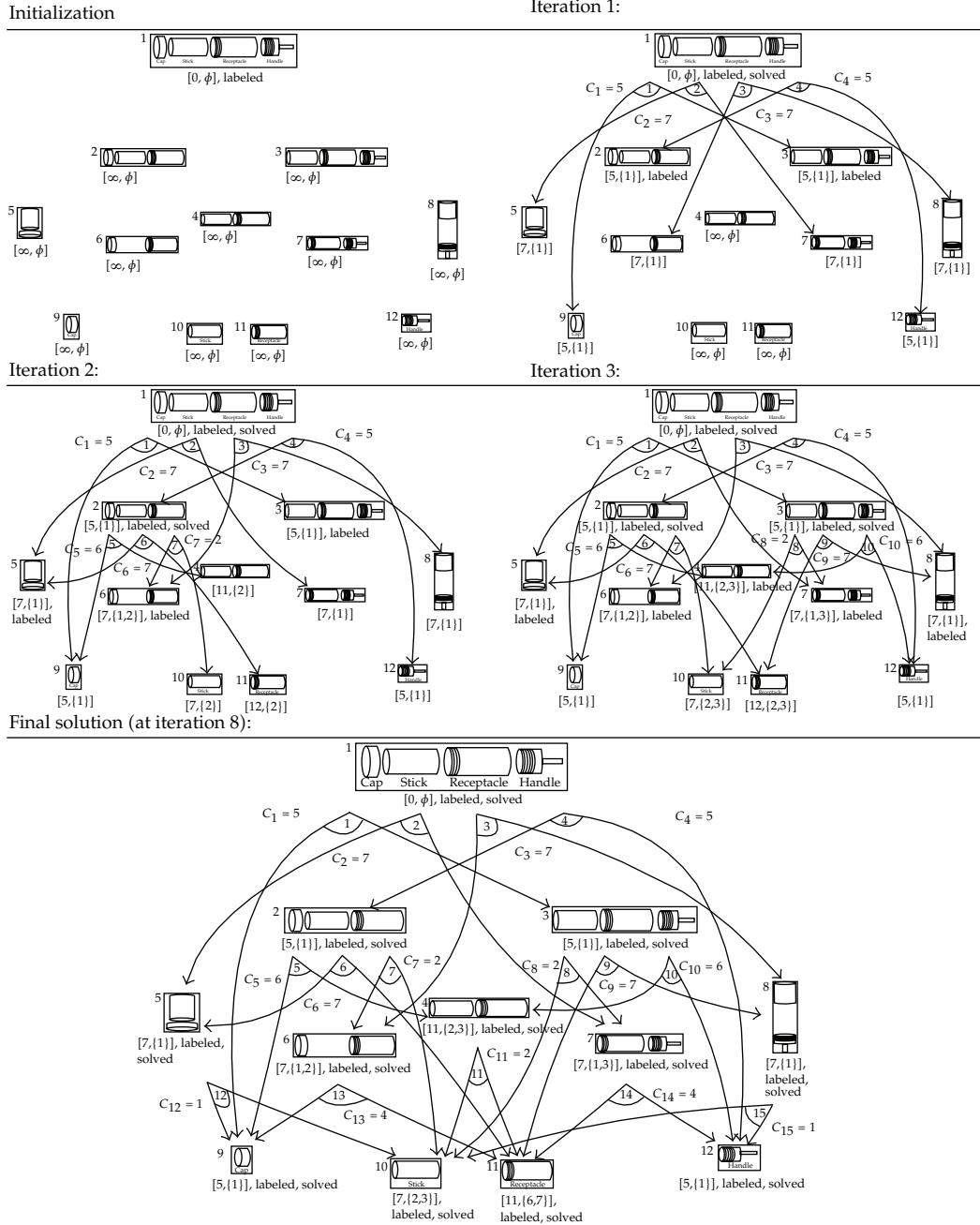
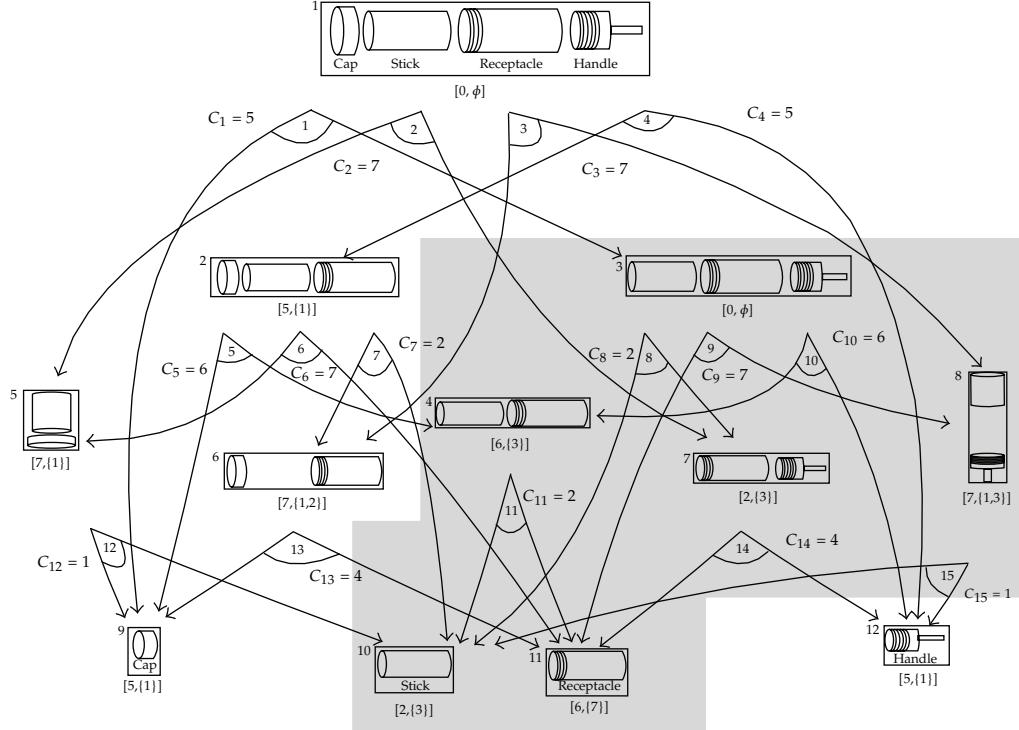


Figure 6: Brief illustration of stepwise label correcting algorithm for the product structure as in Figure 5(a).

path  $P_9 = \{1\}$ . Calculate the flow  $\Delta = \min\{x_1, (b_9 - x_9)\} = \min\{4, (3 - 0)\} = 3$  on this path. Since  $P_9$  has only one arc  $j = 1$ , update the solution  $y_1 = 3$ ,  $x_1 = 4 - 3 = 1$ ,  $x_3 = 3$ , and  $x_9 = 3$ .

Continue to Phase 2 to check for the termination criteria. Update the current total disassembly cost  $z = z + \Delta \cdot d_9 = 0 + 3 * 5 = 15$ . There still exists unfulfilled demand since  $x_{11} = 0 < b_{11}$ . So, first update the set of source nodes  $I_s = \{1, 3\}$  to include the intermediate node  $\{3\}$  as  $x_3 = 3 > 0$ . Then labels for subparts are updated using the label correcting



**Figure 7:** Updated labels with an additional source node  $\{3\}$  as in the shaded area.

algorithm. It is worth noting that, as shown in Figure 7, not every label is changed, and the effected nodes with associated arcs are shown in bold in the shaded area.

In the second iteration, select node  $\{11\}$  to fulfil the remaining demand  $b_{11} = 1$  with the disassembly path  $P_{11} = \{8, 14\}$ . Calculate the maximum flow  $\Delta = \min\{x_3, (b_{11} - x_{11})\} = \min\{3, (1-0)\} = 1$  and update the solution starting from the first arc  $j = 8$ :  $y_8 = 1$ ,  $x_3 = 3-1 = 2$ ,  $x_7 = 1$ , and  $x_{10} = 1$ . Then update for the second arc  $j = 14$ :  $y_{14} = 1$ ,  $x_7 = 1-1 = 0$ ,  $x_{11} = 1$ , and  $x_{12} = 1$ . This procedure terminates since all demands are fulfilled with the total cost  $z = 15 + 1 * 6 = 21$ . The resulting quantities of subparts are  $x_1 = 1$ ,  $x_3 = 2$ ,  $x_9 = 3$ ,  $x_{11} = 1$ , and  $x_{12} = 1$ , and the required disassembly operations are  $y_1 = 3$ ,  $y_8 = 1$  and  $y_{14} = 1$  for three caps and one stick.

## 5. Conclusions

In this paper, we investigate a single period partial disassembly optimization (SPPDO) problem to generate an optimal disassembly sequence in product recovery of the end-of-life (EoL) products. An AND/OR graph representation and associated transition matrix are used in the mathematical formulation of the SPPDO problem to minimize the total disassembly cost. Since the transition matrix does not have the property of total unimodularity, this SPPDO model can only be solved as a pure integer programming (IP) problem, which is NPcomplete.

A label correcting algorithm is proposed to find an optimal disassembly sequence when the reusable subpart is retrieved directly from original return. To solve the SPPDO

problem in general case, this paper presents a heuristic procedure that utilizes this polynomial-time algorithm to find a solution. This heuristic procedure can quickly provide a good disassembly plan for problems with more complicated disassembly structures in a real-world setting within a reasonable computation effort. It can be further integrated in the production planning for end-of-life (EoL) products to improve the profitability of product recovery.

## Acknowledgment

This research was supported by the National Science Council of Taiwan, China, under Grant no. NSC 98-2218-E-027-019.

## References

- [1] H. Griese, H. Poetter, K. Schischke, O. Ness, and H. Reichl, "Reuse and lifetime extension strategies in the context of technology innovations, global markets, and environmental legislation," in *Proceedings of the IEEE International Symposium on Electronics and the Environment*, pp. 173–178, Washington, DC, USA, May 2004.
- [2] E. J. Grenchus, R. A. Keene, and C. R. Nobs, "A decade of processing end of life IT equipment—lessons learned at the IBM endicott asset recovery center," in *Proceedings of the IEEE International Symposium on Electronics and the Environment*, pp. 195–198, Washington, DC, USA, May 2004.
- [3] M. Fleischmann, *Quantitative Models for Reverse Logistics*, Springer, New York, NY, USA, 2001.
- [4] E. Grenchus, S. Johnson, and D. McDonnell, "Improving environmental performance through reverse logistics at IBM," in *Proceedings of the IEEE International Symposium on Electronics and the Environment*, pp. 236–240, May 2001.
- [5] V. D. R. Guide Jr., "Production planning and control for remanufacturing: industry practice and research needs," *Journal of Operations Management*, vol. 18, no. 4, pp. 467–483, 2000.
- [6] A. Kasmara, M. Muraki, S. Matsuoka, A. Sukoyo, and K. Suryadi, "Production planning in remanufacturing/ manufacturing production system," in *Proceedings of the 2nd International Symposium on Environmentally Conscious Design and Inverse Manufacturing*, pp. 708–713, 2001.
- [7] A. J. Clegg, D. J. Williams, and R. Uzsoy, "Production planning for companies with remanufacturing capability," in *Proceedings of the IEEE International Symposium on Electronics and the Environment (ISEE '95)*, pp. 186–191, May 1995.
- [8] V. D. R. Guide Jr., "Scheduling using drum-buffer-rope in a remanufacturing environment," *International Journal of Production Research*, vol. 34, no. 4, pp. 1081–1091, 1996.
- [9] V. D. R. Guide Jr. and M. S. Spencer, "Rough-cut capacity planning for remanufacturing firms," *Production Planning and Control*, vol. 8, no. 3, pp. 237–244, 1997.
- [10] V. Daniel, V. D. R. Guide Jr., and R. Srivastava, "An evaluation of order release strategies in a remanufacturing environment," *Computers and Operations Research*, vol. 24, no. 1, pp. 37–47, 1997.
- [11] V. D. R. Guide Jr., R. Srivastava, and M. E. Kraus, "Product structure complexity and scheduling of operations in recoverable manufacturing," *International Journal of Production Research*, vol. 35, no. 11, pp. 3179–3199, 1997.
- [12] V. D. R. Guide Jr., M. E. Kraus, and R. Srivastava, "Scheduling policies for remanufacturing," *International Journal of Production Economics*, vol. 48, no. 2, pp. 187–204, 1997.
- [13] M. E. Ketzenberg, G. C. Souza, and V. D. R. Guide Jr., "Mixed assembly and disassembly operations for remanufacturing," *Production and Operations Management*, vol. 12, no. 3, pp. 320–335, 2003.
- [14] P. Dewhurst, "Product design for manufacture: design for disassembly," *Industrial Engineering*, vol. 25, no. 9, pp. 26–28, 1993.
- [15] K. N. Taleb and S. M. Gupta, "Disassembly of multiple product structures," *Computers and Industrial Engineering*, vol. 32, no. 4, pp. 949–961, 1997.
- [16] I. M. Langella, "Heuristics for demand-driven disassembly planning," *Computers and Operations Research*, vol. 34, no. 2, pp. 552–577, 2007.
- [17] A. Gungor and S. M. Gupta, "An evaluation methodology for disassembly processes," *Computers and Industrial Engineering*, vol. 33, no. 1-2, pp. 329–332, 1997.
- [18] D. Navin-Chandra, "The recovery problem in product design," *Journal of Engineering Design*, vol. 5, no. 1, pp. 65–86, 1994.

- [19] S. C. Sarin, H. D. Sherali, and A. Bhootra, "A precedence-constrained asymmetric traveling salesman model for disassembly optimization," *IIE Transactions*, vol. 38, no. 3, pp. 223–237, 2006.
- [20] M. R. Johnson and M. H. Wang, "Economical evaluation of disassembly operations for recycling, remanufacturing and reuse," *International Journal of Production Research*, vol. 36, no. 12, pp. 3227–3252, 1998.
- [21] A. J. D. Lambert, "Optimizing disassembly processes subjected to sequence-dependent cost," *Computers and Operations Research*, vol. 34, no. 2, pp. 536–551, 2007.
- [22] T. L. De Fazio and D. E. Whitney, "Simplified generation of all mechanical assembly sequences," *IEEE Journal of Robotics and Automation*, vol. 3, no. 6, pp. 640–658, 1987.
- [23] H. C. Zhang and T. C. Kuo, "Graph-based approach to disassembly model for end-of-life product recycling," in *Proceedings of the IEEE/CPMT 19th International Electronics Manufacturing Technology Symposium*, pp. 247–254, October 1996.
- [24] T. C. Kuo, "Disassembly sequence and cost analysis for electromechanical products," *Robotics and Computer-Integrated Manufacturing*, vol. 16, no. 1, pp. 43–54, 2000.
- [25] L. S. Homem de Mello and A. C. Sanderson, "AND/OR graph representation of assembly plans," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 2, pp. 188–199, 1990.
- [26] L. S. Homem de Mello and A. C. Sanderson, "A correct and complete algorithm for the generation of mechanical assembly sequences," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 2, pp. 228–240, 1991.
- [27] A. J. D. Lambert, "Exact methods in optimum disassembly sequence search for problems subject to sequence dependent costs," *Omega*, vol. 34, no. 6, pp. 538–549, 2006.
- [28] C. L. Chang and J. R. Slagle, "An admissible and optimal algorithm for searching AND/OR graphs," *Artificial Intelligence*, vol. 2, no. 2, pp. 117–128, 1971.
- [29] G. Levi and F. Sirovich, "Generalized and/or graphs," *Artificial Intelligence*, vol. 7, no. 3, pp. 243–259, 1976.
- [30] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.

## Research Article

# Improved Degree Search Algorithms in Unstructured P2P Networks

**Guole Liu,<sup>1,2,3</sup> Haipeng Peng,<sup>1,2,3</sup> Lixiang Li,<sup>1,2,3</sup>  
Yixian Yang,<sup>1,2,3</sup> and Qun Luo<sup>1,2,3</sup>**

<sup>1</sup> *Information Security Center, Beijing University of Posts and Telecommunications,  
P.O. Box 145, Beijing 100876, China*

<sup>2</sup> *Research Center on Fictitious Economy and Data Science, Chinese Academy of Sciences,  
Beijing 100190, China*

<sup>3</sup> *National Engineering Laboratory for Disaster Backup and Recovery,  
Beijing University of Posts and Telecommunications, Beijing 100876, China*

Correspondence should be addressed to Haipeng Peng, penghaipeng2003@yahoo.com.cn

Received 1 November 2011; Revised 21 February 2012; Accepted 14 April 2012

Academic Editor: Yi-Chung Hu

Copyright © 2012 Guole 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.

Searching and retrieving the demanded correct information is one important problem in networks; especially, designing an efficient search algorithm is a key challenge in unstructured peer-to-peer (P2P) networks. Breadth-first search (BFS) and depth-first search (DFS) are the current two typical search methods. BFS-based algorithms show the perfect performance in the aspect of search success rate of network resources, while bringing the huge search messages. On the contrary, DFS-based algorithms reduce the search message quantity and also cause the dropping of search success ratio. To address the problem that only one of performances is excellent, we propose two memory function degree search algorithms: memory function maximum degree algorithm (MD) and memory function preference degree algorithm (PD). We study their performance including the search success rate and the search message quantity in different networks, which are scale-free networks, random graph networks, and small-world networks. Simulations show that the two performances are both excellent at the same time, and the performances are improved at least 10 times.

## 1. Introduction

Searching and retrieving the demanded correct information is becoming more and more important with the emergence of the huge amounts of information and the growth in the size of computer networks [1]. Especially, in unstructured P2P networks, the node's joining and failure are both random and dynamic [2], and in this case, it is unfeasible and unpractical that each node of the network has known and stored the global information about the whole

network topology and the location of queried resources. Thus, designing efficient search algorithms according to the local network information is critical to the performance of unstructured P2P networks.

Considerable amount of work has been done in this field, so far, a number of search algorithms have been proposed, including BFS algorithm [1], modified BFS algorithm [1, 3, 4], local search algorithm [5, 6], rumor broadcasting algorithm [7–10], the betweenness [11, 12], shortest path algorithm [13], iterative deepening algorithm [1, 14, 15], update propagation algorithm [16], and random walks search [17–30]. These search algorithms can be classified into two categories: BFS-based method and DFS-based method. Although these algorithms achieve relatively satisfying effects, these two types of search algorithms tend to be inefficient, either generating too much load on the networks [1, 2] or not meeting the search success rate of network resources. On the one hand, BFS-based algorithm shows the perfect performance in the aspect of search success rate of network resources, but at the same time, it brings the huge search messages. The number of search messages will grow exponentially with the hop counts in the search process [17, 18]. On the other hand, DFS-based algorithm generates the search message quantity far smaller compared with BFS-based algorithm, and the search message quantity will grow linearly with the hop counts [20]. The main drawback is to drop the search success ratio of network resources.

The references [3, 4, 7–10, 17–30] adopt the BFS-based method or DFS-based method to achieve their goal, respectively, but do not overcome the problem that only one of performances of the network loads, and the search success rate is excellent. To address this problem, in this paper, according to the degree of nodes [2], we propose memory function maximum degree algorithm (MD) and memory function preference degree algorithm (PD). These two algorithms can combine the advantages of the BFS-based algorithm and DFS-based algorithm, which can be efficiently used to search random graph [31, 32] networks and power-law networks such as scale-free networks and small-world networks [33, 34]. We have studied their performances in the search success rate of network resources and the search message quantity. Simulations illustrate their validity and feasibility. The results show that MD algorithm is better than PD algorithm in the search success rate. The search success rate of MD algorithm is average 14 times better than the standard random walks algorithm; the search message quantity is the same order of magnitude with it. Compared with modified BFS algorithm, the search success rate of MD algorithm is higher than it, and the search message quantity averagely reduces by over 18 times. Although PD algorithm can reduce the huge search message quantity, the search success rate of it is inferior to the modified BFS algorithm.

## 2. The Improved Degree Algorithms Methods

The degree of a node in a network (sometimes referred to as the connectivity of a graph) is the number of connections or edges the node has to other nodes. The degree of a node is an important index for some problems, which is used to measure the importance of the node. For the aspect of information transmission speed, the more edges connected to the node, the faster information dissemination by the node. Namely, the node is more important. For the aspect of the shortest path viewpoint, the greater betweenness centrality, the more importance of the node. Meanwhile, the degree of the node may be very small. Considering these, adopting the idea of assigning unique ID to each node in unstructured P2P networks, we, respectively, propose the memory function maximum degree algorithm (MD) and the memory function preference degree algorithm (PD). In the context, “Memory function” has two aspects contents. The first is that one node needs to store its neighbors’ ID and degree

information. The second is that one node has to remember the return node's ID according to its memory information. This requires the nodes of networks to save their neighbors' ID and, at the same time, save their related degree information. It is the advantage of these two algorithms from the point of view reducing the unnecessary search messages, and it is the shortcoming from the point of view occupying the storage space. Compared with the MB algorithm and the random walks algorithms, they double the storage space. In the context, the degree of a node is the number of connections or edges the node has to other nodes, including the traversed nodes.

## 2.1. The Memory Function Maximum Degree Algorithm

In this strategy, when a node starts the resource search procedure, it firstly traverses all its adjacent nodes according to the BFS method to determine whether they contain the resources or not. If all the neighbors do not contain the resources, it changes the flag of its neighbors to denote that these nodes have been traversed, then broadcasts the search messages along directions of nodes with the highest degree according to DFS method, and updates the flags to denote that message has passed these nodes. (When the procedure is over, all the flags recover zero.) If it does not find the resource along directions of nodes with the highest degree, the search message will return the precursor node and broadcast along its neighbor node with the second highest degree. This procedure will stop until the age counter is increased to threshold or all the nodes in the network have been traversed. In extreme cases, when the neighbor's degrees are all the same, the algorithm degenerates into the standard BFS algorithm. When all the nodes' degrees are different, the algorithm degenerates into the standard DFS algorithm.

In the context, assuming that the red node is the source node and the blue node is the resource node, the green nodes are the intermediate nodes passed by search messages in the process. The orange nodes are the labeled nodes that can be found, which do not need to send the search messages. The digits on the top of arrows are the age values. The solid line arrows represent the spread direction of messages, and the dotted line arrows denote the direction of response messages.

Figure 1 shows the search process that MD algorithm search the resource node located in the path composed by the highest-degree nodes. Figure 1(a) is the first step; unlike the maximum degree algorithm [11], node 1 firstly traverses all its neighbors and ascertains whether the neighbors have the resource or not. All its neighbors do not contain the resource, and node 1 changes the flags of its neighbors according to BFS method; then according to DFS method it broadcasts the search message to node 3 whose degree is the highest among the neighbors. Figures 1(b) and 1(c) repeat the search process. In Figure 1(d), the search message finds the resource node and responds to the required message.

This algorithm has two aspects of difference compared with the maximum degree algorithm. On the one hand, in maximum degree algorithm the search message broadcasts along the nodes with highest degree. Although, a node can send several messages to its neighbor's nodes at the same time (it has the neighbors with the same highest degree), the algorithm does not search resources according to BFS method in essence. Thus, the maximum degree algorithm can only find the resource nodes located in the path composed by the highest degree nodes. By querying neighbors first before sending out search messages, MD algorithm provides higher success rate. So the search success ratio of maximum degree algorithm is inferior to that of MD algorithm. Figure 2 shows the search process of the maximum degree algorithm and the difference compared with the MD algorithm. Figures 2(a)–2(d) are the search process of the maximum degree algorithm. Figure 2(a) is the first step; the degree

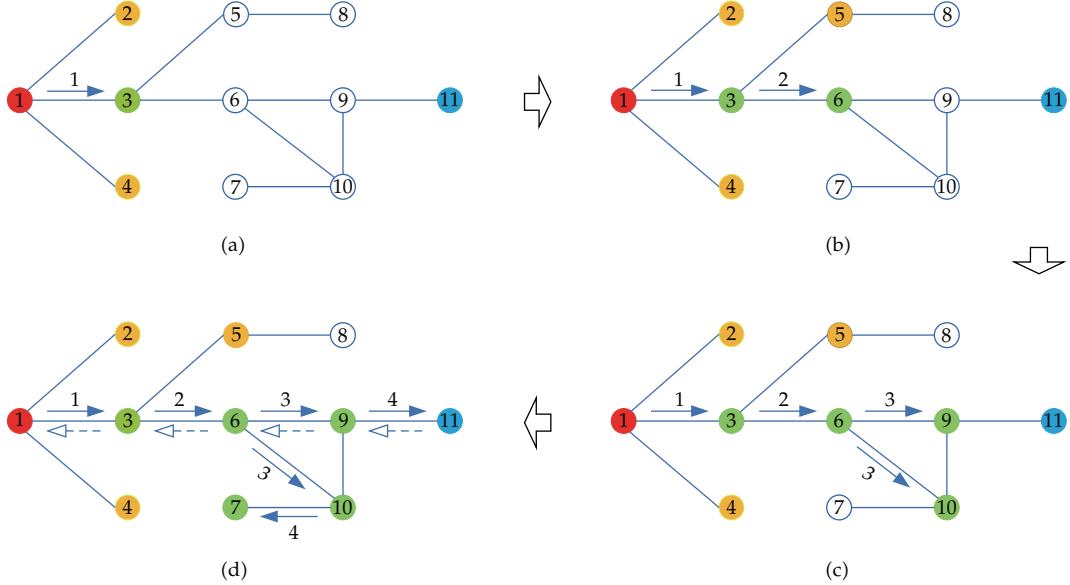


Figure 1: Search process of the MD algorithm.

of node 3 is the largest, so the search message is sent to node 3. Figure 2(b) is the second step; node 3 sends the search message to node 6. Figure 2(c) is the third step; the neighbor's nodes of node 6 have the same degree, and it sends two search messages to the neighbors. Figure 2(d) is the forth step; the algorithm returns failed. If the resource is located in the path composed by the highest-degree nodes, such as node 3, 6, 7, 9, 10, and 11, the maximum degree algorithm can easily find these nodes. Figures 2(e) and 2(f) are the difference with MD algorithm. Obviously, the MD algorithm easily finds the resource node. The maximum degree algorithm searches the resource in failure, while MD algorithm searches the same resource in success.

On the other hand, MD algorithm can search more resources according to the memory of the ID and degree information. Figure 3 shows how to search the resource node with memory function using MD algorithm, assuming that the age value is large enough. In Figure 3(a), the source node 1 traverses all its neighbors and labels the neighbors according to BFS method; then according to DFS method it broadcasts the search message to node 3 whose degree is the highest among the neighbors. Figures 3(b)–3(d) repeat the search process. In Figure 3(e), node 7 and node 11 are the terminal nodes; the search messages return node 9 and node 10 according to the memory information which are the ID information of their precursors. Node 9 and node 10 remember the returned node's ID, which the search messages do not broadcast along the direction of these nodes. In Figure 3(f), the neighbors of node 9 and node 10 are all traversed, so the search messages continue to return. In Figure 3(g), the search message returns to node 3. In Figure 3(h), the search message will broadcast along the node with the second highest degree, because it does not find the resource along the direction of nodes with the highest degree. In Figure 3(i), the search message finds the resource node, then node 8 responds to the required messages.

In summary, MD algorithm can search both categories resource nodes: highest degree nodes and nonhighest degree nodes. It labels the nodes with different flags and uses the nodes' ID to reduce messages and, at the same time, to improve the search success ratio.

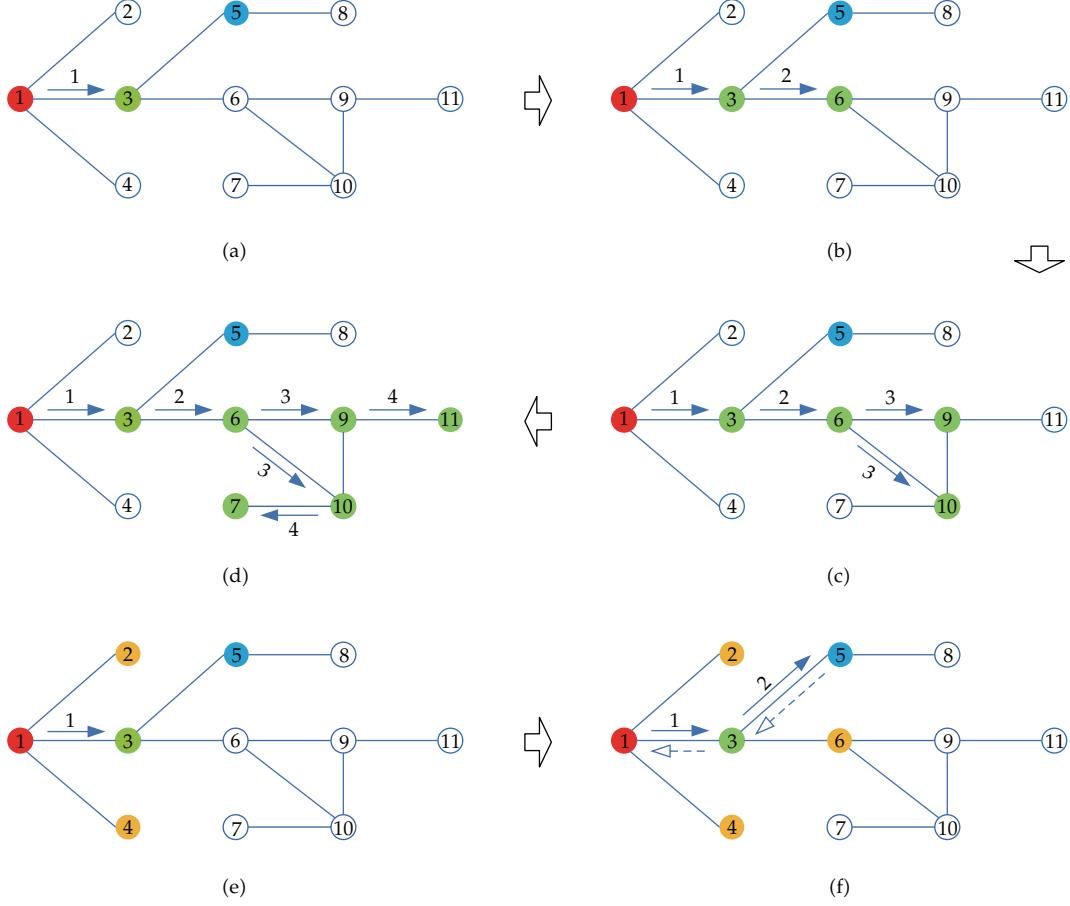


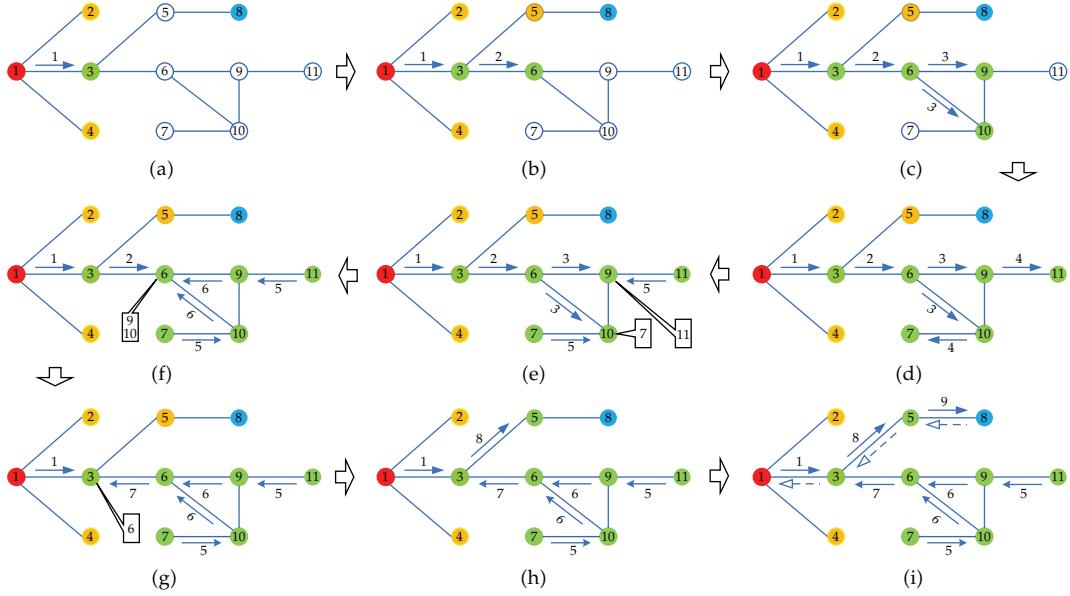
Figure 2: Search process of the maximum degree algorithm and the difference with the MD algorithm.

## 2.2. The Memory Function Preference Degree Algorithm

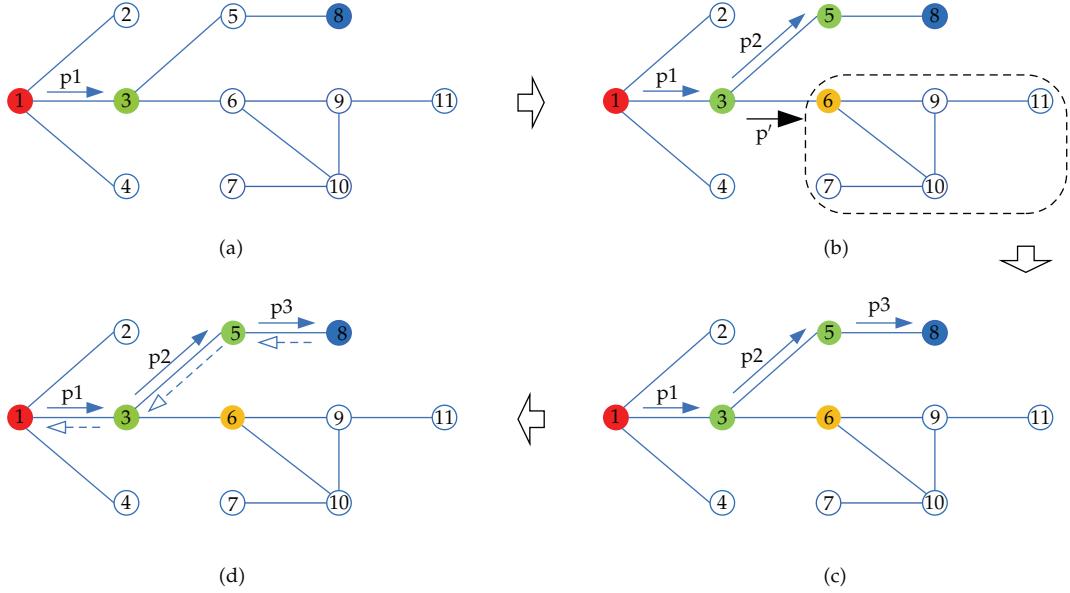
Compared to MD algorithm, the difference of PD algorithm is the search process. In the strategy of PD algorithm, when a node finds all the neighbors do not contain the resources, it randomly chooses the corresponding node with the preference probability  $\Pi$ . Assuming that the neighbors of the node  $A$  are  $A_1, A_2, \dots, A_n$ , and the degree of these neighbors are  $d_1, d_2, \dots, d_n$ , these degrees meet  $d_1 \leq d_2 \leq \dots \leq d_n$ . How to choose the preference node  $A_i$  is to compute as follows:

$$\begin{aligned}
 d &= \sum_{i=1}^n d_i, \quad i = 1, 2, \dots, n, \\
 \Pi_i &= \frac{d_i}{d}, \quad i \rightarrow A_i.
 \end{aligned} \tag{2.1}$$

In the worst conditions, the PD algorithm degenerates into the standard DFS algorithm for either same degree case or different-degree case. The search process of this algorithm is



**Figure 3:** Process of the MD algorithm searching the resource node with memory function.



**Figure 4:** Search process of the PD algorithm.

random, so the success rates are stochastic. The search process of this algorithm is shown in Figure 4. Unlike the MD algorithm, it randomly generates a preference probability  $p$  according to the neighbor nodes' degree. Figures 4(a)–4(d) are the specific process in the ideal conditions. In Figure 4(b), if the node 3 chooses the neighbor node 6 according to the stochastic preference, this algorithm will search resources in failure. It is shown as the dotted line box in Figure 4(b).

ID		Degree		Neighbor ID		Neighbor degree		Traverse flag		Returned ID		Resource key
----	--	--------	--	-------------	--	-----------------	--	---------------	--	-------------	--	--------------

**Figure 5:** List of message type.

### 3. Simulations and Discussions

Search message quantity, network resource search success rate, and search response time are the key parameters in unstructured P2P networks. Random graphs [31, 32] are widely used models, which help the study of the network. Because the degree distribution of the Internet nodes presents a power-law distribution, scale-free network and small-world network [33, 34] are the typical network structure to study the Internet. In this section, scale-free BA network, Watts-Strogatz (WS) small-world network, and ER random graph network are taken as the examples of network models, and numerical simulations are given to study these parameters. All the parameters are contrasted under the condition of search success. The maximum age is chosen 100 in the simulations. The complete list of message type is shown in Figure 5. The “traverse\_flag” of the list denotes the status of this node. When its value is 0, the node is not traversed. When its value is 1, the node is traversed and the search message does not pass the node. When its value is 2, the node is traversed and the search message passes the node.

#### 3.1. Search Messages Quantity

In modified BFS algorithm (MB), each node instead of forwarding a search message to all its neighbors, it randomly selects a subset of its neighbors to propagate the search request messages. The fraction of neighbors that are selected is a parameter to the mechanism. Given a P2P network, a node can search the others of the network more efficiently with a smaller number of messages compared with the standard BFS algorithm. So the total number of search messages is average to

$$\phi = \sum_{\text{age}=1}^T (d \cdot q)^{\text{age}}, \quad (3.1)$$

where  $T$  is the threshold,  $d$  is the average degree of a node,  $q$  in the chosen ratio and is general not more than 0.5. Here it is 0.25 in our simulations.

When considering the random walks case, the requesting node sends out one search message to a randomly chosen neighbor, that is, standard random walks algorithm. This search message is seen as a walker. Then the walker directly keeps in touch with the source node in the process of walking and asks whether to proceed to the next step. If the requestor is agreed to continue walking (the termination conditions have not been satisfied), it randomly chooses a neighbor to forward the walker. Otherwise, the algorithm terminates the walking process. The search message quantity of this algorithm is related to the age value; thus, it reduces the network loads and achieves a message reduction by over an order of magnitude compared to the standard BFS algorithm (it is also called flooding search algorithm in some literatures [1, 2]). In order to improve the search success rate, the requesting node sends out  $k$  search messages to its  $k$  neighbors, that is,  $k$  random walks algorithm, assuming that

the number of search messages for each hop keeps fixed as  $k$ , that is, the number of walkers. Therefore, the total number of search messages for random walks algorithm is

$$\phi = \sum_{\text{age}=1}^T k \cdot \text{age}, \quad k = 1, 2, \dots, N, \quad (3.2)$$

where  $T$  is the threshold of age. When the walker  $k$  meets  $k = 1$ , it is standard random walks algorithm(R1), and when  $k$  meets  $k > 1$ , it is  $k$  random walks algorithm (RK).  $k$  is general not more than 16 [22], and here it is 4 in our simulations.

In degree search algorithms, the query messages spread to these neighbor's nodes with the characteristics (the preference degree or the maximum degree) every step in the search process. In the context, degree search algorithms include the MD and PD. The characteristics include the maximum degree and preference degree. So the total number of search messages for degree search algorithms is

$$\phi = \sum_{\text{age}=1}^T m \cdot \text{age}, \quad (3.3)$$

where  $m$  is the number of nodes with the characteristics. For instance, in Figures 1(a) and 1(b),  $m$  value is 1 because the number of nodes with the maximum degree is 1. In Figure 1(c),  $m$  value is 2 because node 9 and node 10 have the same maximum degree.

Figure 6 shows the search message quantity ( $\phi$ ) generated by the various algorithms in different topology networks, where the average degree ( $D$ ) of the networks is chosen from 2 to 10. Figure 6(a) is the scale-free BA network, Figure 6(b) is the ER random graph network, and Figure 6(c) is the WS small-world network. The number of nodes of the networks is 5,000.

Simulations show that in the three cases, the search message quantity is increasing with the growth of the average degree of the networks. The search message quantity of MB algorithm confirms the exponential growth, and random walk algorithms reduce the message quantity into linear growth. The search message quantity of MD algorithm and PD algorithm is slightly less than the standard random walks algorithm, but, in general, they are the same order of magnitude according to the simulations. Due to memory function of MD algorithm and PD algorithm proposed in this paper, they reduce unnecessary search messages in the search process. Compared with  $k$  random walks algorithm, the search message quantity of MD algorithm and PD algorithm is less. These two algorithms can decrease the message quantity about 18 times than MB algorithm.

### 3.2. Search Success Rate

Search success is at least one request message sent from the requesting node seeks out the requested resources. Assume that the queried resources are uniformly distributed in the network with a replication ratio  $r$ . We calculate the search success rate ( $\Psi$ ) according to the following formula:

$$\Psi = \frac{C \cdot r}{N} \quad (3.4)$$

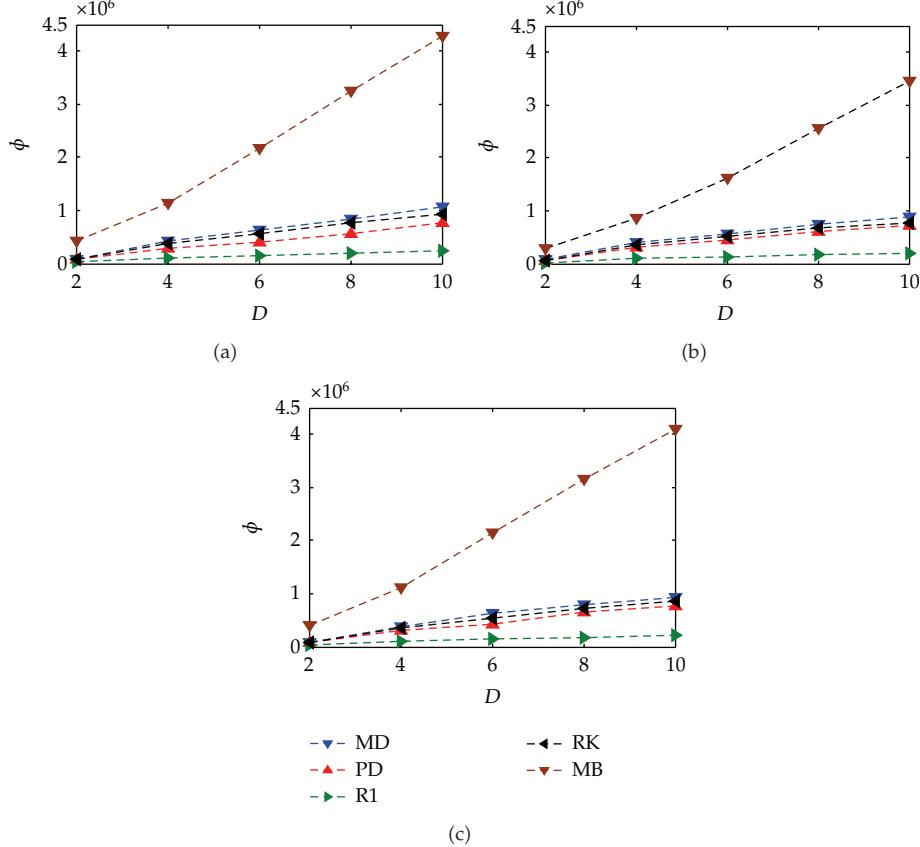


Figure 6: Message quantity ( $\phi$ ) generated by the various algorithms in different topology networks.

where  $r$  is the replication ratio,  $C$  is the number of nodes covered by the algorithm, and  $N$  is the total number of nodes of the network. This formula shows that the search success rate highly depends on the coverage of the search algorithms. The age value determines the coverage of the search algorithms. MB algorithm, random walks algorithm, and PD algorithm have random factors. Thus, their search success rates vary greatly depending on network topology and the random choices which have been made.

According to the results of our simulations, the age value of the search algorithms is not too large except the standard random walks algorithm. The maximum age value is chosen 100 in our simulations. In random walks algorithms, the walker  $k$  is chosen 4.

Figure 7 shows the search success ratio ( $\Psi$ ) of the various algorithms in different topology networks, where the average degree ( $D$ ) of the networks is chosen from 2 to 10. Figure 7(a) is the scale-free BA network, Figure 7(b) is the ER random graph network, and Figure 7(c) is the WS small-world network. Simulations show that in the three cases, the search success rates of the various search algorithms are increasing with the growth of the average degree of the networks. The search success rate of MD algorithm is the highest; the search success rate of RK algorithm is slightly higher than that of MB algorithm. The search success rate of R1 algorithm is the least. The search success rate of PD algorithm is better than the R1 algorithm and is inferior to the MB algorithm. The messages of MD algorithm can

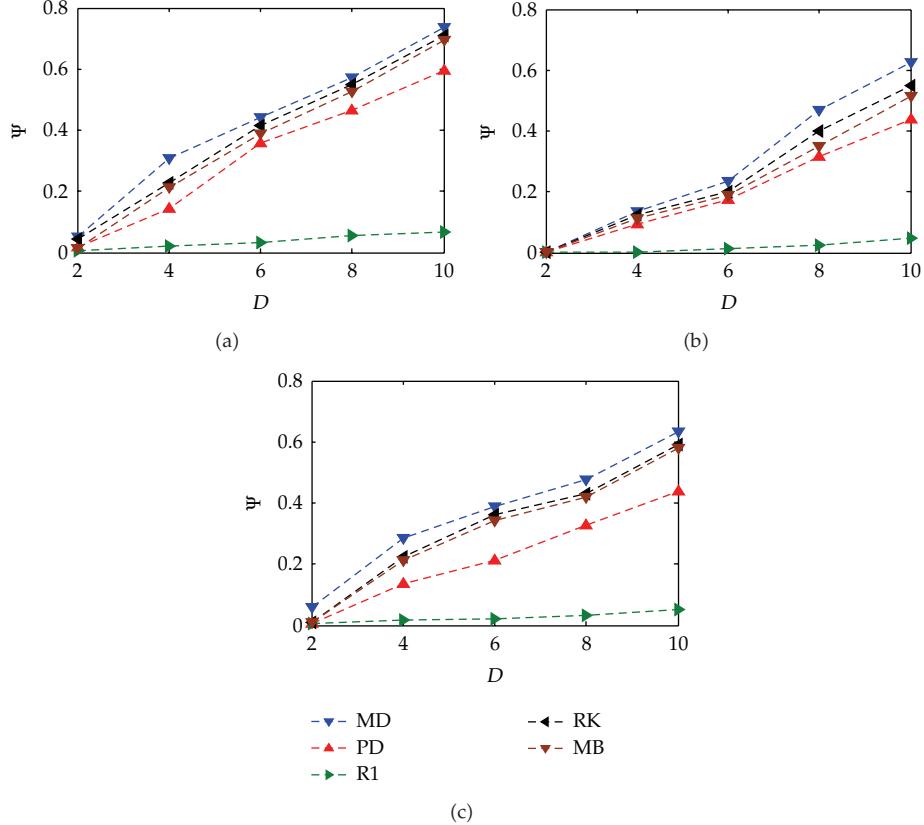
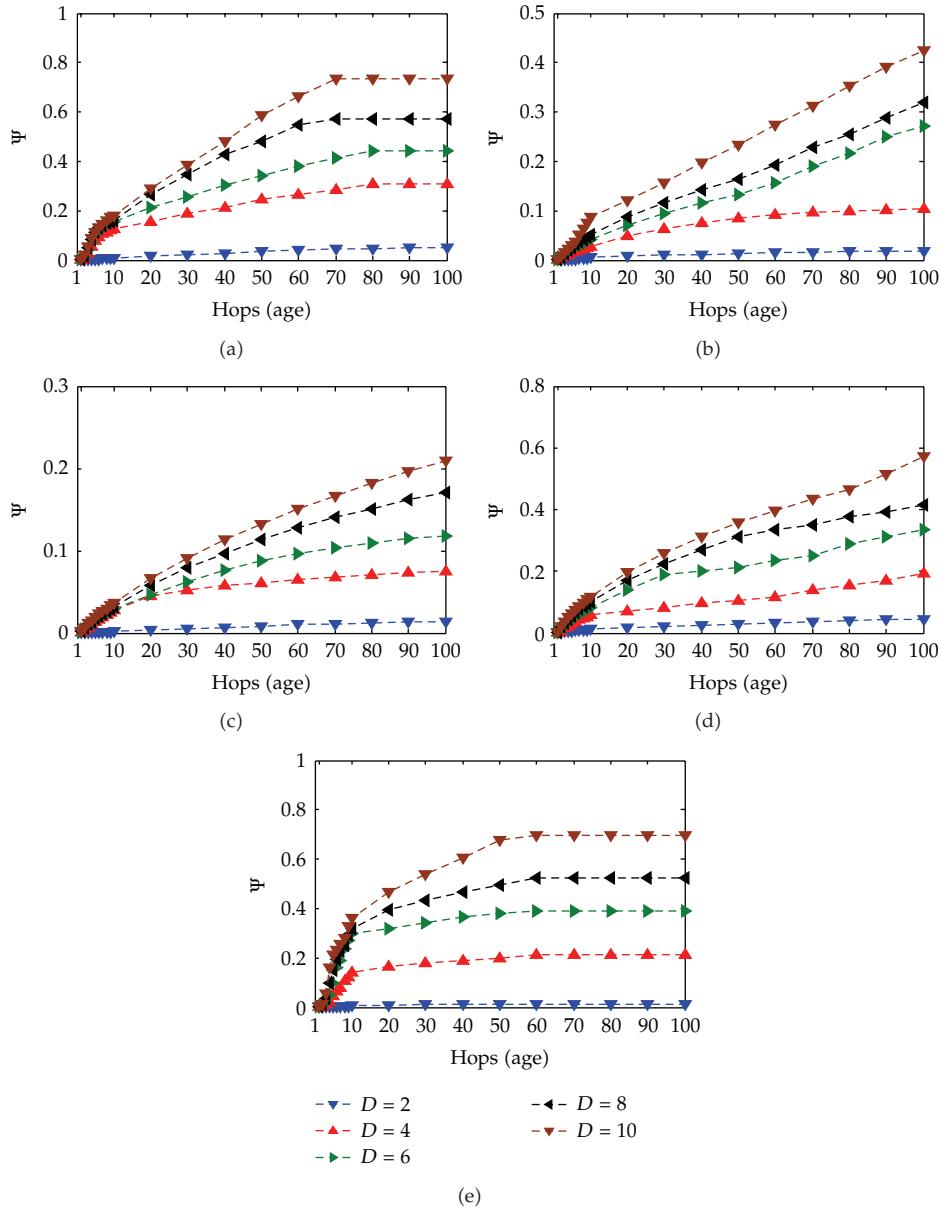


Figure 7: Search success ratio ( $\Psi$ ) of the various algorithms in different topology networks.

return the precursor, so it has more search scope. In RK algorithm, there are  $k$  random walkers to search the resources at the same time; the success rate is higher than that of the R1 algorithm. The MB algorithm, a random algorithm, can find the resources quickly if these resources locate in the chosen search paths. And it will generate the massive redundancy messages if the chosen search paths do not exit the resources. In our simulation, we calculate the mean value; the chosen proportion of MB algorithm is 0.25 and small, so the success rate is slightly inferior to that of MD and RK algorithm.

Figure 8 shows the search success ratio ( $\Psi$ ) of the various algorithms in scale-free BA networks, where the age value is changed from 1 to 100. Figure 8(a) is the MD algorithm, Figure 8(b) is the PD algorithm, Figure 8(c) is the R1 algorithm, Figure 8(d) is the RK algorithm, and Figure 8(e) is the MB algorithm. We can see that all the search success rates are increasing with the growth of the average degree. With the constraint of search ages, only the search success rate of MB algorithm is not affected. In particular, in the same scale of network and the same average degree of network, the search success rate of MD algorithm is the highest among the five algorithms, that of R1 algorithm is the least, and the search success rate of PD algorithm is between the R1 algorithm and the MB algorithm. To the search ages, the age value of MB algorithm is the least when the search success rate reaches the maximum, and the age value of R1 algorithm is the largest. The age values of MD algorithm and PD algorithm are between the R1 algorithm and the MB algorithm.



**Figure 8:** Search success ratio ( $\Psi$ ) of the various algorithms in scale-free BA networks.

Figure 9 shows the search success ratio ( $\Psi$ ) of the various algorithms in ER random graph networks, where the age value is changed from 1 to 100. Figure 9(a) is the MD algorithm, Figure 9(b) is the PD algorithm, Figure 9(c) is the R1 algorithm, Figure 9(d) is the RK algorithm, and Figure 9(e) is the MB algorithm. The simulations show that the search success rate is increasing with the growth of the average degree. In particular, in the same scale of network and the same average degree of network, the search success rate of MD algorithm is still the highest among the five algorithms, and that of R1 algorithm is the least. The search success rate of PD algorithm is between the R1 algorithm and the MB algorithm. To the search

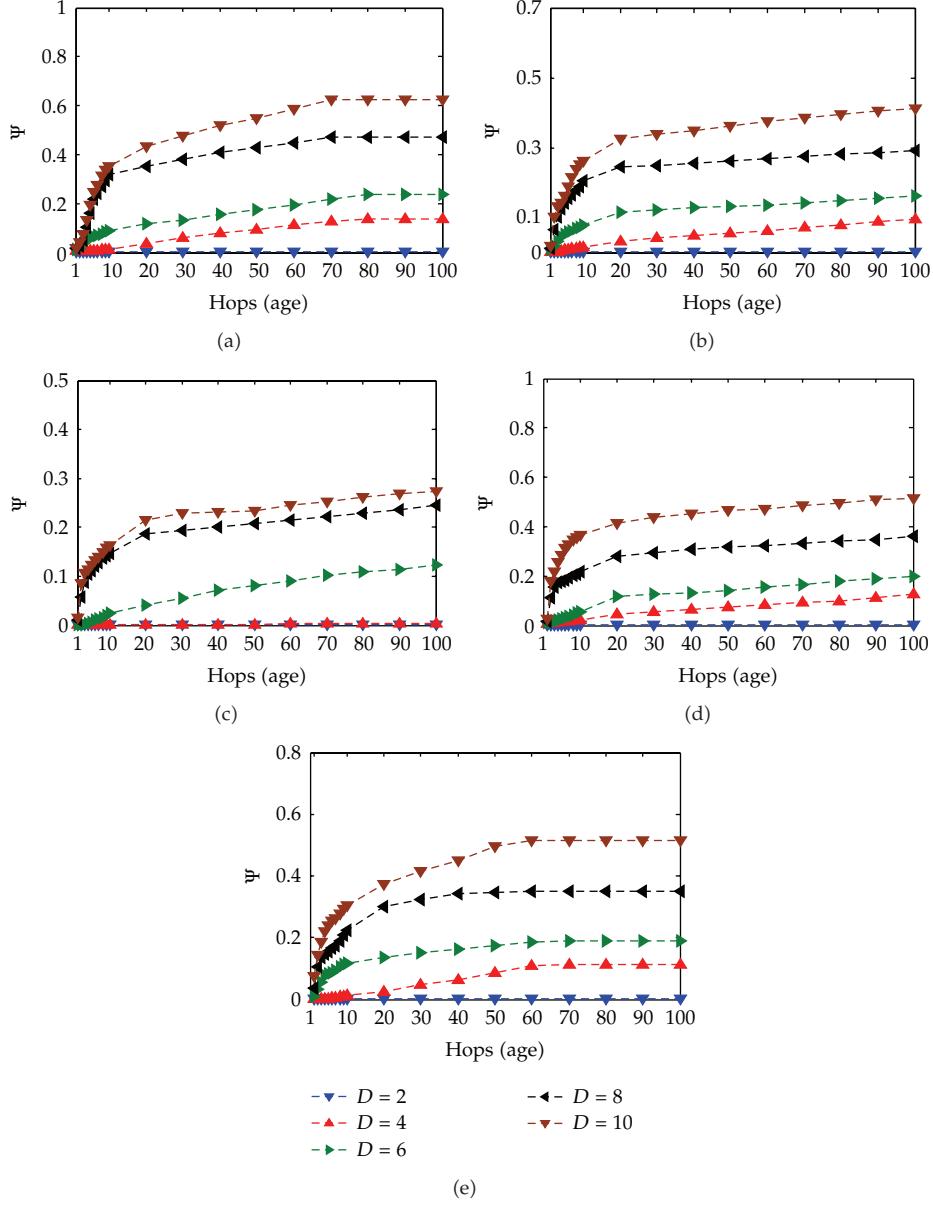


Figure 9: Search success ratio ( $\Psi$ ) of the various algorithms in ER random graph networks.

hops, the age value of MB algorithm is the least when the search success rate reaches the maximum, and that of R1 algorithm is the largest. The age values of MD algorithm and PD algorithm are between the R1 algorithm and the MB algorithm.

Figure 10 shows the search success ratio ( $\Psi$ ) of the various algorithms in WS small-world networks, where the age value is changed from 1 to 100. Figure 10(a) is the MD algorithm, Figure 10(b) is the PD algorithm, Figure 10(c) is the R1 algorithm, Figure 10(d) is the RK algorithm, and Figure 10(e) is the MB algorithm. We can see that the search success rate is increasing with the growth of the average degree. In particular, in the same scale of network

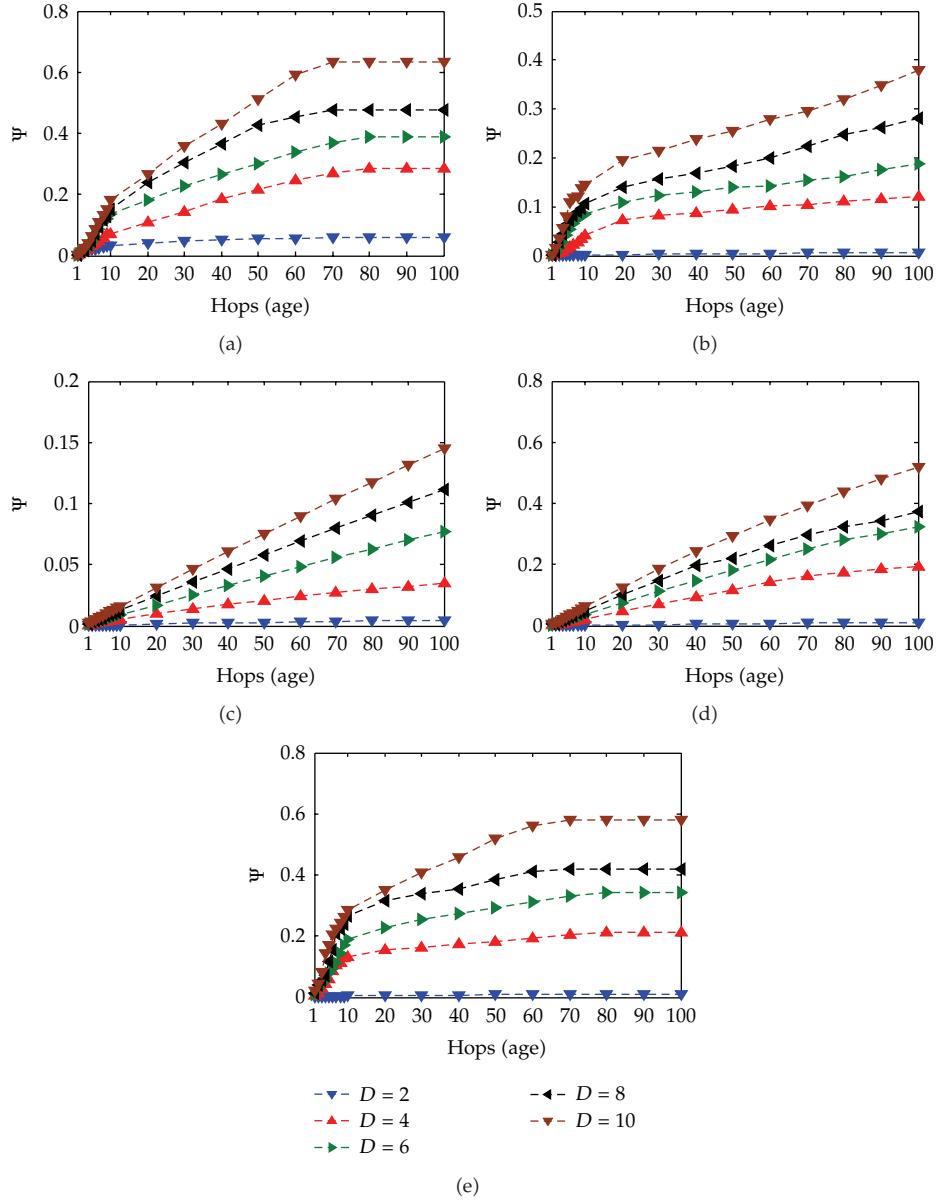


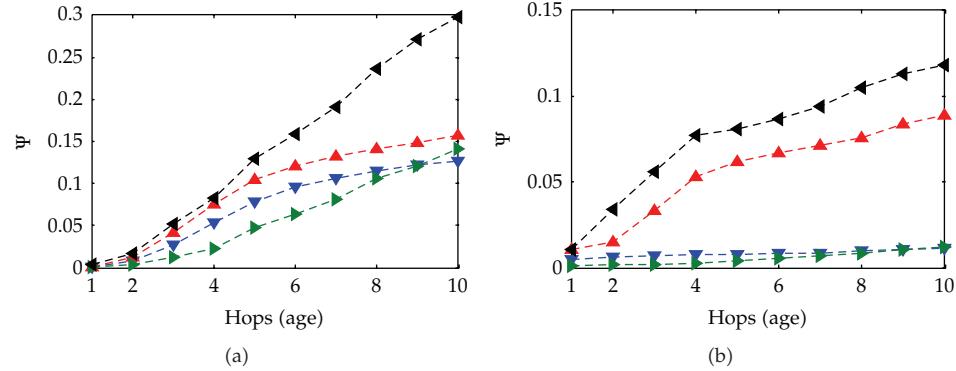
Figure 10: Search success ratio ( $\Psi$ ) of the various algorithms in WS small-world networks.

and the same average degree of network, the search success rate of MD algorithm is the highest among the five algorithms, that of R1 algorithm is the least, and the search success rate of PD algorithm is between the R1 algorithm and the RK algorithm. To the search hops, the age value of MB algorithm is the least when the search success rate reaches the maximum, and that of R1 algorithm is the largest. The age values of MD algorithm and PD algorithm are between the R1 algorithm and the MB algorithm.

In general, the search success rate of MD algorithm excels the MB algorithm and the random walks algorithm in the same conditions. But considering the smaller value of age,

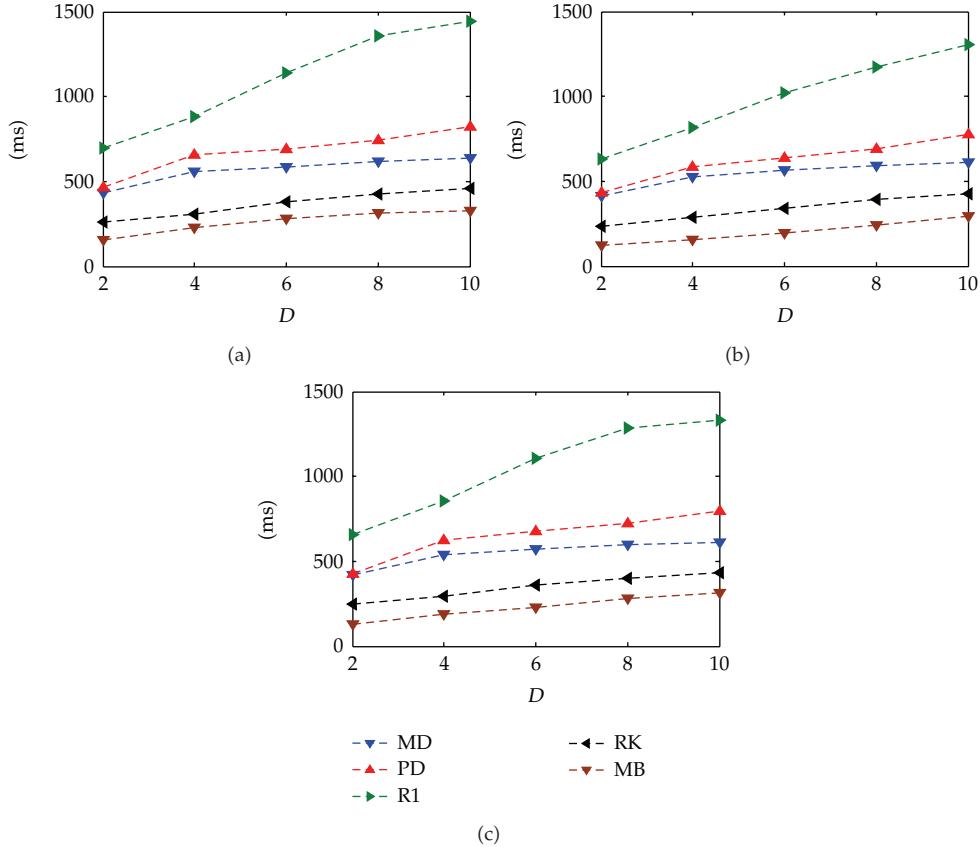
**Table 1:** The message quantity of the MD and MB algorithms in different topology networks.

D	Algorithm	Network topology	Message quantity
$D = 4$	MD	BA	$6.1986 \times 10^4$
		ER	$5.3581 \times 10^4$
		WS	$5.9506 \times 10^4$
	MB	BA	$2.8833 \times 10^5$
		ER	$2.4758 \times 10^5$
		WS	$2.6752 \times 10^5$
$D = 6$	MD	BA	$6.8913 \times 10^4$
		ER	$6.1551 \times 10^4$
		WS	$6.5726 \times 10^4$
	MB	BA	$1.1347 \times 10^6$
		ER	$7.4835 \times 10^5$
		WS	$9.3142 \times 10^5$

**Figure 11:** Search success ratio ( $\Psi$ ) of the MD and MB algorithms in different topology networks (BA, ER, and WS).

the success rate of MB is slightly higher than or equal to that of MD (e.g., the success rate of MB when the age value is 10 shown in Figures 8, 9, and 10). This is because the MD algorithm does not have enough hops to return the nodes with second highest degree when it cannot find the resource along the direction of nodes with highest degree. Thus, it returns search failure. With the age increasing, the success rate of MD algorithm gradually transcends that of MB algorithm. To PD algorithm, the search success rate is better than R1 algorithm but is inferior to RK algorithm and MB algorithm.

It is obvious that the search success rate highly depends on the coverage of the search algorithms defined by the age value. However, a large value of age will incur an essential large response time for a requester to obtain search results. A smaller value of age is more appropriate. Thus, we have given the MD and MB algorithm success rate and message quantity in the case of small age value. The response time of these two algorithms is very short and almost the same. The results are shown in Figure 11 and Table 1. From Figure 11, we can see that the success rate of MD algorithm is slightly better than that of MB algorithm when the average degree is small. Instead, the success rate of MB is higher when the average degree is large. And at the same time, MB algorithm generates a mass of messages as shown



**Figure 12:** Average response time generated by the various algorithms in different topology networks.

in Table 1; it is about 15 times higher than that of MD algorithm. Therefore, the comprehensive performances of MD algorithm are slightly higher than MB algorithm.

### 3.3. Search Response Time

An unstructured P2P network is a highly dynamic network, and the nodes of the network can join and leave freely. Thus, the search response time is a critical metric for measuring the performance. The search time should be short enough to make sure that the search result is update to date. We define the search response time of a query as the time period when the query is issued until when the source peer receives a response result from the first responder. We calculate the average response time in the condition of search success.

Figure 12 shows the average response time of the various algorithms in different topology networks, where the average degree ( $D$ ) of the networks is chosen from 2 to 10. Figure 12(a) is the scale-free BA network, Figure 12(b) is the ER random graph network, and Figure 12(c) is the WS small-world network. Simulations show that in the three cases, the average response time is increasing with the growth of the average degree of the networks. The search response time of MB algorithm is the least, and the R1 algorithm is the most time

consuming. The average response time of MD and PD algorithm is about twice as much as that of MB algorithm.

Although the average response time of MD algorithm is about twice higher than that of MB as shown in Figure 12, the message quantity of MB algorithm is about 18 times higher than that of MD as shown in Figure 6. When the age value is small as shown in Figure 11, the response time of MD and MB algorithm is almost the same, and the search success rate of these two algorithms is also close under the same values of D. However, the message quantity of MB is about 15 times higher than that of MD. The algorithm does not exit whose all performance indexes are perfect. Thus, in view of the comprehensive indexes, MD algorithm outperforms MB algorithm.

#### 4. Conclusion

This paper presents the design and evaluation of two memory function search algorithms over the unstructured P2P networks, which, respectively, built on the top of scale-free BA networks, ER random graph networks, and WS small-world networks. The performance of these two algorithms has been compared with the current algorithms used in existing unstructured P2P networks. The search success rate of MD algorithm is averagely 14 times better than the standard random walks algorithm, while the search message quantity is the same order of magnitude with it. Compared with modified BFS algorithm, the search success rate of MD algorithm is higher than that, while the search message quantity averagely reduces by over 18 times. Although PD algorithm can reduce the huge search message quantity, the search success rate of it is inferior to the modified BFS algorithm.

#### Acknowledgments

This paper is supported by the National Natural Science Foundation of China (Grant nos. 61170269, 61121061), the Foundation for the Author of National Excellent Doctoral Dissertation of PR China (Grant no. 200951), the Asia Foresight Program under NSFC (Grant no. 61161140320), the National Science Foundation of China Innovative (Grant no. 70921061), the CAS/SAFEA International Partnership Program for Creative Research Teams and the Program for New Century Excellent Talents in University of the Ministry of Education of China (Grant no. NCET-10-0239).

#### References

- [1] E. Meshkova, J. Riihijarvi, M. Petrova, and P. Mahonen, "A survey on resource discovery mechanisms, peer-to-peer and service discovery frameworks," *Computer Networks*, vol. 52, no. 11, pp. 2097–2128, 2008.
- [2] A. Iamnitchi, M. Ripeanu, I. Foster et al., "Small-world file-sharing communities," in *Proceedings of the IEEE INFOCOM*, vol. 2, pp. 952–963, 2004.
- [3] V. Kalogeraki, D. Gunopulos, and D. Zeinalipour-Yazti, "A local search mechanism for peer-to-peer networks," in *Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM '02)*, pp. 300–307, 2002.
- [4] B. F. Cooper and H. Garcia-Molina, "Ad Hoc, self-supervising peer-to-peer search networks," *ACM Transactions on Information Systems*, vol. 23, no. 2, pp. 169–200, 2005.
- [5] J. Risson and T. Moors, "Survey of research towards robust peer-to-peer networks: search methods," *Computer Networks*, vol. 50, no. 17, pp. 3485–3521, 2006.

- [6] J. Jeong and P. Berman, "Low-cost search in scale-free networks," *Physical Review E*, vol. 75, no. 3, Article ID 036104, 4 pages, 2007.
- [7] V. Isham, J. Kaczmarcza, and M. Nekovee, "Spread of information and infection on finite random networks," *Physical Review E*, vol. 83, no. 4, Article ID 046128, 12 pages, 2011.
- [8] D. Trpevski, W. K. S. Tang, and L. Kocarev, "Model for rumor spreading over networks," *Physical Review E*, vol. 81, no. 5, Article ID 056102, 11 pages, 2010.
- [9] D. Shah and T. Zaman, "Rumors in a network: who's the culprit?" *Institute of Electrical and Electronics Engineers. Transactions on Information Theory*, vol. 57, no. 8, pp. 5163–5181, 2011.
- [10] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *Institute of Electrical and Electronics Engineers. Transactions on Information Theory*, vol. 52, no. 6, pp. 2508–2530, 2006.
- [11] L. A. Adamic, R. M. Lukose, A. R. Puniyani, and B. A. Huberman, "Search in powerlaw networks," *Physical Review E*, vol. 64, no. 4, Article ID 046135, 8 pages, 2001.
- [12] A. Tizghadam and A. Leon-Garcia, "On traffic-aware betweenness and network criticality," in *Proceedings of the INFOCOM IEEE Conference on Computer Communications Workshops*, pp. 1–6, 2010.
- [13] S. Misra and B. J. Oommen, "An efficient dynamic algorithm for maintaining all-pairs shortest paths in stochastic networks," *IEEE Transactions on Computers*, vol. 55, no. 6, pp. 686–702, 2006.
- [14] A. Reinefeld and T. Marsland, "Enhanced iterative-deepening search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 7, pp. 701–710, 1994.
- [15] Y. Yao and J.-C. Yao, "On modified iterative method for nonexpansive mappings and monotone mappings," *Applied Mathematics and Computation*, vol. 186, no. 2, pp. 1551–1558, 2007.
- [16] Z. Wang, S. K. Das, M. Kumar, and H. Shen, "An efficient update propagation algorithm for P2P systems," *Computer Communications*, vol. 30, no. 5, pp. 1106–1115, 2007.
- [17] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker, "Search and replication in unstructured peer-to-peer networks," in *Proceedings of the 16th international conference on Supercomputing (ICS '02)*, pp. 84–95, 2002.
- [18] C. Gkantsidis, M. Mihail, and A. Saberi, "Random walks in peer-to-peer networks," in *Proceedings of the IEEE INFOCOM*, vol. 1, pp. 1–12, 2004.
- [19] A. Kumar, J. Xu, and E. W. Zegura, "Effcient and scalable query routing for unstructured peer-to-peer networks," in *Proceedings of the IEEE INFOCOM*, vol. 2, pp. 1162–1173, 2005.
- [20] C. Gkantsidis, M. Mihail, A. Saberi et al., "Random walks in peer-to-peer networks: algorithms and evaluation," *Performance Evaluation*, vol. 63, no. 3, pp. 241–263, 2006.
- [21] M. Zhong, K. Shen, and J. Seiferas, "The convergence-guaranteed random walk and its applications in peer-to-peer networks," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 57, no. 5, pp. 619–633, 2008.
- [22] T. Lin, P. Lin, H. Wang, and C. Chen, "Dynamic search algorithm in unstructured peer-to-peer networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 5, pp. 654–666, 2009.
- [23] X. Ming, Z. Shuigeng, G. Jihong, and H. Xiaohua, "A path-traceable query routing mechanism for search in unstructured peer-to-peer networks," *Journal of Network and Computer Applications*, vol. 33, no. 2, pp. 115–127, 2010.
- [24] K. Oikonomou, D. Kogias, I. Stavrakakis et al., "Probabilistic flooding for efficient information dissemination in random graph topologies," *Computer Networks*, vol. 54, no. 10, pp. 1615–1629, 2010.
- [25] R. Beraldì, L. Querzoni, and R. Baldoni, "Low hitting time random walks in wireless networks," *Wireless Communications and Mobile Computing*, vol. 9, no. 5, pp. 719–732, 2009.
- [26] F. Guofu, Z. Jincheng, L. Sanglu, C. Guihai, and C. Daoxu, "RER: a replica efficiency based replication strategy," *Chinese Journal of Electronics*, vol. 17, no. 4, pp. 589–594, 2008.
- [27] G. Huiwei and I. H. S. Horace, "A study of parallel data mining in a peer-to-peer network," *Concurrent Engineering-Research and Applications*, vol. 15, no. 3, pp. 281–289, 2007.
- [28] N. Bisnik and A. A. Alhussein, "Optimizing random walk search algorithms in P2P networks," *Computer Networks*, vol. 51, no. 6, pp. 1499–1514, 2007.
- [29] Y. Liu, J. Han, J. Wang et al., "Rumor riding: anonymizing unstructured peer-to-peer systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 3, pp. 464–475, 2011.
- [30] C.-K. Chau and P. Basu, "Analysis of latency of stateless opportunistic forwarding in intermittently connected networks," *IEEE/ACM Transactions on Networking*, vol. 19, no. 4, pp. 1111–1124, 2011.
- [31] M. E. J. Newman, "Scientific collaboration networks. I. Network construction and fundamental results," *Physical Review E*, vol. 64, no. 1, Article ID 016131, 8 pages, 2001.
- [32] A. Fronczak, P. Fronczak, and J. A. Holyst, "Average path length in random networks," *Physical Review E*, vol. 70, no. 5, Article ID 056110, 7 pages, 2004.

- [33] L. A. Adamic and B. A. Huberman, "Growth dynamics of the world wide web," *Nature*, vol. 401, p. 131, 1999.
- [34] M. Ostilli, A. L. Ferreira, and J. F. F. Mendes, "Critical behavior and correlations on scale-free small-world networks: application to network design," *Physical Review E*, vol. 83, no. 6, Article ID 061149, 21 pages, 2011.

*Research Article*

## **The Number of Students Needed for Undecided Programs at a College from the Supply-Chain Viewpoint**

**Jin-Ling Lin,<sup>1</sup> Jy-Hsin Lin,<sup>2</sup> and Kao-Shing Hwang<sup>3</sup>**

<sup>1</sup> Department of Information Management, Shih Hsin University, Taipei 11678, Taiwan

<sup>2</sup> Department of Industrial Engineering and Management Information, Huafan University, Taipei 223, Taiwan

<sup>3</sup> Department of Electrical Engineering, National Sun Yat-Sen University, Kaohsiung 80424, Taiwan

Correspondence should be addressed to Jin-Ling Lin, jllin@cc.shu.edu.tw

Received 30 October 2011; Revised 11 March 2012; Accepted 25 April 2012

Academic Editor: Yi-Chung Hu

Copyright © 2012 Jin-Ling Lin 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 objective of this study is to determine how to do forward-looking analysis to determine the future need for various professionals, so that educational institutes can produce an appropriate labor force for national development. The concept of component part commonality, which derives from research on effective inventory management using material resource planning (MRP), can also be applied to human resource planning and educational resource management in higher education systems. Therefore, this paper proposed a systematic method to analyze student recruitment numbers for future needs, based on the concept of MRP. The research studied the relationship between a curricular structure tree and the associated commonalities. It explored the relationship between the commonality of students and the aggregated level of student replenishment. Based on that, systematic guidelines for curriculum design can be established for undeclared programs at collages. Two simple examples were used to illustrate the implementation of MRP in analysis of the replenishment levels (necessary safety stock levels) in an education system such as an engineering college.

### **1. Introduction**

Technology can improve on conventional methods of allocating a nation's resources, including human resources. Meanwhile, international economic competition is increasing with the widening trade. Government protected industries, especially in newly industrialized countries, such as Taiwan and its neighbors, are encountering stronger competition than ever before. Therefore, these countries must join with foreign countries, even if involuntarily, to

open their domestic markets. To deal with this situation, these newly industrialized countries need to coordinate and integrate their limited resources to obtain optimal utilization. In other words, they should take steps to speed up technological research and development to increase their ability to compete. Success in these steps depends strongly on whether there are sufficient human resources in each professional area [1]. Therefore, a proper human resource policy is one of the determining factors for a national survival in light of this keen economic competition. Since educating a professional usually takes several years, a responsible government studies the future needs and develops policies, in advance, so that the education system can develop enough of the needed professionals [2].

In Taiwan, colleges or universities are the major institutes for training high quality workers. Qualified entrants have to fulfill strict curricular requirements before graduating. The study procedure in schools is analogous to the production process in industrial manufacturing. Thus, the concept of material requirements planning (MRP) can be applied to an education system. The very nature of an MRP system revolves around commonality, which means different final products still consist of some common components. An MRP system, based on the degree of commonality, determines the minimum aggregate safety stock levels needed to prevent out-of-stock situations and satisfy customer demands. If an undeclared programs were developed in universities, the designing of an appropriate curriculum for undeclared students needs to be prepared in advance. Curriculum decisions would be dependent on the undeclared program's student recruitment totals.

Nevertheless, a college student's major is determined by his score on the National Union College Entry Examination in Taiwan. A semigovernmental organization assigns prospective students to a department at a college or university according to their score on the examination and their choice list. Most students are assigned to a college by their examination scores, in spite of their interests. Meanwhile, in the second year of high school students are categorized into four groups according to their future majors. A type of professional streaming is premature for pupils, who are just fifteen or sixteen year-old. Early streaming reduces the breadth of a student's knowledge so that student may lack the capability to integrate related professional areas. Confucian concepts emphasize respect for the individual's will and educational interests. Therefore, students should be encouraged not to make any premature career decisions before gathering enough information and understanding their abilities and interests. Therefore, the Taiwanese Ministry of Education (MOE) has issued a proposal suggesting that the college education system should postpone professional streaming until the junior year of university. The government expects that the higher education system should be able to educate a professional with strongly integrated technological abilities by postponing professional curricular streaming [3]. Recently some Taiwanese universities, such as National Tsing Hua University and National Sun Yat-sen University, have provided undeclared programs for incoming freshmen. The primary objective of the program is to let each student develop an interest-oriented academic career and to broaden the recruitment pool for prospective student.

The objective of this study is to determine how to do forward-looking analysis to determine the future need for various professionals, so that educational institutes can produce an appropriate labor force for national development. According to the Ministry of Education proposal, universities should recruit students without major streaming at the college entry level. Recruited students can choose not to declare their major until they have finished a two-year common curriculum. Students are then assigned to a department based on their interests and capabilities as determined by their academic performance. This method is completely different from the current one, where a student's major is determined when they obtain

admission to a university. However, two problems arise from this proposal. First, a school's reputation would be the main consideration for a student instead of their major after the two-year common curriculum. This situation will worsen the current problem encountered by the Taiwanese education system, where most national universities are too crowded, while many private universities cannot recruit enough students, causing poor allocation of limited educational resources [4].

Second, since students in the same institute would have a common curriculum, their professional training would have a high degree of commonality. Although student choices among various professional majors would be more flexible, it can be predicted that most students would gravitate to popular departments, even if they were not interested or capable. This will lead to a problem where some departments will not be able to recruit as many students as under the present curricular streaming system. Moreover, each university will invest most of its budget in popular departments to attract more students without considering the social costs. From a long-term point of view, the planning and training of the professional labor force may be distorted, and some resources for certain professions could be diluted [5]; this will surely have educational and social costs.

The dropout rate, especially in the upper years, is high in private universities. The phenomenon leaves the fewer college students, especially in the upper years, and it has been a problem in Taiwan for a long time. The higher education system deals with this problem by recruiting transfer students from the technical education system (High schools in Taiwan are categorized into two groups: the so-called ordinary schools, where students are educated to enter colleges, or technical schools, which provide practical vocational training, so students can be employed by industry upon graduation or go to technical colleges. However, due to convention, most students from technical schools would rather compete with students from ordinary high schools to enter ordinary universities, instead of technical colleges. This phenomenon totally distorts the authority's original plan for the higher education system.) This approach lacks an integrated view and forward-looking consideration of future industrial needs and resource allocation. Since educational resources are being drained and curricular streaming is going to be postponed, how to allocate the limited resources efficiently is an important matter for Taiwanese educational authorities. Moreover, Taiwan's MOE subsidizes universities by the number of students recruited, but penalizes universities with dropout or vacancy rates that are too high. Under such circumstance, accurate estimation student number is crucial for curriculum design. This paper proposes a systematic method for analyzing student recruitment numbers to meet future needs based on commonality of academic content if streaming were postponed until the junior year.

## 2. Research Approach

Material requirement planning (MRP) [6] has been applied to inventory control management manufacturing in recent years and can likewise be applied to determining educational needs. This approach calculates the total supplies of product components at various manufacturing stages based on the commonality of each component part among the manufactured products, the product structures, and lead-time factors, such as ordering, manufacturing, and delivery. In other words, it uses the relationship between commonality and material costs, production, ordering, and delivery lead-time offsets, to consolidate and transfer the final products demanded during different time periods into the needed subassemblies of components. The objective of this approach is to combine independent orders for final products and maintain the quantity of orders at the highest possible level, so a factory can maintain raw materials

and component stocks at lower levels since it is economical. This approach reduces total costs by minimizing the cost per unit and setup costs. The commonality of component parts among different products has a significant impact on inventory levels for components and subassemblies. As commonality increases, the same component can be placed in many different products. When demand for a final product changes, the inventory of the lowest common component used must be properly adjusted to accommodate the change and avoid waste. Hence, the determination of the inventory level must be based on a global view so that limited resources can be utilized efficiently and flexibly to satisfy variations in the quantity of final products ordered.

The concept of component part commonality derives from research on effective inventory management using MRP. However, the concept can also be applied to human resource planning and educational resource management in higher education systems. In MRP, the master production schedule (MPS) must be established first according to predictions about end-product needs. This ensures that varying demand over different time periods can be met by production capacity. Similarly, MPS concepts can also be applied to long-term human resource planning, where the future demand for various professionals dictates the skills needed by the talent pool. The education and production processes are very similar as value-added processes. The fulfillment of curriculum requirements by a student is analogous to the production process of any industrial product. College freshmen, like raw material, are prepared for various professions through different core courses taken during the learning process. Alumni with different majors can be viewed as different end products. Hence, a learning process which trains the freshmen for different professions is analogous to the assembly process in a production line. Like MRP where the end product can be represented by the bill of materials (BOM), the training of a student can be represented by a similar hierarchical structure.

During the learning process, college students have a high degree of commonality after they have completed the basic courses. Once students begin to take the core courses in their major; however, even though some core courses are common to different majors (curricula), the streaming process begins. During the streaming process, students' core courses follow a predetermined order. Those who fail a prerequisite course must retake the course before they can proceed to advanced courses. Some students may be forced to change their major or drop out if they continuously fail in core courses. This process is just like the production process, where defective products must be reworked or repaired before they can be processed onto the next stage of manufacturing. Those with serious defects which cannot be repaired have to be scrapped. Hence, from the operational viewpoint, the learning process is very similar to the production process.

In MRP, to satisfy the demand for the end product, it is necessary to have some safety stock to compensate for unavoidable waste during production. The safety stock is a cushion for production variations; it ensures that raw materials and components can be supplied in time without shortage. Due to limited resources, the amount of safety stock must be planned from the aggregate viewpoint to reduce unnecessary costs and to increase production efficiency. Similarly, the safety stock concept can be applied to the education process; some students drop out of the system during the process and cause a shortage that can usually be replenished by enrolling more transfer students or freshmen. However, graduate quotas are based on the individual program requirements; that is, each program decides how many supplementary transfer students or freshmen it needs without considering the replenishment of students from an aggregate viewpoint. This is deemed a waste of precious resources. The problem of not satisfying the preset target for professionals can be resolved by means of

the commonality concept in MRP, which uses an aggregate student replenishment scheme to solve the unavoidable problem of a shortage of students.

Since similarities exist between the education process and the production process, it is possible to implement an MRP and commonality models in the education system. This paper studied the relationship between a curricular structure tree and the associated commonalities. Two simple examples were used to illustrate the implementation of MRP in analysis of the replenishment levels (necessary safety stock levels) in an education system such as an engineering college.

### **3. Model Analysis**

This paper describes how to apply the MRP method to analysis of student replenishment and demonstrates that the MRP concept can be suitably implemented in an education system [7]. The compatibility of implementation between manufacturing systems and education systems can be observed from the MRP structure trees for a product and curriculum. In a manufacturing system, the MRP structure tree of a product represents the relationship among final products, semiproducts, and components. This hierarchical structure is composed of components at the lowest level, semiproducts and final products at the highest level. Similarly, the common and advanced courses in a curriculum can be used to construct an MRP tree based on a predetermined sequence, as shown in Figure 1.

In Figure 1, each node (element) denotes a course, which a student should take to gain essential knowledge. From another viewpoint, each node also represents the learning (training) status of a student. Therefore, for the purpose of illustration, in this paper, it was assumed that a node, that is, a course offered, also represented a student whose learning status has reached this course's level. Therefore, the inventory control method in MRP can also be applied to analyze student recruiting in an education system. The following example shows how the concept of commonality between products can significantly affect the safety levels of inventory required. MRP can be used to analyze how the commonality among qualified students in different disciplinary curricula affects the number of supplementary students.

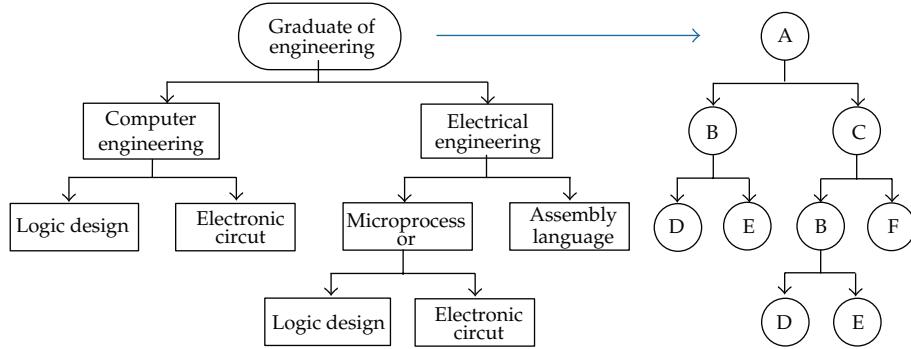
#### **3.1. Degree of Commonality**

The degree of commonality index, DCI, denoted a measurement of commonality, the degree of difference among some products [8–10]. It was expressed as  $C = \sum_{j=1}^d \phi_j / d$ , where  $\phi_j$  was the number of parents of course  $j$  in the curricular structure tree;  $d$  was the number of prerequisite courses, which had any ancestor; the index range was  $1 \leq C \leq \sum_{j=1}^d \phi_j = \beta$ . For example, in Figure 2, the DCIs were evaluated.

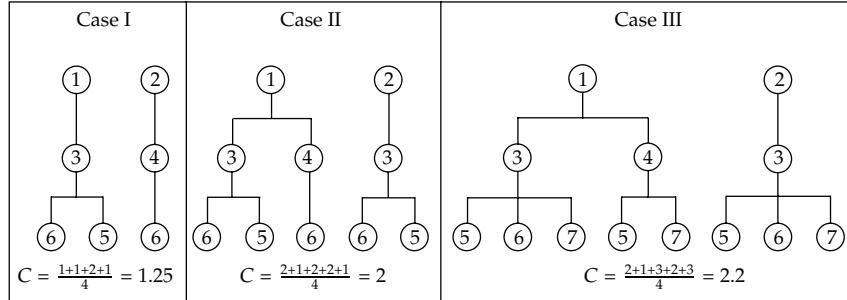
In other words, the DCI is a ratio between the number of prerequisite courses at the lower levels and the total number of their parents (related upper year courses). It can be used to represent the degree of commonality between students through different curricular disciplines [11, 12]. If two curricula in a college were independent, then  $C = 1$ , as shown in Figure 3.

#### **3.2. Curricular Commonality and the Number of Supplementary Students**

*Example 3.1.* A simplified two-level curricular structure tree were used to represent the case where a college recruits students without postponing streaming, Figure 4(a). In this



**Figure 1:** The curricular structure tree of an engineering college.



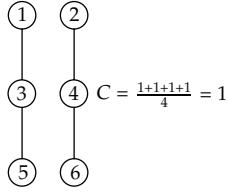
**Figure 2:** Examples for evaluating the degree of commonality indexes.

example, there are ten different curricula (majors) in a college. Students should take the course represented by the upper node after they fulfill the lower node course's requirements.

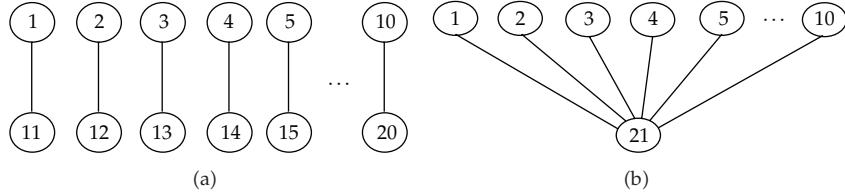
If the college postponed streaming, the freshman and sophomore students took the common core courses regardless of their future majors. Thus, most of the prerequisite courses at the lower level were aggregated, as shown in Figure 4(b), for example. The prerequisites (11–20) for these ten majors were merged into a common prerequisite (21). Using this example, the main effect resulting from increased commonality can be studied.

The DCI can be applied to any node or level in the curricular tree. Therefore, the students that reached a certain node, that is, who took the prerequisite courses (antecedents) for that node, were tagged by the DCI. For the course or subject  $j$ , the level of student replenishment  $S_j$  was equal to the value of the safety parameter  $k$  multiplied by the predicted standard deviation  $\sigma_j$ . If the actual demand  $y$  exceeded the expected demand  $u$  and was greater than the probability of student replenishment level, it can be expressed by Tchebysheff's inequality:

$$\text{Prob}\{(y - u) \geq k\sigma_j\} \leq \frac{1}{k^2 + 1}, \quad (3.1)$$



**Figure 3:** An example where there was no commonality between two curricula.



**Figure 4:** Two-level curricular structure trees.

where parameter  $k$  represents the upper bounds of the safety parameter. In the worst case, the probability of a student replenishment shortage at any level can be derived by substituting  $S_j/\sigma_j$  for  $k$ :

$$\Psi = \text{Prob}\{(y - u) \geq S_j\} = \frac{\sigma_j^2}{S_j^2 + \sigma_j^2}. \quad (3.2)$$

If there were a demand for students (called differentiation-type students) who already took the various prerequisite courses, (11··20) in Figure 4, then these students could be replaced by students (called common-type students) who have taken the common course, for example, (21) in Figure 4. The variance of the independent variable C is

$$\sigma_c^2 = \sigma_1^2 + \sigma_2^2 + \cdots + \sigma_d^2, \quad (3.3)$$

and the demand variance of prerequisite  $j$  is

$$\sigma_j^2 = \frac{1}{d} \cdot \sigma_c^2. \quad (3.4)$$

Substituting (3.4) into (3.2), the supplementary level of the  $j$ -type student is

$$S_j = \sqrt{\frac{1 - \Psi}{\Psi}} \cdot \sigma_c \sqrt{\frac{1}{d}}. \quad (3.5)$$

If there are  $d$  common-type students, then

$$\sum_{j=1}^d S_j = \sqrt{\frac{1-\Psi}{\Psi}} \cdot \sigma_c \sqrt{d}. \quad (3.6)$$

The optimal number of supplementary students was assumed to be  $S_c = \sqrt{(1 - \Psi)/\Psi} \cdot \sigma_c$  if the differentiation-type students were replaced by the affine common-type students. The total number of supplementary students was  $1/\sqrt{d}$  of the accumulated number of differentiation-type students; that is,  $S_c = 1/\sqrt{d} \sum_{j=1}^d S_j$ .

Since  $Cd = \sum_{j=1}^d \phi_j = \beta$  and we let  $\beta = 1$ , as in the case of Figure 4,  $d$  differentiation-type students were replaced by affine common-type students. Therefore,  $d = 1, C = d/1$ . And

$$S_c = \frac{1}{\sqrt{C}} \sum_{j=1}^d S_j. \quad (3.7)$$

Equation (3.7) expresses the relationship between the number of incoming common-type students and the number of replaced differentiation-type students. Due to the degree of commonality, the number of needed supplementary common-type students was less than that of differentiation-type students. From this example, it can be noted that

- (a) a simplified two-level structure tree was used. However, if the number of structure levels were more than two, the demand would not have been fully independent due to the correlation between the elements in the tree,
- (b) if  $N$  common-type students were needed and each of them was equivalent to  $d$  differentiation-type students, then the CDI was

$$C = \sum_i^N \frac{d}{N} = \frac{Nd}{N} = d, \quad (3.8)$$

and the total number of supplementary students was

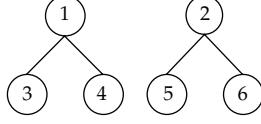
$$\sum_{i=1}^N S_c = \frac{\left( \sum_{i=1}^N \sum_{j=1}^d S_j \right)}{\sqrt{C}}. \quad (3.9)$$

*Example 3.2.* From the above example, it was noted that the demand for supplementary students can be reduced if the degree of commonality increases. The other example, where there are two simplified two-level curricula, can be used to show how the common-type student demand and differentiation-type student demand interacted with each other.

Example 3.1 depicts a simple case where the differentiation-type students were replaced by common-type students. However, further study is needed on the situation where there are some partially common components. The following example assumed two professions were needed, and that the distribution of demand probability for both professions was uniform:  $(0, b_1)$  for Profession1 and  $(0, b_2)$  for Profession2. If  $b_1 \geq b_2$  and  $(Z_1, Z_2)$  was a vector representing the demand for these professions, then the example describes the resolution of an optimization problem where the total cost of student replenishment should be minimized, subject to the service level. The aggregate service level, ASL, is the probability that all the demands will be satisfied [13]. ASL was assumed to be  $\gamma$  in this example.

*Case 1.* There was no commonality between the two curricula as shown in Figure 5.

It is a unique phenomenon in Taiwan that a college's desirability among senior high school students is strongly affected by the college's rankings in the previous year's entry

**Figure 5:** Two independent curricula.

examinations. Therefore, a college's department may become a low-priority choice based on the low scores of students recruited in the previous year. Meanwhile, too many colleges have been established in recent years and the population of potential students is decreasing dramatically due to the rapidly declining birthrates. Many private universities have suffered from lower student recruitment. Therefore many universities are working on strategies to raise their rankings on student priority lists by reducing the new student quota to ensure less vacancy during registration. Therefore, the problem can be formulated as follows:

$$\begin{aligned} \text{Minimize} \quad & T = S_3 + S_4 + S_5 + S_6, \\ \text{S.T.} \quad & \text{ASL} = \gamma, \end{aligned} \tag{3.10}$$

where  $S_i$  represents the number of  $i$ -type students;  $T$  is the total number of students.

Since the number of supplementary students was equal to the difference between the total number of students and the expected demand, that is, the *safe supplementary level* =  $T - (b_1 + b_2)$ , minimizing the number of supplementary students is equivalent to minimizing the total number of students. Therefore, to satisfy all the constraints, the conditions  $Z_1 \leq \min(S_3, S_4)$  and  $Z_2 \leq (S_5, S_6)$  must exist. Let  $S_3 = S_4$  and  $S_5 = S_6$ , then  $\text{ASL} = \text{Prob}\{Z_1 \leq Z_3\} * \text{Prob}\{Z_2 \leq Z_6\} = S_3 S_6 / b_1 b_2 = \gamma$ .

Meanwhile,

$$\begin{aligned} \text{Minimize} \quad & T = S_3 + S_4 + S_5 + S_6 = 2(S_3 + S_6) \\ \text{S.T.} \quad & S_3 S_6 = \gamma b_1 b_2 \\ & S_3^* = \gamma b_1, \quad S_6^* = b_2, \quad T^* = 2(\gamma b_1 + b_2). \end{aligned} \tag{3.11}$$

Table 1 lists the solutions for  $S_3$  and  $S_6$  in different ranges of ASL and  $T$  [8].

*Case 2.* There was some commonality between the two curricula as shown in Figure 6.

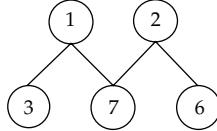
In this example, the common-type node 7 replaced the differentiation-type node 4 and node 5 from the previous example. The inequality of  $S_7 \geq S_3$ ,  $S_7 \geq S_6$ , and  $S_7 \leq (S_3 + S_6)$  must hold. Meanwhile,  $\text{ASL} = \text{Prob}\{Z_1 \leq S_3 \text{ and } Z_2 \leq S_6 \text{ and } (Z_1 + Z_2) \leq S_7\}$ . The shadow region in Figure 7 represents the probability of the ASL.

From Figure 7, the ASL was evaluated as  $[(S_7 - S_3)(S_7 - S_6) + (S_3 + S_6 - S_7)(S_7 - S_3) + (S_3 + S_6 - S_7)(S_7 - S_6) + (S_3 + S_6 - S_7)/2]/b_1 b_2$ . That is  $[(S_7 - S_3)(S_7 - S_6) + (S_3 + S_6 - S_7)(3S_7 - S_3 - S_6)/2]/b_1 b_2$ . The problem was reformulated as follows:

$$\begin{aligned} \text{Minimize} \quad & T = S_3 + S_6 + S_7 \\ \text{S.T.} \quad & 2(S_7 - S_3)(S_7 - S_6) + (S_3 + S_6 - S_7)(3S_7 - S_3 - S_6) = 2r b_1 b_2. \end{aligned} \tag{3.12}$$

**Table 1:** Solutions for  $S_3$  and  $S_6$  in different ranges of ASL and  $T$ .

ASL	$T^*$	$S_3^*$	$S_6^*$
$\gamma \leq \frac{b_2}{b_1}$	$\sqrt{(16\gamma b_1 b_2)}$	$\sqrt{(\gamma b_1 b_2)}$	$\sqrt{(\gamma b_1 b_2)}$
$\gamma \geq \frac{b_2}{b_1}$	$2[\gamma b_1 + b_2]$	$\gamma b_1$	$b_2$

**Figure 6:** The two curricula overlapped.

Furthermore,  $S_3 \leq \min(b_1, S_7)$ ,  $S_6 \leq \min(b_2, S_7)$ , and  $S_7 \leq (S_3 + S_6)$ . Using Lagrangian methods, the solutions in Table 2 were obtained.

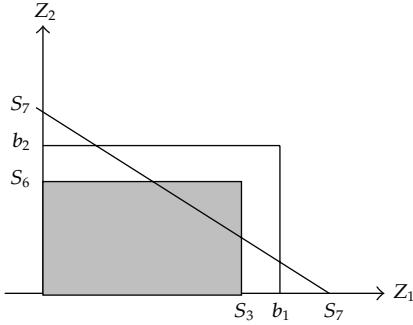
Some conclusions can be drawn from the tables.

- (i) The total number of students  $T^*$  decreases when there is commonality among the curricula.
- (ii) If the students who have taken course 7 replace the ones who should have taken courses 4 and 5, the number of common-type students is less than the number of differentiation-type students, that is,  $S_7 < S_4 + S_5$ .
- (iii) When commonality among the professional curricula increases, the number of differentiation-type students who have taken courses 3 and 6 also increases.

Since the number of supplementary students is equal to the difference between the total number of students and the expected demand, the number of supplementary students is proportional to the total number. The above comments actually describe the relationship between the number of supplementary students and the total number. When streaming is postponed, common courses must replace some professional courses. It means that some commonality exists in the curriculum. In order to satisfy industrial demand, the student recruiting policy must be flexible. To maintain the necessary overstock level, for safety, transfer students are the primary resource to replenish programs, especially in schools with high quality controls, where some students may leave for failure to fulfill the curricular requirements. If a school wants to go with the streaming postponement policy and replenishment is needed, the number of transfer students who have been streamed should be increased. In other words, if a college needs to recruit transfer students to maintain the safety stock level, it should take transfer students in their junior year. In addition, it can establish systematic guidelines to curriculum design for undeclared programs at colleges.

#### 4. Conclusion

The direction of higher education is to allow more people have opportunities to attend college. However, as the government budget for education is shrinking, the need to expand private contributions has grown. Despite pressure from a limited budget, high quality education still needs to be maintained while costs are minimized. From the economic point of view, the optimal balance among cost per student, governmental support, and self-generated



**Figure 7:** The probability of the aggregate service level.

**Table 2:** Solutions obtained by Lagrangian methods.

ASL	$T^*$	$S_3^*$	$S_6^*$
$\gamma \leq \frac{7b_2}{8b_1}$	$\sqrt{(14\gamma b_1 b_2)}$	$\frac{2}{7}\sqrt{(14\gamma b_1 b_2)}$	$\frac{2}{7}\sqrt{(14\gamma b_1 b_2)}$
$\frac{7b_2}{8b_1} \leq \gamma \leq 1 - \frac{b_2}{8b_1}$	$2\gamma b_1 + 7\frac{b_2}{4}$	$rb_1 + \frac{b_2}{8}$	$b_2$
$\gamma \geq 1 - \frac{b_2}{8b_1}$	$2(b_1 + b_2) - \sqrt{(2(1 - \gamma)b_1 b_2)}$	$b_2$	$b_2$

income would lead to effective management of resources. This is also a good index for planning and controlling the number of students enrolling in a school with undeclared status. In accordance with the current interest of undergraduates maintaining undeclared majors until their upper years and the need to maintain a certain level of professionals in training, many universities have proposed a variety of curricular programs to broaden the recruitment pool of prospective students. The proposed approach is based on the commonality among college level curricula. By decreasing the total number of students recruited from different departments, this program can reduce costs. For schools with student recruiting problems, their enrollment depends on transfer and extension students. The proposed program is more cost-effective because recruitment takes place at the college level rather than at the department level. At the same time, advanced students who have declared their majors can be recruited. In other words, students can transfer during all four years of their undergraduate program.

This paper also proposes a quantified model of commonality and recruitment planning for appropriate curriculum design. Two simple cases are used to illustrate the method, which can be divided into two phases—before undeclared students declare majors and after. However, with the challenges higher education is facing with majors, general education, and professional programs, our model still needs more research into quantity and quality. The future direction of the proposed quantification model is two-fold. First, optimization of resource allocation needs to be studied when the commonality changes depending on different types of professional training. Secondly, in order to satisfy the needs of the system, methods for controlling student distribution need to provide for transfer student quality, and appropriate and a sufficient curriculum needs to be offered accordingly.

## References

- [1] W. C. Lou, "The review of the development of Taiwanese college education in recent forty years," *The Ideal College Education*, pp. 21–36, 2004.

- [2] Ministry of Education, Taiwan, "Trend and demand in professional education for Taiwan in the 21st century," 2001.
- [3] A. Geddam, "Mechatronics for engineering education: undergraduate curriculum," *International Journal of Engineering Education*, vol. 19, no. 4, pp. 575–580, 2003.
- [4] G. Psachoropoulos and M. Woodhall, *Education for Development: An Analysis of Investment Choices*, Wu-Nan Publication, 2005, Chinese version interpreted by Bo-In Sheai.
- [5] G. Wolf, "Foundations of competency assessment & program review," *Evaluation in Education*, 2000.
- [6] C. A. Ptak, C. Smith, and J. Örlicky, *Material Requirements Planning*, McGraw-Hill, New York, NY, USA, 2011.
- [7] J. Jiao and M. M. Tseng, "Customizability analysis in design for mass customization," *CAD Computer Aided Design*, vol. 36, no. 8, pp. 745–757, 2004.
- [8] D. A. Collier, "Aggregated safety stock levels and component part commonality," *Management Science*, vol. 28, no. 11, pp. 1296–1303, 1982.
- [9] D. A. Collier, "The measurement and operating benefits of component part commonality," *Decision Science*, vol. 12, no. 1, pp. 85–96, 1991.
- [10] A. Khajavirad and J. J. Michalek, "An extension of the commonality index for product family optimization," in *Proceedings of the 33rd Design Automation Conference, presented at—2007 ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE '07)*, pp. 1001–1010, Las Vegas, Nev, USA, September 2007.
- [11] T. Blecker and N. Abdelkafi, "The development of a component commonality metric for mass customization," *IEEE Transactions on Engineering Management*, vol. 54, no. 1, pp. 70–85, 2007.
- [12] M. A. Wazed, S. Ahmed, and N. Yusoff, "Commonality and its measurement in manufacturing resources planning," *Journal of Applied Sciences*, vol. 9, no. 1, pp. 69–78, 2009.
- [13] K. R. Baker, M. J. Magazine, and H. L. W. Nuttle, "The effect of commonality on safety stock in a simple inventory model," *Management Science*, vol. 32, no. 8, 1986.

*Review Article*

## **A Review of Deterministic Optimization Methods in Engineering and Management**

**Ming-Hua Lin,<sup>1</sup> Jung-Fa Tsai,<sup>2</sup> and Chian-Son Yu<sup>1</sup>**

<sup>1</sup> Department of Information Technology and Management, Shih Chien University, No. 70, Ta-Chih Street, Taipei City 10462, Taiwan

<sup>2</sup> Department of Business Management, National Taipei University of Technology, No. 1, Section 3, Chung-Hsiao E. Road, Taipei City 10608, Taiwan

Correspondence should be addressed to Jung-Fa Tsai, jftsai@ntut.edu.tw

Received 24 February 2012; Accepted 18 April 2012

Academic Editor: Yi-Chung Hu

Copyright © 2012 Ming-Hua Lin 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.

With the increasing reliance on modeling optimization problems in practical applications, a number of theoretical and algorithmic contributions of optimization have been proposed. The approaches developed for treating optimization problems can be classified into deterministic and heuristic. This paper aims to introduce recent advances in deterministic methods for solving signomial programming problems and mixed-integer nonlinear programming problems. A number of important applications in engineering and management are also reviewed to reveal the usefulness of the optimization methods.

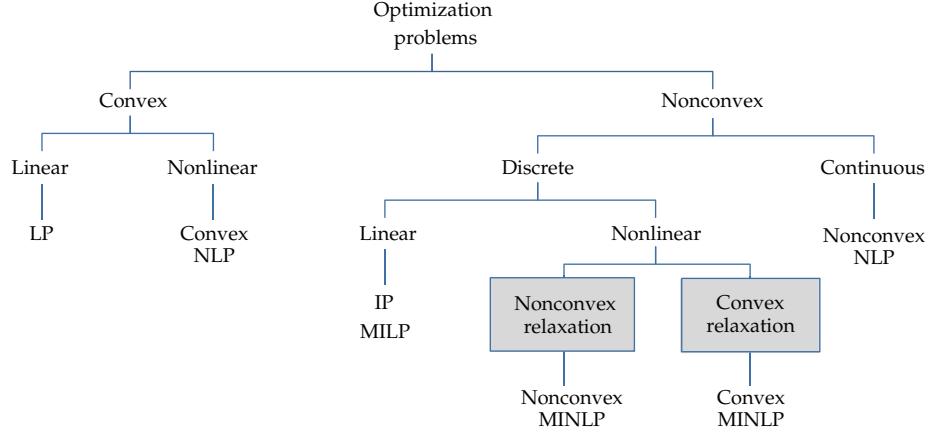
### **1. Introduction**

The field of optimization has grown rapidly during the past few decades. Many new theoretical, algorithmic, and computational contributions of optimization have been proposed to solve various problems in engineering and management. Recent developments of optimization methods can be mainly divided into deterministic and heuristic approaches. Deterministic approaches take advantage of the analytical properties of the problem to generate a sequence of points that converge to a global optimal solution. Heuristic approaches have been found to be more flexible and efficient than deterministic approaches; however, the quality of the obtained solution cannot be guaranteed. Moreover, the probability of finding the global solution decreases when the problem size increases. Deterministic approaches (e.g., linear programming, nonlinear programming, and mixed-integer nonlinear programming, etc.) can provide general tools for solving optimization problems to obtain a global or an approximately global optimum. With the increasing reliance on

modeling optimization problems in real applications, a number of deterministic methods for optimization problems have been presented. This paper focuses on discussing and reviewing the recent advances in deterministic optimization approaches.

Optimization methods have been applied in different fields, such as finance [1–3], allocation and location problems [4–6], engineering design [7–12], system and database design [13–17], chemical engineering design and control [18–22], and molecular biology [23]. For additional literature on real-world applications or developments of optimization methods, readers may refer to the following works. Mockus et al. [24] treated network problems, knapsack, travelling salesman, flow-shop problems, and batch process scheduling problems by Bayesian heuristic approaches. Grossmann [25] discussed the global optimization algorithms and their applications in chemical engineering design. Bomze et al. [26] investigated decision support systems and techniques for solving molecular structures, queuing systems, image reconstruction, location analysis, and process network synthesis problems. Migdalas et al. [27] presented multilevel optimization algorithms and their applications. Mistakidis and Stavroulakis [28] studied engineering applications of the finite element method. De Leone et al. [29] proposed various interesting applications of high-performance software for nonlinear optimization. Hendrix [30] utilized global optimization techniques on environmental management, geometric design, robust product design, and parameter estimation. Corliss and Kearfott [31] presented a rigorous global search method on industrial applications. Floudas and Pardalos [32] studied optimization approaches in the fields of computational chemistry and molecular biology. Laguna and González-Velarde [33] discussed advanced computing tools for tackling various challenging optimization problems. Papalambros and Wilde [34] investigated the principles and practice of optimal engineering design. Atkinson et al. [35] gave a detailed discussion on optimum experimental design. Edgar et al. [36] explored the optimization methods of chemical processes. Pardalos and Romeijn [37] provided a more complete and broad spectrum of approaches including deterministic and heuristic techniques for dealing with global optimization problems. Tawarmalani and Sahinidis [38] provided an insightful and comprehensive treatment of convexification and global optimization of continuous and mixed-integer nonlinear programs. Hadjisavvas et al. [39] investigated generalized convexity and generalized monotonicity and offered an advanced and broad overview of the state of the field. Moreover, Floudas et al. [40] presented an overview of the research progress in optimization during 1998–2003, including the deterministic global optimization advances in mixed-integer nonlinear programming and related applications. Pintér [41] illustrated the applicability of global optimization modeling techniques and solution strategies to real-world problems such as agroecosystem management, assembly line design, bioinformatics, biophysics, cellular mobile network design, chemical product design, composite structure design, controller design for induction motors, electrical engineering design, feeding strategies in animal husbandry, the inverse position problem in kinematics, laser design, radiotherapy planning, robot design, and satellite data analysis. Mishra and Giorgi [42] presented results on invex function, and their properties in smooth and nonsmooth cases, pseudolinearity, and eta-pseudolinearity. Mishra et al. [43] discussed the Kuhn-Tucker optimality, Karush-Kuhn-Tucker necessary and sufficient optimality conditions in presence of various types of generalized convexity assumptions. Floudas and Gounaris [44] also presented an overview of the research progress in deterministic global optimization during the last decade (1998–2008).

Figure 1 gives an overview of the problem types related to optimization problems. Each type of problems has received substantial attention from the practitioners and the



**Figure 1:** Overview of problem types related to optimization problems.

researchers in the last few decades. In this paper, we investigate the advances in deterministic global optimization of nonconvex nonlinear programming (NLP) problems and nonconvex mixed-integer nonlinear programming (MINLP) problems. For NLP problems, we focus on signomial programming problems that are an important class of NLP problems and have played a crucial role in applications.

The rest of this paper is organized as follows. Section 2 discusses the deterministic methods for signomial programming problems. Section 3 reviews the theoretical and algorithmic developments of mixed-integer nonlinear programming problems. Conclusions are made in Section 4.

## 2. Signomial Programming

Signomial programming (SP) is an optimization technique for solving a class of nonconvex nonlinear programming problems. Although SP problems occur frequently in engineering and management science, SP problems with nonconvex functions are still difficult to be solved to obtain a global optimum. The term geometric program (GP) was introduced by Duffin et al. [7] since the analysis of geometric programs relied heavily upon geometric-arithmetic mean inequality. The early work by Duffin and Peterson [45] solved the posynomial geometric program analytically via the dual problem. Then, Duffin [46] developed a numerical method for GPs based on solving a sequence of linear programs [47]. For solving SP problems, Duffin and Peterson [48] reformulated an SP problem as a geometric program with reversed constraints. The reversed constraints give rise to a nonconvex feasible region that the local minima for SP problems are not guaranteed to be global minima [49]. The developed methods for SP can be divided into two approaches. The first class of SP approaches includes various heuristic techniques such as genetic algorithms, simulated annealing, tabu search, ant colony optimization, and particle swarm optimization. Although the heuristic methods have the advantage of easy implementation and offer a better potential for complex problems, the obtained solution is not guaranteed to be a globally optimal solution. The second class of SP approaches is the deterministic method. For example, Maranas and Floudas [50], Floudas and Pardalos [51], Maranas and

Floudas [22], and Floudas [19–21] developed global optimization algorithms for solving SP problems based on the exponential variable transformation, the convex relaxation, and the branch and bound type algorithm. These methods transform the original nonconvex problem into a convex problem and then solve it to obtain the global optimum. The use of the logarithmic/exponential transformation in global optimization algorithms on SP problems restricts these exponential-based methods to handle the problems with strictly positive variables. Although positive variables are employed frequently to represent engineering and scientific systems, it is also common to introduce nonpositive variables in modeling the management problems or the system behavior, such as investment decisions, stresses, temperatures, electrical currents, velocities, and accelerations. For treating free variable  $x$ , Pörn et al. [52] suggested a simple translation,  $x + \tau = e^X$ . However, inserting the transformed result into the original signomial term will bring additional signomial terms and therefore increase the computation burden. Tsai et al. [11] proposed an approach to treat zero boundary signomial discrete programming problems and suggested some convexification rules. Li and Tsai [53], Tsai and Lin [54–56], Tsai et al. [57], Tsai [58], and Li and Lu [59] applied convexification strategies and piecewise linearization techniques to solve SP problems with free discrete/continuous variables. However, the optimal solution obtained is an approximate solution by the piecewise linearization approach. Lin and Tsai [60] presented a generalized method to solve signomial discrete programming problems with free variables for finding exactly alternative optima. Tsai and Lin [61] also integrated the convexification techniques and the bounding schemes to solve a posynomial geometric program with separable functions for finding a global optimal solution efficiently.

Convexification strategies for signomial terms are important techniques in global optimization for SP problems. With different convexification approaches, an SP problem can be reformulated into another convex program solvable to obtain an approximately global optimum. For solving SP problems, Pörn et al. [52] integrated the exponential transformation and piecewise linear approximations for reformulating nonconvex signomial problems. The results were extended by Björk [62], Björk et al. [63], and Pörn et al. [64], by including certain power transformations for convexification of nonconvex signomial terms. They discussed that the right choice of transformation for convexifying nonconvex signomial terms has a clear influence on the efficiency of the optimization approach. The concept of power convex functions is introduced to improve the solution efficiency for certain SP problems. T. Westerlund and J. Westerlund [65] proposed the generalized geometric programming extended cutting plane (GGPECP) algorithm for nonconvex optimization problems by using the cutting plane and transformation techniques. The GGPECP algorithm was described in more detail in Westerlund [66]. Lundell and Westerlund [67] and Lundell et al. [68] combined the GGPECP algorithm with an optimization framework for the transformations used to convexify the signomial terms into a signomial global optimization algorithm. The signomial global optimization algorithm was further extended by Lundell and Westerlund [69, 70]. Lin and Tsai [71] and Tsai and Lin [56] also presented similar reformulation and range reduction techniques to enhance the computational efficiency for solving SP problems.

For solving an SP problem, the above-mentioned convexification techniques are used to reformulate the original SP problem into a convex and underestimating problem solvable by a standard mixed-integer nonlinear programming (MINLP) solver [72–81]. Different transformations for positive and negative signomial terms have been proposed and discussed by Björk et al. [63], Westerlund [66], Lundell and Westerlund [67], Pörn et al. [64], Lundell et al. [68], and Lundell and Westerlund [69, 70]. For a positive signomial term  $cx_1^{\alpha_1}x_2^{\alpha_2}\cdots x_m^{\alpha_m}x_{m+1}^{\alpha_{m+1}}\cdots x_n^{\alpha_n}$  ( $c > 0, \alpha_1, \dots, \alpha_m > 0$  and  $\alpha_{m+1}, \dots, \alpha_n < 0$ ), they suggested either the

exponential transformation (ET) or the power convex transformation (PCT) is applied based on the characteristics of the problems. The ET strategy and the PCT strategy are described as follows [63].

The ET strategy:

$$cx_1^{\alpha_1}x_2^{\alpha_2}\cdots x_m^{\alpha_m}x_{m+1}^{\alpha_{m+1}}\cdots x_n^{\alpha_n} \iff \begin{cases} x_i = e^{X_i}, & i = 1, \dots, m, \\ c \frac{e^{\alpha_1 X_1 + \dots + \alpha_m X_m}}{x_{m+1}^{|\alpha_{m+1}|} \cdots x_n^{|\alpha_n|}}. \end{cases} \quad (2.1)$$

The PCT strategy: this technique aims at constructing 1-convex signomial terms. First, transform all variables with positive exponents by an inverse transformation (IT),  $x = X^{-1}$ , except the one with the greatest exponent denoted as  $\alpha_{\max}$ . Let  $S$  be defined as  $S = \sum_{i=1}^n |\alpha_i| - \alpha_{\max}$ . If  $\alpha_{\max} < S + 1$ , then transform the variable with the exponent  $\alpha_{\max}$  to that with the exponent  $S + 1$ . If  $\alpha_{\max} > S + 1$ , then change one of the ITs, with the exponent  $\alpha_j$  to  $x_j = X_j^{-\tau}$ , where  $\tau > 1$  so that  $\alpha_{\max} = S + 1 + (\tau - 1)\alpha_j$ .

Since some negative signomial terms may exist in SP problems, they suggested the potential transformation (PT) for a negative signomial term  $cx_1^{\alpha_1}x_2^{\alpha_2}\cdots x_m^{\alpha_m}x_{m+1}^{\alpha_{m+1}}\cdots x_n^{\alpha_n}$  ( $c < 0$ ,  $\alpha_1, \dots, \alpha_m > 0$  and  $\alpha_{m+1}, \dots, \alpha_n < 0$ ) expressed as follows.

The PT strategy:

$$cx_1^{\alpha_1}x_2^{\alpha_2}\cdots x_m^{\alpha_m}x_{m+1}^{\alpha_{m+1}}\cdots x_n^{\alpha_n} \iff \begin{cases} x_i = X_i^{1/R}, & i = 1, \dots, m, \\ x_i = X_i^{-1/R}, & i = m+1, \dots, n, \\ c X_1^{\alpha_1/R} \cdots X_m^{\alpha_m/R} X_{m+1}^{|\alpha_{m+1}|/R} \cdots X_n^{|\alpha_n|/R}, & R = \sum_{i=1}^n |\alpha_i|. \end{cases} \quad (2.2)$$

In addition to convexification strategies, convex envelopes and convex underestimators of nonconvex functions are frequently applied in global optimization algorithms such as the  $\alpha$ BB algorithm [19, 82, 83] to underestimate the nonconvex functions. A good convex underestimator should be as tight as possible and contain minimal number of new variables and constraints thus to improve the computational effect of processing a node in a branch-bound tree [40].

Tawarmalani and Sahinidis [84] developed the convex envelope and concave envelope for  $x/y$  over a unit hypercube, proposed a semidefinite relaxation of  $x/y$ , and suggested convex envelopes for functions of the form  $f(x)y^2$  and  $f(x)/y$ . Ryoo and Sahinidis [85] studied the use of arithmetic intervals, recursive arithmetic intervals, logarithmic transformation, and exponential transformation for multilinear functions. Tawarmalani et al. [86] studied the role of disaggregation in leading to tighter linear programming relaxations. Tawarmalani and Sahinidis [38] introduced the convex extensions for lower semicontinuous functions, proposed a technique for constructing convex envelopes for nonlinear functions, and studied the maximum separation distance for functions such as  $x/y$ . Tawarmalani et al. [87] studied 0-1 hyperbolic programs and developed eight mixed-integer convex reformulations. Liberti and Pantelides [88] proposed a nonlinear continuous and differentiable convex envelope for monomials of odd degree, derived its linear relaxation, and compared to other relaxations. Björk et al. [63] studied convexifications for signomial terms, introduced properties of power convex functions, compared the effect of the convexification schemes for heat exchanger network problems, and studied quasiconvex convexifications. Meyer and Floudas [89]

studied trilinear monomials with positive or negative domains, derived explicit expressions for the facets of the convex and concave envelopes, and showed that these outperform the previously proposed relaxations based on arithmetic intervals or recursive arithmetic intervals. Meyer and Floudas [90] presented explicit expressions for the facets of convex and concave envelopes of trilinear monomials with mixed-sign domains. Tardella [91] studied the class of functions whose convex envelope on a polyhedron coincides with the convex envelope based on the polyhedron vertices and proved important conditions for a vertex polyhedral convex envelope. Meyer and Floudas [92] described the structure of the polyhedral convex envelopes of edge-concave functions over polyhedral domains using geometric arguments and proposed an algorithm for computing the facets of the convex envelopes.

Caratzoulas and Floudas [93] proposed novel convex underestimators for trigonometric functions, which are trigonometric functions themselves. Akrotirianakis and Floudas [94, 95] introduced a new class of convex underestimators for twice continuously differentiable nonlinear programs, studied their theoretical properties, and proved that the resulting convex relaxation is improved compared to the  $\alpha$ BB one. Meyer and Floudas [90] proposed two new classes of convex underestimators for general  $C^2$  nonlinear programs, which combine the  $\alpha$ BB underestimators within a piecewise quadratic perturbation, derived properties for the smoothness of the convex underestimators, and showed the improvements over the classical  $\alpha$ BB convex underestimators for box-constrained optimization problems.

Three popular convex underestimation methods, arithmetic intervals (AIs) [96], recursive arithmetic intervals (rAIs) [50, 85, 96], and explicit facets (EFs) for convex envelopes of trilinear monomials [89, 90], are effective to underestimate a trilinear term  $x_1x_2x_3$  for  $x_i$  to be bounded variables. However, these existing methods have difficulty to treat a posynomial function. According to Ryoo and Sahinidis [85], for underestimating a multilinear function  $x_1x_2 \cdots x_n$  with  $n$  variables, the AI scheme needs to use  $\prod_{k=2}^{n-1} \Theta_k^{\binom{n}{k}} \sum_{i=1}^{\lfloor n/2 \rfloor} \binom{n}{2i}$  linear constraints maximally.  $\Theta_k$  denotes the number of linear functions that the AI scheme generates to lower bound  $k$ -cross-product terms,  $k = 2, 3, \dots, n - 1$ . Since the number of linear constraints of convex envelopes for a multilinear function with  $n$  variables grows doubly exponentially in  $n$ , AI bounding scheme may only treat  $n \leq 3$  cases. It is more difficult for AI to treat a posynomial function for  $n > 3$  cases. Moreover, applying rAI scheme to underestimate a multilinear function  $x_1x_2 \cdots x_n$  needs to use the maximum of exponentially many  $2^{n-1}$  linear inequalities. Therefore, the rAI bounding scheme has difficulty to treat posynomial functions as well as the AI scheme. EF [89, 90] provided the explicit facets of the convex and concave envelopes of trilinear monomials and demonstrated that these result in tighter bounds than the AI and rAI techniques. An important difference between EF and other bounding schemes is that these explicit facets are linear cuts, which were proven to define the convex envelope. Explicit facets (EFs) of the convex envelope are effective in treating general trilinear monomials, but the derivation of explicit facets for the convex envelope of general multilinear monomials and signomials is an open problem. Li et al. [97] and Lu et al. [98] proposed a novel method for the convex relaxation of posynomial functions. The approach is different from the work of Maranas and Floudas [50], which provided an alternative way of generating convex underestimators for generalized geometric programming problems via the exponential transformation and linear underestimation of the concave terms. Applications of this approach include the area of process synthesis and design of separations, phase equilibrium, nonisothermal complex reactor networks, and molecular conformation problems (e.g., [99–101]).

### 3. Mixed-Integer Nonlinear Programming

Mixed-integer nonlinear programming (MINLP) problems involving both continuous and discrete variables arise in many applications of engineering design, chemical engineering, operations research, and management. Biegler and Grossmann [102] provided a retrospective on optimization techniques that have been applied in process systems engineering. They indicated that design and synthesis problems have been dominated by NLP and MINLP models. With the increasing reliance on modeling optimization problems in practical problems, a number of theoretical and algorithmic contributions of MINLP have been proposed. Many deterministic methods for convex MINLP problems have been reviewed by Biegler and Grossmann [102], Grossmann [103], and Grossmann and Biegler [104]. The methods include branch-and-bound (BB) [53, 72, 77], generalized benders decomposition (GBD) [76], outer-approximation (OA) [73, 74, 79], extended cutting plane method (ECP) [81], and generalized disjunctive programming (GDP) [78]. The BB method can find the global solution only when each subproblem can be solved to global optimality. The GBD method, the OA method, and the ECP method cannot solve optimization problems with nonconvex constraints or nonconvex objective functions because the subproblems may not have a unique optimum in the solution process. The GDP models address discrete/continuous optimization problems that involve disjunctions with nonlinear inequalities and logic propositions. The objective functions and the constraints in the GDP problem are assumed to be convex and bounded [56].

For deterministic optimization methods, these optimization problems are characterized by the convexity of the feasible domain or the objective function and may involve continuous and/or discrete variables. Although continuous and discrete optimization problems constitute two classes of global optimization problems, they primarily differ in the presence or absence of convexity rather than other features. Since the convexity of the objective function or the feasible domain is very important, understanding how to convexify the nonconvex parts is an essential area of research. As long as the formulated problem is a convex problem, efficient numerical methods are available to treat the optimization problem. However, optimization problems often include nonconvex functions that cannot be dealt with by the standard local optimization techniques to guarantee global optimality efficiently. For solving nonconvex or large-scale optimization problems, deterministic methods may not be easy to derive an optimal solution within reasonable time due to the high complexity of the problem.

Sherali et al. [105] presented an extension of the reformulation linearization technique (RLT) that is designed to exploit special structures and explored the strengthening of the RLT constraints through conditional logical expressions. Sinha et al. [106] studied a solvent design problem that is constructed as a nonconvex MINLP problem. They identified the sources of nonconvexities in the properties and solubility parameter design constraints and proposed linear underestimators based on a multilevel representation approach for the functions. A reduced space branch-and-bound global optimization algorithm was then presented for solving a single component blanket wash design problem. Pörn et al. [52] introduced different convexification strategies for nonconvex MINLP problems with both posynomial and negative binomial terms in the constraints. Harjunkoski et al. [107] studied the trim loss minimization problem for the paper converting industry and formulated the model as a nonconvex MINLP. They also proposed transformations for the bilinear terms based on linear representations and convex expressions. Pörn and Westerlund [108] proposed a cutting plane method for addressing global MINLP problems with pseudoconvex objective function

and constraints and tested the proposed method on several benchmark problems arising in process synthesis and scheduling applications. Parthasarathy and El-Halwagi [109] studied a systematic framework for the optimal design of condensation, which is an important technology for volatile organic compounds, and formulated the problem as a nonconvex MINLP model. They also proposed an iterative global optimization approach based on physical insights and active constraint principles that allow for decomposition and efficient solution and applied it to a case study for the manufacture of adhesive tapes. Adjiman et al. [82, 83, 110, 111] proposed two global optimization approaches, SMIN- $\alpha$ BB and GMIN- $\alpha$ BB, for nonconvex MINLP problems based on the concept of branch-and-bound. These two approaches rely on optimization or interval-based variable-bound updates to enhance efficiency. Although one possible approach to circumvent nonconvexities in nonlinear optimization models is reformulation, for instance, using the exponential transformation to treat the generalized geometric programming problems in which a signomial term  $x_1^\alpha x_2^\beta$  is transferred into an exponential term  $e^{\alpha \ln x_1 + \beta \ln x_2}$ , the exponential transformation technique can only be applied to strictly positive variables and is thus unable to deal with nonconvex problems with free variables. Tsai et al. [11] proposed an approach to treat zero boundary optimization problems and suggested some convexification rules for the signomial terms with only three nonnegative discrete/integer variables. Björk and Westerlund [112] studied the global optimization of heat exchanger network synthesis through the simplified superstructure representation that allows only series and parallel schemes and applied convexification approaches for signomials by piecewise linear approximations. They also formulated convex MINLP lower bounding models using the Patterson formula for the log mean temperature difference considering both isothermal and nonisothermal mixing. Ostrovsky et al. [113] studied nonconvex MINLP models in which most variables are in the nonconvex terms and the number of linear constraints is much larger than the number of nonlinear constraints for solvent design and recovery problems. The work presents a tailored branch-and-bound approach using linear underestimators for tree functions based on a multilevel function representation and shows that there is a significant reduction in the branching variable space. Tawarmalani and Sahinidis [114] developed a branch and bound framework for the global optimization of MINLP problems. The framework involves novel linear relaxation schemes, a Lagrangian/linear duality-based theory for domain and range reduction, and branching strategies that guarantee finiteness of the solution sequence for certain classes of problems. They also discuss implementation issues and present computational results with a variety of benchmark problems. Kesavan et al. [115] presented outer-approximation algorithms for finding an optimal solution of a separable nonconvex MINLP program. Emet and Westerlund [116] conducted a computational comparison of solving a cyclic chromatographic separation problem using MINLP methods and reported that the extended cutting plane method compares favourably against traditional outer-approximation and branch-and-bound methods. A review of the recent advances in MINLP optimization of planning and design problems in the process industry was presented by Kallrath [117]. Tawarmalani and Sahinidis [118] introduced a polyhedral branch-and-cut approach in global optimization. Their algorithm exploits convexity in order to generate the polyhedral cuts and relaxations for multivariate nonconvex problems. Meyer and Floudas [119] studied superstructures of pooling networks, which are important to the petrochemical, chemical, and wastewater treatment industries, and formulated this generalized pooling problem as a nonconvex MINLP problem that involves many bilinear terms in the constraint functions. They proposed a global optimization algorithm based on a novel piecewise linear reformulation-linearization technique (RLT) formulation. Karuppiah and Grossmann [120]

addressed the problem of optimal synthesis of an integrated water system, where water using processes and water treatment operations are jointly considered. The designed MINLP model was solved with a new deterministic spatial branch and contract algorithm, in which piecewise under- and overestimators are used for constructing the relaxations at each node. Bergamini et al. [121] formulated an MINLP model for the global optimization of heat exchanger networks and presented a new solution methodology that is based on outer-approximation and utilizes piecewise underestimation. Rigorous constraints obtained from physical insights are also included in the formulation, and the authors reported computationally efficient global solutions for problems with up to nine process streams. Tsai and Lin [54, 56] proposed a method for solving a signomial MINLP problem with free variables by the convexification strategies and piecewise linearization techniques. However, the optimal solution obtained is an approximate solution by the piecewise linearization approach. Karuppiah et al. [122] presented an outer-approximation algorithm to globally solve a nonconvex MINLP formulation that corresponds to the continuous time scheduling of refinery crude oil operations. The solution procedure relies on effective mixed-integer linear relaxations that benefit from additional cuts derived after spatially decomposing the network. Foteinou et al. [123] presented a mixed-integer optimization framework for the synthesis and analysis of regulatory networks. Their approach integrates prior biological knowledge regarding interactions between genes and corresponding transcription factors, in an effort to minimize the complexity of the problem. Misener et al. [124] proposed an extended pooling problem to maximize the profit of blending reformulated gasoline on a predetermined network structure of feed stocks, intermediate storage tanks, and gasoline products subject to applicable environmental standards. They formulated the problem as a nonconvex MINLP model due to the presence of bilinear, polynomial, and fractional power terms. A mixed-integer linear programming relaxation of the extended pooling problem is proposed for several small- to large-scale test cases. Misener et al. [125] introduced a formulation for the piecewise linear relaxation of bilinear functions with a logarithmic number of binary variables and computationally compared their performance of this new formulation to the best performing piecewise relaxations with a linear number of binary variables. They also unified the new formulation into the computational tool APOGEE that globally optimizes standard, generalized, and extended pooling problems. Westerlund et al. [126] considered some special but fundamental issues related to convex relaxation techniques in nonconvex MINLP optimization, especially for optimization problems including nonconvex inequality constraints and their relaxations.

The alternative global optima of an MINLP problem can be found if more than one solution satisfies the same optimal value of the objective function. In practice, alternative optima are useful because they allow the decision maker to choose from many solutions without experiencing any deterioration in the objective function. For the case involving only 0-1 variables, Balas and Jeroslow [127] introduced the well-known binary cut with only one constraint and no additional variables. Duran and Grossmann [73] used this binary cut in their OA algorithm to exclude binary combinations. Tawarmalani and Sahinidis [38] mentioned that BARON can identify the  $K$  best solutions for a mixed-integer nonlinear program, where  $K$  is an option specified by the user. Tsai et al. [57] proposed a general integer cut to identify all alternative optimal solutions of a general integer linear programming problem. Lin and Tsai [60] proposed a generalized method to find multiple optimal solutions of an MINLP problem with free variables by means of variable substitution and convexification strategies. The problem is first converted into another convex MINLP problem solvable to obtain an exactly global optimum. Then, a general cut is utilized to

exclude the previous solution and an algorithm is developed to locate all alternative optimal solutions.

## 4. Conclusions

Given the rapid advances in computing technology over the past decades, large optimization theories and algorithms have been proposed to solve various real-world engineering and management problems. Therefore, to give a systematic overview of the extant literature is a challenge and motivates this study, particularly for that the field of optimization has grown and evolved rapidly. This work first reviewed methods for continuous variable optimization and survey advances in signomial programming. Then, mixed-integer nonlinear programming methods for optimization problems with discrete components were introduced. Contributions related to theoretical and algorithmic developments, formulations, and applications for these two classes of optimization problems were also discussed.

Although deterministic approaches take advantage of analytical properties of the problem to generate a sequence of points that converge to a global solution, heuristic approaches have been found to be more flexible and efficient than deterministic approaches. For solving nonconvex or large-scale optimization problems, deterministic methods may not be easy to derive a globally optimal solution within reasonable time due to the high complexity of the problem. Heuristic approaches therefore are presented to reduce the computational time of solving an optimization problem, but the obtained solution is not guaranteed to be a feasible or globally optimal solution. These two types of optimization methods have different pros and cons. Therefore, integrating deterministic and heuristic approaches may be a good way of solving large-scale optimization problems for finding a global optimum. It is hoped that this paper will stimulate further research on developing more advanced deterministic and heuristic methods to enhance the computational efficiency of finding a globally optimal solution for various real application problems.

## Acknowledgment

The research is supported by Taiwan NSC Grants NSC 99-2410-H-158-010-MY2 and NSC 99-2410-H-027-008-MY3.

## References

- [1] W. Sharpe, "A linear programming approximation for the general portfolio analysis," *Journal of Financial and Quantitative Analysis*, vol. 6, pp. 1263–1275, 1971.
- [2] B. Stone, "A linear programming formulation of the general portfolio selection model," *Journal of Financial and Quantitative Analysis*, vol. 8, pp. 621–636, 1973.
- [3] M. R. Young, "A minimax portfolio selection rule with linear programming solution," *Management Science*, vol. 44, no. 5, pp. 673–683, 1998.
- [4] C. S. Chen, S. M. Lee, and Q. S. Shen, "An analytical model for the container loading problem," *European Journal of Operational Research*, vol. 80, no. 1, pp. 68–76, 1995.
- [5] L. Faina, "A global optimization algorithm for the three-dimensional packing problem," *European Journal of Operational Research*, vol. 126, no. 2, pp. 340–354, 2000.
- [6] H. L. Li, C. T. Chang, and J. F. Tsai, "Approximately global optimization for assortment problems using piecewise linearization techniques," *European Journal of Operational Research*, vol. 140, no. 3, pp. 584–589, 2002.
- [7] R. J. Duffin, E. L. Peterson, and C. Zener, *Geometric programming: Theory and application*, John Wiley & Sons Inc., New York, 1967.

- [8] J. S. Arora, *Introduction to Optimum Design*, McGraw-Hill, New York, NY, USA, 1989.
- [9] E. Sandgren, "Nonlinear integer and discrete programming in mechanical design optimization," *Journal of Mechanisms, Transmissions, and Automation in Design*, vol. 112, no. 2, pp. 223–229, 1990.
- [10] J. F. Fu, R. G. Fenton, and W. L. Cleghorn, "A mixed integer-discrete -continuous programming method and its application to engineering design optimization," *Engineering Optimization*, vol. 17, pp. 263–280, 1991.
- [11] J. F. Tsai, H. L. Li, and N. Z. Hu, "Global optimization for signomial discrete programming problems in engineering design," *Engineering Optimization*, vol. 34, no. 6, pp. 613–622, 2002.
- [12] J. F. Tsai, "Global optimization for signomial discrete programming problems in engineering design," *Engineering Optimization*, vol. 42, no. 9, pp. 833–843, 2010.
- [13] D. Rotem, G. A. Schloss, and A. Segev, "Data allocation for multidisk databases," *IEEE Transactions on Knowledge and Data Engineering*, vol. 5, no. 5, pp. 882–887, 1993.
- [14] X. Lin and M. Orlowska, "An integer linear programming approach to data allocation with the minimum total communication cost in distributed database systems," *Information Sciences*, vol. 85, no. 1–3, pp. 1–10, 1995.
- [15] R. Sarathy, B. Shetty, and A. Sen, "A constrained nonlinear 0-1 program for data allocation," *European Journal of Operational Research*, vol. 102, no. 3, pp. 626–647, 1997.
- [16] J. F. Tsai and H. L. Li, "Technical note—on optimization approach for multidisk vertical allocation problems," *European Journal of Operational Research*, vol. 165, no. 3, pp. 835–842, 2005.
- [17] M. H. Lin, "An optimal workload-based data allocation approach for multidisk databases," *Data and Knowledge Engineering*, vol. 68, no. 5, pp. 499–508, 2009.
- [18] H. S. Ryoo and N. V. Sahinidis, "Global optimization of nonconvex NLPs and MINLPs with applications in process design," *Computers and Chemical Engineering*, vol. 19, no. 5, pp. 551–566, 1995.
- [19] C. A. Floudas, "Global optimization in design and control of chemical process systems," *Journal of Process Control*, vol. 10, no. 2, pp. 125–134, 2000.
- [20] C. A. Floudas, "Recent advances in global optimization for process synthesis, design and control: enclosure of all Solutions," *Computers and Chemical Engineering*, vol. 23, no. 1, pp. S963–S973, 1999.
- [21] C. A. Floudas, *Deterministic Global Optimization: Theory, Methods and Application*, Kluwer Academic Publishers, Boston, Mass, USA, 2000.
- [22] C. D. Maranas and C. A. Floudas, "Global optimization in generalized geometric programming," *Computers and Chemical Engineering*, vol. 21, no. 4, pp. 351–369, 1997.
- [23] J. G. Ecker, M. Kupferschmid, C. E. Lawrence, A. A. Reilly, and A. C. H. Scott, "An application of nonlinear optimization in molecular biology," *European Journal of Operational Research*, vol. 138, no. 2, pp. 452–458, 2002.
- [24] J. Mockus, W. Eddy, L. Mockus, and G. Reklaitis, *Bayesian Heuristic Approach to Discrete and Global Optimization*, Kluwer Academic Publishers, London, UK, 1996.
- [25] I.E. Grossmann, *Global Optimization in Engineering Design*, Kluwer Academic Publishers, London, UK, 1996.
- [26] I. M. Bomze, T. Csendes, R. Horst, and P. M. Pardalos, *Developments in Global Optimization*, Kluwer Academic Publishers, London, UK, 1997.
- [27] A. Migdalas, P. M. Pardalos, and P. Varbrand, *Multilevel Optimization: Algorithms and Applications*, Kluwer Academic Publishers, London, UK, 1997.
- [28] E. S. Mistakidis and G. E. Stavroulakis, *Nonconvex Optimization in Mechanics: Algorithm, Heuristics and Engineering Applications by the F.E.M*, Kluwer Academic Publishers, London, UK, 1997.
- [29] R. De Leone, A. Murli, P. M. Pardalos, and G. Toraldo, *High Performance Software for Nonlinear Optimization*, Kluwer Academic Publishers, London, UK, 1998.
- [30] E. M. T. Hendrix, *Global optimization at work*, Ph.D. thesis, Agriculture University Wageningen, Wageningen, The Netherlands, 1998.
- [31] G. F. Corliss and R. B. Kearfott, "Rigorous global search: industrial applications," in *Developments in Reliable Computing*, T. Csendes, Ed., pp. 1–16, Kluwer Academic Publishers, London, UK, 1999.
- [32] C. A. Floudas and P. M. Pardalos, *Optimization in Computational Chemistry and Molecular Biology-Local and Global Approaches*, Kluwer Academic Publishers, London, UK, 2000.
- [33] M. Laguna and J. L. González-Velarde, *Computing Tools for Modeling Optimization and Simulation*, Kluwer Academic Publishers, London, UK, 2000.
- [34] P. M. Papalambros and D. J. Wilde, *Principles of Optimal Design-modeling and Computation*, Cambridge University Press, Cambridge, UK, 2nd edition, 2000.
- [35] A. Atkinson, B. Bogacka, and A. A. Zhigljavsky, *Optimum Design 2000*, Kluwer Academic, London, UK, 2001.

- [36] T. F. Edgar, D. M. Himmelblau, and L. S. Lasdon, *Optimization of Chemical Processes*, McGraw-Hill, Boston, Mass, USA, 2nd edition, 2001.
- [37] P.M. Pardalos and H.E. Romeijn, *Handbook of Global Optimization*, vol. 2, Kluwer Academic Publishers, London, UK, 2002.
- [38] M. Tawarmalani and N. V. Sahinidis, "Convex extensions and envelopes of lower semi-continuous functions," *Mathematical Programming A*, vol. 93, no. 2, Ser. A, pp. 247–263, 2002.
- [39] N. Hadjisavvas, S. Komlósi, and S. S. Schaible, *Handbook of Generalized Convexity and Generalized Monotonicity*, Kluwer Academic Publishers, London, UK, 2005.
- [40] C. A. Floudas, I. G. Akrotirianakis, S. Caratzoulas, C. A. Meyer, and J. Kallrath, "Global optimization in the 21st century: advances and challenges," *Computers and Chemical Engineering*, vol. 29, pp. 1185–1202, 2005.
- [41] J. D. Pintér, *Global Optimization: Scientific and Engineering Case Studies*, Kluwer Academic Publishers, London, UK, 2006.
- [42] S. K. Mishra and G. Giorgi, *In vexity and Optimization*, Kluwer Academic Publishers, London, UK, 2008.
- [43] S. K. Mishra, S. Wang, and K. K. Lai, *Generalized Convexity and Vector Optimization*, Kluwer Academic Publishers, London, UK, 2009.
- [44] C. A. Floudas and C. E. Gounaris, "A review of recent advances in global optimization," *Journal of Global Optimization*, vol. 45, no. 1, pp. 3–38, 2009.
- [45] R. J. Duffin and E. L. Peterson, "Duality theory for geometric programming," *SIAM Journal on Applied Mathematics*, vol. 14, pp. 1307–1349, 1966.
- [46] R. J. Duffin, "Linearizing geometric programs," *SIAM Review*, vol. 12, pp. 211–227, 1970.
- [47] S. Boyd, S.-J. Kim, L. Vandenberghe, and A. Hassibi, "A tutorial on geometric programming," *Optimization and Engineering*, vol. 8, no. 1, pp. 67–127, 2007.
- [48] R. J. Duffin and E. L. Peterson, "Geometric programming with signomials," *Journal of Optimization Theory and Applications*, vol. 11, pp. 3–35, 1973.
- [49] J. G. Ecker, "Geometric programming: methods, computations and applications," *SIAM Review*, vol. 22, no. 3, pp. 338–362, 1980.
- [50] C. D. Maranas and C. A. Floudas, "Finding all solutions of nonlinearly constrained systems of equations," *Journal of Global Optimization*, vol. 7, no. 2, pp. 143–182, 1995.
- [51] C. A. Floudas and P. M. Pardalos, *State of the Art in Global Optimization: Computational Methods and Applications*, Kluwer Academic Publishers, Boston, MA, USA, 1996.
- [52] R. Pörn, I. Harjunkoski, and T. Westerlund, "Convexification of different classes of non-convex MINLP problems," *Computers and Chemical Engineering*, vol. 23, no. 3, pp. 439–448, 1999.
- [53] H. L. Li and J. F. Tsai, "Treating free variables in generalized geometric global optimization programs," *Journal of Global Optimization*, vol. 33, no. 1, pp. 1–13, 2005.
- [54] J. F. Tsai and M. H. Lin, "An optimization approach for solving signomial discrete programming problems with free variables," *Computers and Chemical Engineering*, vol. 30, no. 8, pp. 1256–1263, 2006.
- [55] J. F. Tsai and M.-H. Lin, "Finding all solutions of systems of nonlinear equations with free variables," *Engineering Optimization*, vol. 39, no. 6, pp. 649–659, 2007.
- [56] J. F. Tsai and M.-H. Lin, "Global optimization of signomial mixed-integer nonlinear programming problems with free variables," *Journal of Global Optimization*, vol. 42, no. 1, pp. 39–49, 2008.
- [57] J. F. Tsai, M.-H. Lin, and Y.-C. Hu, "On generalized geometric programming problems with non-positive variables," *European Journal of Operational Research*, vol. 178, no. 1, pp. 10–19, 2007.
- [58] J. F. Tsai, "Treating free variables in generalized geometric programming problems," *Computers and Chemical Engineering*, vol. 33, no. 1, pp. 239–243, 2009.
- [59] H. L. Li and H. C. Lu, "Global optimization for generalized geometric programs with mixed free-sign variables," *Operations Research*, vol. 57, no. 3, pp. 701–713, 2009.
- [60] M.-H. Lin and J. F. Tsai, "Finding multiple optimal solutions of signomial discrete programming problems with free variables," *Optimization and Engineering*, vol. 12, no. 3, pp. 425–443, 2011.
- [61] J. F. Tsai and M.-H. Lin, "An efficient global approach for posynomial geometric programming problems," *INFORMS Journal on Computing*, vol. 23, no. 3, pp. 483–492, 2011.
- [62] K. M. Björk, *A global optimization method with some heat exchanger network applications*, Ph.D. thesis, Åbo Akademi University, 2002.
- [63] K. M. Björk, P. O. Lindberg, and T. Westerlund, "Some convexifications in global optimization of problems containing signomial terms," *Computers and Chemical Engineering*, vol. 27, no. 5, pp. 669–679, 2003.

- [64] R. Pörn, K.-M. Björk, and T. Westerlund, "Global solution of optimization problems with signomial parts," *Discrete Optimization*, vol. 5, no. 1, pp. 108–120, 2008.
- [65] T. Westerlund and J. Westerlund, "GGPECP—An algorithm for solving non-convex MINLP problems by cutting plane and transformation techniques," *Chemical Engineering Transactions*, vol. 3, pp. 1045–1050, 2003.
- [66] T. Westerlund, "Some transformation techniques in global optimization," in *Global Optimization*, vol. 84, pp. 45–74, Springer, New York, NY, USA, 2006.
- [67] A. Lundell and T. Westerlund, "Optimization of power transformations in global optimization," *Chemical Engineering Transactions*, vol. 11, pp. 95–100, 2007.
- [68] A. Lundell, J. Westerlund, and T. Westerlund, "Some transformation techniques with applications in global optimization," *Journal of Global Optimization*, vol. 43, no. 2-3, pp. 391–405, 2009.
- [69] A. Lundell and T. Westerlund, "Convex underestimation strategies for signomial functions," *Optimization Methods & Software*, vol. 24, no. 4-5, pp. 505–522, 2009.
- [70] A. Lundell and T. Westerlund, "On the relationship between power and exponential transformations for positive signomial functions," *Chemical Engineering Transactions*, vol. 17, pp. 1287–1292, 2009.
- [71] M.-H. Lin and J. F. Tsai, "Range reduction techniques for improving computational efficiency in global optimization of signomial geometric programming problems," *European Journal of Operational Research*, vol. 216, no. 1, pp. 17–25, 2012.
- [72] B. Borchers and J. E. Mitchell, "An improved branch and bound algorithm for mixed integer nonlinear programs," *Computers & Operations Research*, vol. 21, no. 4, pp. 359–367, 1994.
- [73] M. A. Duran and I. E. Grossmann, "An outer-approximation algorithm for a class of mixed-integer nonlinear programs," *Mathematical Programming*, vol. 36, no. 3, pp. 307–339, 1986.
- [74] R. Fletcher and S. Leyffer, "Solving mixed integer nonlinear programs by outer approximation," *Mathematical Programming*, vol. 66, no. 3, Ser. A, pp. 327–349, 1994.
- [75] C. A. Floudas, *Nonlinear and Mixed Integer Optimization:Fundamentals and Applications*, Oxford University Press, New York, NY, USA, 1995.
- [76] A. M. Geoffrion, "Generalized Benders decomposition," *Journal of Optimization Theory and Applications*, vol. 10, pp. 237–260, 1972.
- [77] S. Leyffer, "Integrating SQP and branch-and-bound for mixed integer nonlinear programming," *Computational Optimization and Applications*, vol. 18, no. 3, pp. 295–309, 2001.
- [78] S. Lee and I. E. Grossmann, "New algorithms for nonlinear generalized disjunctive programming," *Computers and Chemical Engineering*, vol. 24, no. 9-10, pp. 2125–2142, 2000.
- [79] I. Quesada and I. E. Grossmann, "An LP/NLP based branch and bound algorithm for convex MINLP optimization problems," *Computers and Chemical Engineering*, vol. 16, no. 10-11, pp. 937–947, 1992.
- [80] R. A. Stubbs and S. Mehrotra, "A branch-and-cut method for 0-1 mixed convex programming," *Mathematical Programming A*, vol. 86, no. 3, pp. 515–532, 1999.
- [81] T. Westerlund and F. Pettersson, "An extended cutting plane method for solving convex MINLP problems," *Computers and Chemical Engineering*, vol. 19, no. 1, supplement, pp. S131–S136, 1995.
- [82] C. S. Adjiman, S. Dallwig, C. A. Floudas, and A. Neumaier, "A global optimization method,  $\alpha$ BB, for general twice-differentiable constrained NLPs-I. Theoretical advances," *Computers and Chemical Engineering*, vol. 22, no. 9, pp. 1137–1158, 1998.
- [83] C. S. Adjiman, I. P. Androulakis, and C. A. Floudas, "A global optimization method,  $\alpha$ BB, for general twice-differentiable constrained NLPs-II. Implementation and computational results," *Computers and Chemical Engineering*, vol. 22, no. 9, pp. 1159–1179, 1998.
- [84] M. Tawarmalani and N. V. Sahinidis, "Semidefinite relaxations of fractional programs via novel convexification techniques," *Journal of Global Optimization*, vol. 20, no. 2, pp. 137–158, 2001.
- [85] H. S. Ryoo and N. V. Sahinidis, "Analysis of bounds for multilinear functions," *Journal of Global Optimization*, vol. 19, no. 4, pp. 403–424, 2001.
- [86] M. Tawarmalani, S. Ahmed, and N. V. Sahinidis, "Product disaggregation in global optimization and relaxations of rational programs," *Optimization and Engineering*, vol. 3, no. 3, pp. 281–303, 2002.
- [87] M. Tawarmalani, S. Ahmed, and N. V. Sahinidis, "Global optimization of 0-1 hyperbolic programs," *Journal of Global Optimization*, vol. 24, no. 4, pp. 385–416, 2002.
- [88] L. Liberti and C. C. Pantelides, "Convex envelopes of monomials of odd degree," *Journal of Global Optimization*, vol. 25, no. 2, pp. 157–168, 2003.
- [89] C. A. Meyer and C. A. Floudas, "Trilinear Monomials with Positive or Negative Domains: Facets of Convex and Concave Envelopes," in *Frontiers in global optimization*, C. A. Floudas and P. M. Pardalos, Eds., pp. 327–352, Kluwer Academic Publishers, Santorini, Greece, 2003.

- [90] C. A. Meyer and C. A. Floudas, "Trilinear monomials with mixed sign domains: facets of the convex and concave envelopes," *Journal of Global Optimization*, vol. 29, no. 2, pp. 125–155, 2004.
- [91] F. Tardella, "On the existence of polyhedral convex envelopes," in *Frontiers in Global Optimization*, C. A. Floudas and P. M. Pardalos, Eds., pp. 563–573, Kluwer Academic Publishers, Santorini, Greece, 2003.
- [92] C. A. Meyer and C. A. Floudas, "Convex envelopes for edge-concave functions," *Mathematical Programming B*, vol. 103, no. 2, pp. 207–224, 2005.
- [93] S. Caratzoulas and C. A. Floudas, "Trigonometric convex underestimator for the base functions in Fourier space," *Journal of Optimization Theory and Applications*, vol. 124, no. 2, pp. 339–362, 2005.
- [94] I. G. Akrotirianakis and C. A. Floudas, "Computational experience with a new class of convex underestimators: box-constrained NLP problems," *Journal of Global Optimization*, vol. 29, no. 3, pp. 249–264, 2004.
- [95] I. G. Akrotirianakis and C. A. Floudas, "A new class of improved convex underestimators for twice continuously differentiable constrained NLPs," *Journal of Global Optimization*, vol. 30, no. 4, pp. 367–390, 2004.
- [96] A. S. E. Hamed, *Calculation of bounds on variables and underestimating convex functions for nonconvex functions*, Ph.D. thesis, The George Washington University, 1991.
- [97] H. L. Li, J. F. Tsai, and C. A. Floudas, "Convex underestimation for posynomial functions of positive variables," *Optimization Letters*, vol. 2, no. 3, pp. 333–340, 2008.
- [98] H. C. Lu, H. L. Li, C. E. Gounaris, and C. A. Floudas, "Convex relaxation for solving posynomial programs," *Journal of Global Optimization*, vol. 46, no. 1, pp. 147–154, 2010.
- [99] A. Aggarwal and C. A. Floudas, "Synthesis of general distillation sequences-nonsharp separations," *Computers and Chemical Engineering*, vol. 14, no. 6, pp. 631–653, 1990.
- [100] A. R. Ceric and C. A. Floudas, "A retrofit approach for heat exchanger networks," *Computers and Chemical Engineering*, vol. 13, no. 6, pp. 703–715, 1989.
- [101] A. C. Kokossis and C. A. Floudas, "Optimization of complex reactor networks-II. Nonisothermal operation," *Chemical Engineering Science*, vol. 49, no. 7, pp. 1037–1051, 1994.
- [102] L. T. Biegler and I. E. Grossmann, "Retrospective on optimization," *Computers and Chemical Engineering*, vol. 28, no. 8, pp. 1169–1192, 2004.
- [103] I. E. Grossmann, "Review of nonlinear mixed-integer and disjunctive programming techniques," *Optimization and Engineering*, vol. 3, no. 3, pp. 227–252, 2002.
- [104] I. E. Grossmann and L. T. Biegler, "Part II. Future perspective on optimization," *Computers and Chemical Engineering*, vol. 28, no. 8, pp. 1193–1218, 2004.
- [105] H. D. Sherali, W. P. Adams, and P. J. Driscoll, "Exploiting special structures in constructing a hierarchy of relaxations for mixed integer problems," *Operations Research*, vol. 46, no. 3, pp. 396–405, 1998.
- [106] M. Sinha, L. E. K. Achenie, and G. M. Ostrovsky, "Environmentally benign solvent design by global optimization," *Computers and Chemical Engineering*, vol. 23, no. 10, pp. 1381–1394, 1999.
- [107] I. Harjunkoski, T. Westerlund, and R. Pörn, "Numerical and environmental considerations on a complex industrial mixed integer non-linear programming (MINLP) problem," *Computers and Chemical Engineering*, vol. 23, no. 10, pp. 1545–1561, 1999.
- [108] R. Pörn and T. Westerlund, "A cutting plane method for minimizing pseudo-convex functions in the mixed integer case," *Computers and Chemical Engineering*, vol. 24, no. 12, pp. 2655–2665, 2000.
- [109] G. Parthasarathy and M. M. El-Halwagi, "Optimum mass integration strategies for condensation and allocation of multicomponent VOCs," *Chemical Engineering Science*, vol. 55, no. 5, pp. 881–895, 2000.
- [110] C. S. Adjiman, I. P. Androulakis, and C. A. Floudas, "Global optimization of MINLP problems in process synthesis and design," *Computers and Chemical Engineering*, vol. 21, no. 1, pp. S445–S450, 1997.
- [111] C. S. Adjiman, I. P. Androulakis, and C. A. Floudas, "Global optimization of mixed-integer nonlinear problems," *AICHE Journal*, vol. 46, no. 9, pp. 1769–1797, 2000.
- [112] K. M. Björk and T. Westerlund, "Global optimization of heat exchanger network synthesis problems with and without the isothermal mixing assumption," *Computers and Chemical Engineering*, vol. 26, no. 11, pp. 1581–1593, 2002.
- [113] G. M. Ostrovsky, L. E. K. Achenie, and M. Sinha, "On the solution of mixed-integer nonlinear programming models for computer aided molecular design," *Computers and Chemistry*, vol. 26, no. 6, pp. 645–660, 2002.

- [114] M. Tawarmalani and N. V. Sahinidis, "Global optimization of mixed-integer nonlinear programs: a theoretical and computational study," *Mathematical Programming A*, vol. 99, no. 3, pp. 563–591, 2004.
- [115] P. Kesavan, R. J. Allgor, E. P. Gatzke, and P. I. Barton, "Outer approximation algorithms for separable nonconvex mixed-integer nonlinear programs," *Mathematical Programming A*, vol. 100, no. 3, pp. 517–535, 2004.
- [116] S. Emet and T. Westerlund, "Comparisons of solving a chromatographic separation problem using MINLP methods," *Computers and Chemical Engineering*, vol. 28, no. 5, pp. 673–682, 2004.
- [117] J. Kallrath, "Solving planning and design problems in the process industry using mixed integer and global optimization," *Annals of Operations Research*, vol. 140, pp. 339–373, 2005.
- [118] M. Tawarmalani and N. V. Sahinidis, "A polyhedral branch-and-cut approach to global optimization," *Mathematical Programming B*, vol. 103, no. 2, pp. 225–249, 2005.
- [119] C. A. Meyer and C. A. Floudas, "Global optimization of a combinatorially complex generalized pooling problem," *AICHE Journal*, vol. 52, no. 3, pp. 1027–1037, 2006.
- [120] R. Karuppiah and I. E. Grossmann, "Global optimization for the synthesis of integrated water systems in chemical processes," *Computers and Chemical Engineering*, vol. 30, no. 4, pp. 650–673, 2006.
- [121] M. L. Bergamini, N. J. Scenna, and P. A. Aguirre, "Global optimal structures of heat exchanger networks by piecewise relaxation," *Industrial and Engineering Chemistry Research*, vol. 46, no. 6, pp. 1752–1763, 2007.
- [122] R. Karuppiah, K. C. Furman, and I. E. Grossmann, "Global optimization for scheduling refinery crude oil operations," *Computers and Chemical Engineering*, vol. 32, no. 11, pp. 2745–2766, 2008.
- [123] P. T. Foteinou, E. Yang, G. K. Saharidis, M. G. Ierapetritou, and I. P. Androulakis, "A mixed-integer optimization framework for the synthesis and analysis of regulatory networks," *Journal of Global Optimization*, vol. 43, no. 2-3, pp. 263–276, 2009.
- [124] R. Misener, C. E. Gounaris, and C. A. Floudas, "Mathematical modeling and global optimization of large-scale extended pooling problems with the (EPA) complex emissions constraints," *Computers and Chemical Engineering*, vol. 34, no. 9, pp. 1432–1456, 2010.
- [125] R. Misener, J. P. Thompson, and C. A. Floudas, "Apogee: global optimization of standard, generalized, and extended pooling problems via linear and logarithmic partitioning schemes," *Computers and Chemical Engineering*, vol. 35, no. 5, pp. 876–892, 2011.
- [126] T. Westerlund, A. Lundelf, and J. Westerlund, "On convex relaxations in nonconvex optimization," *Chemical Engineering Transactions*, vol. 24, pp. 331–336, 2011.
- [127] E. Balas and R. Jeroslow, "Canonical cuts on the unit hypercube," *SIAM Journal on Applied Mathematics*, vol. 23, pp. 61–69, 1972.

*Research Article*

## A Hybrid Genetic Algorithm for the Multiple Crossdocks Problem

**Zhaowei Miao,<sup>1</sup> Ke Fu,<sup>2</sup> and Feng Yang<sup>1</sup>**

<sup>1</sup> School of Management, Xiamen University, Xiamen 361005, China

<sup>2</sup> Lingnan (University) College, Sun Yat-Sen University, Guangzhou 510275, China

Correspondence should be addressed to Ke Fu, fuke@mail.sysu.edu.cn

Received 8 December 2011; Revised 1 April 2012; Accepted 15 April 2012

Academic Editor: John Gunnar Carlsson

Copyright © 2012 Zhaowei Miao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We study a multiple crossdocks problem with supplier and customer time windows, where any violation of time windows will incur a penalty cost and the flows through the crossdock are constrained by fixed transportation schedules and crossdock capacities. We prove this problem to be  $\mathcal{NP}$ -hard in the strong sense and therefore focus on developing efficient heuristics. Based on the problem structure, we propose a hybrid genetic algorithm (HGA) integrating greedy technique and variable neighborhood search method to solve the problem. Extensive experiments under different scenarios were conducted, and results show that HGA outperforms CPLEX solver, providing solutions in realistic timescales.

### 1. Introduction

As companies seek more profitable supply chains, there has been a desire to optimize distribution networks to reduce logistics costs. This includes finding the best locations for facilities, minimizing inventories, and minimizing transportation costs. A distinct recent industry example is the successful implementation of crossdocking strategy at Wal-Mart, whose crossdocks require coordinating 2000 dedicated trucks over a large network of warehouses, crossdocks, and retail points [1]. While there is a rich literature on conventional facility location problems, crossdocking strategies—which minimize inventory by processing goods quickly for reshipment—have recently attracted the attention of researchers (see, e.g., [2–4]). In conventional transshipment-inventory models, a common assumption is that demand (usually stochastic) that cannot be met from one supply point can be fulfilled through some other point. The objective is then to evaluate a control policy for replenishment. Work on this subject has been extensive and can be found in, for example, Krishnan and Rao [5], Karmarkar and Patel [6], Karmarkar [7], Tagaras [8], Robinson [9], Rudi et al. [10],

Grahovac and Chakravarty [11], Herer and Tzur [12], Herer et al. [13], Axsäter [14], and Axsäter [15]. For the general  $n$  location transshipment model, heuristics were proposed by Robinson [9] who developed a large-scale LP by discretizing demand. Although these studies considered inventory and transshipment costs, they did not address time constraints that occur during the transshipment process, for example, constraints imposed by transportation schedules. There has been relatively limited research on distribution and system design, which includes crossdocks. Some recent attempts can be found in Donaldson et al. [16], Ratcliff et al. [17], Gumus and Bookbinder [4], Li et al. [3], Miao et al. [18], and Boysen et al. [19]. In particular, Gumus and Bookbinder [4] modeled location-distribution networks that include crossdock facilities to determine the impact on the supply chain. Li et al. [3] developed a heuristic algorithm to find JIT schedules within a single crossdock. Miao et al. [18] and Boysen et al. [19] studied how to schedule the inbound and outbound trucks to achieve high operational efficiency within a single crossdock.

Our work differs from the above research in that we study a kind of multiple crossdocks problem where transportation is available at fixed schedules, and where both shipping and delivery at supply and demand locations can be executed within specified time windows with normal transportation cost, and any shipment that cannot be met will be fulfilled through external channels, which causes penalty costs. Supplier time windows allow, for example, flexibility in planning for the best shipping times to fit production and operating schedules. Time windows at demand points satisfy customer requirements, for example, when service deadlines must be met. Moreover, in the real-world applications, sometimes the time windows are allowed to be violated. There are two primary reasons for this: one is exogenous, for example, it might not be practical to satisfy all the time windows constraints when the demands are too high during a certain period; the other is endogenous, for example, some shipments are emergent and have to be executed outside the normal time windows. Clearly, such abnormal arrangements usually incur additional costs we call penalty in this paper. Furthermore, we consider inventory cost at the crossdocks, which includes storing cost and handling cost and are one of the cost sources in any transshipment strategy.

To the best of our knowledge, there are some papers closest to our work, including Lim et al. [20], Chen et al. [21], and Ma et al. [22]. Lim et al. [20] studied the complexity of different types of multiple crossdocks problems where transportation schedules such as flexible schedules and fixed schedules, and time constraints at manufacturers and customers are included. Our model extends one of the cases studied by them, which is called single shipping and single delivery by fixed schedules. However, their problem requires all the demands should be satisfied by the suppliers, which is very difficult to achieve even to find a feasible solution. We relax this constraint and allow demands of some customers to be unfulfilled, but penalty cost will be incurred. In addition, most importantly, in the earlier works, they did not set up any mathematical formulation and provide any implementable algorithms that were able to solve this practical problem. We model this problem as an integer programming problem, prove it to be  $\mathcal{NP}$ -hard in the strong sense, and then focus on developing efficient heuristic algorithms. Chen et al. [21] extended another case studied by Lim et al. [20], which is called single shipping and single delivery by flexible schedules, simplified the problem by discretizing the time horizon, and designed metaheuristic algorithms to solve it. Despite the constraint of single shipping and single delivery, Ma et al. [22] took into consideration setup cost of each vehicle in their multiple crossdocks problem, which is also strong  $\mathcal{NP}$ -hard and solved by a two-stage heuristic algorithm developed by them.

While one of the main objectives in this paper is to develop an efficient heuristic algorithm to solve the above multiple crossdocks problem, we find that numerous researches

have been dedicated to design meta-heuristic algorithms to solve transportation problems. For example, Hai and lim [23] used a Tabu-embedded simulated annealing algorithm that restarted from the current best solution after several nonimproving iterations to solve the pickup and delivery problem with time windows. Chen et al. [21] developed three kinds of meta-heuristics, namely, simulated annealing, tabu search, and integrated simulated annealing and tabu search with local search technique to solve multiple crossdocks problems with inventory and time windows. In a dynamic vehicle dispatching problem with pickups and deliveries, Michel et al. [24] proposed a tabu search with neighborhood search based on ejection chains to explore the proposed problem. All these meta-heuristic algorithms are focused on two aspects, one is how to generate initial feasible solutions efficiently according to the specific structure of the problems, and the other is how to improve the current best solution. Based on this, neighborhood search and integrated meta-heuristic algorithm are then developed. In this paper, we adopt the basic idea of the genetic algorithm and meanwhile take advantage of the problem structure to develop a hybrid genetic algorithm (HGA) integrating greedy technique and variable neighborhood search to solve the proposed problem. The unique feature of HGA is that it makes assignment of crossdocks and routes to suppliers and customers in *two stages* and is further refined by the greedy technique and variable neighborhood search. Since the problem we consider is a fairly general transshipment problem with complicated constraints, we believe that our method could be useful to solve other transshipment problems of this sort. We conduct various computational experiments with different problem scales, and the results show that the proposed HGA can yield better solutions for various problem instances of different scales, especially for the large-sized problems compared with the CPLEX solver, which gives solutions before getting terminated within the stipulated time limit of execution.

The rest of this paper is organized as follows. In Section 2, we formulate the problem as an integer programming problem and provide complexity analysis. In Section 3, we explore the problem structure and develop HGA. Section 4 demonstrates HGA with computational results for various problem instances of different scales. The paper is concluded in Section 5.

## 2. The Multiple Crossdocks Problem

In this section, we describe the multiple crossdocks problem and introduce basic notation in Section 2.1. We then formally formulate the problem as an integer programming problem and provide complexity analysis in Section 2.2.

### 2.1. Problem Description

The problem studied here extends the well-known transshipment problem to include constraints imposed by time and inventory considerations, which arise in applications. The following assumptions are made. First, a supplier (customer) is allowed to ship goods to crossdocks (receive goods from crossdocks) within a specified time window with a normal cost level; however, if the shipment cannot be met, a much higher cost called penalty here is incurred. When the shipments take place outside the time windows, then it means that the current transportation network is unable or too busy to fulfill this shipment requirement and external channels have to be used to ship the cargos at a much higher cost, which may include the higher transportation costs and the order earliness or lateness costs through the external channels. Second, shipped goods can be delayed at crossdocks, which is helpful to

satisfy time windows constraints and also helpful to potential consolidation. Third, shipping schedules offered by transportation providers are fixed, that is, departure and arrival times of any schedule are fixed. For example, the schedules in the railway network or in airline operations are usually fixed. We assume that each schedule has a set of associated shipping costs and route capacities. Fourth, the setup cost of each shipment sometimes is very high in real world, so in order to reduce setup cost as much as possible, it requires each supplier make only one batch shipment to some crossdock within its specified time window, and each customer can receive goods only one time from some crossdock within its time window, which is called single shipping and single delivery case [20], and for this case, consolidation is quite important. Finally, the objective of our problem is to satisfy the demands of the customers with minimum total costs including shipping cost, inventory cost and penalty cost without violating the capacity constraints of crossdocks and routes through given fixed schedules.

The underlying problem can be represented by a network. Let  $\Sigma := \{1, \dots, n\}$  be the set of supply nodes (suppliers) where, for each  $i \in \Sigma$ ,  $s_i$  units of goods are available, which can be shipped (released) in the time window  $[b_i^r, e_i^r]$ ,  $\Delta := \{1, \dots, m\}$  the set of demand nodes (customers) where each  $k \in \Delta$  requires  $d_k$  units of goods, which must be delivered (accepted) within the time window  $[b_k^a, e_k^a]$ , and  $\mathbf{X} := \{1, \dots, l\}$  the set of crossdocks, where each  $j \in \mathbf{X}$  has inventory capacity  $c_j$  and inventory cost  $h_j$  per unit per time. Take  $S_1$  to denote all fixed scheduled routes between points in  $\Sigma$  and points in  $\mathbf{X}$ , that is, routes serviced by transport providers, each with a scheduled departure (begin) and arrival (end) time, capacity and unit transportation cost. Similarly, let  $S_2$  denote the set of fixed schedules between the crossdocks  $\mathbf{X}$  and customers  $\Delta$ .

## 2.2. Mathematical Formulation

We now introduce more notation that is used in our formulation as follows

$S_{i,j}$ : set of fixed transportation schedules between supplier  $i$  and crossdock  $j$ , and  $|S_{i,j}| = \gamma_{i,j}$ , where  $|\cdot|$  represents the cardinality of a set.

$S'_{j,k}$ : set of fixed transportation schedules between crossdock  $j$  and customer  $k$ , and  $|S'_{j,k}| = \gamma'_{j,k}$ .

$(b_{i,j,q}^r, e_{i,j,q}^r)$ :  $q$ th fixed transportation schedule in  $S_{i,j}$ , where  $b_{i,j,q}^r$  and  $e_{i,j,q}^r$  are the beginning time point and ending time point of this fixed schedule, respectively.

$(b_{j,k,q}^a, e_{j,k,q}^a)$ :  $q$ th fixed transportation schedule in  $S'_{j,k}$ , where  $b_{j,k,q}^a$  and  $e_{j,k,q}^a$  are the beginning time point and ending time point of this fixed schedule, respectively.

$c_{i,j,q}$ : unit shipping cost from supplier  $i$  to crossdock  $j$  through  $q$ th fixed transportation schedule in  $S_{i,j}$ .

$c'_{j,k,q}$ : unit shipping cost from crossdock  $j$  to customer  $k$  through  $q$ th fixed transportation schedule in  $S'_{j,k}$ .

$P_i$ : unit penalty cost for supplier  $i$  if its cargo cannot be shipped out.

$P'_k$ : unit penalty cost for customer  $k$  if its demand cannot be met.

$T_j$ : set of ending time points of all fixed transportation schedules in  $\cup_{i=1}^n S_{i,j}$ , that is,  $T_j = \{e_{i,j,q}^r : 1 \leq i \leq n, 1 \leq q \leq \gamma_{i,j}\}$ .

$T'_j$ : set of beginning time points of all fixed transportation schedules in  $\cup_{k=1}^m S'_{j,k}$ , that is,  $T'_j = \{b_{j,k,q}^a : 1 \leq k \leq m, 1 \leq q \leq \gamma'_{j,k}\}$ .

$\tilde{T}_j$ : set of time points when the inventory level of crossdock  $j$  is likely to be changed, that is,  $\tilde{T}_j = T_j \cup T'_j$ . Let  $|\tilde{T}_j| = \tau_j$  and all the elements in  $\tilde{T}_j$  are sorted in an increasing order, and let  $t_{j,g}$  ( $g = 1, 2, \dots, \tau_j$ ) correspond to these  $\tau_j$  time points such that  $t_{j,1} \leq t_{j,2} \leq \dots \leq t_{j,\tau_j}$ . Using this notation, we can easily formulate the set of flow conservation constraints later.

$CAP_{i,j,q}$ : shipping capacity of  $q$ th fixed transportation schedule in  $S_{i,j}$ .

$CAP'_{j,k,q}$ : shipping capacity of  $q$ th fixed transportation schedule in  $S'_{j,k}$ .

$\theta_{i,j,q}$ : a binary parameter, which is 0 if the beginning time point of  $q$ th fixed transportation schedule in  $S_{i,j}$  is within the time window of supplier  $i$ , that is,  $b_{i,j,q}^r \in [b_i^r, e_i^r]$ , and 1 otherwise.

$\theta'_{j,k,q}$ : a binary parameter, which is 0 if the ending time point of  $q$ th fixed transportation schedule in  $S'_{j,k}$  is within the time window of customer  $k$ , that is,  $e_{j,k,q}^a \in [b_k^a, e_k^a]$ , and 1 otherwise.

The following are decision variables.

$x_{i,j,q}$ : binary, which is 1 if to deliver cargos from supplier  $i$  is bound for crossdock  $j$  through  $q$ th fixed transportation schedule in  $S_{i,j}$ , and 0 otherwise.

$x'_{j,k,q}$ : binary, which is 1 if to receive cargos from crossdock  $j$  to customer  $k$  is through  $q$ th fixed transportation schedule in  $S'_{j,k}$ , and 0 otherwise.

$y_{j,t_{j,g}}$ : integer, which is inventory level in crossdock  $j$  at time  $t_{j,g}$ , where  $t_{j,g} \in \tilde{T}_j$ .

We are now ready to formulate the transshipment problem, which hereafter is called problem (P):

$$\min \text{COST}_{\text{Transportation}} + \text{COST}_{\text{Penalty}} + \text{COST}_{\text{Inventory}}, \quad (\text{P})$$

where

$$\begin{aligned} \text{COST}_{\text{Transportation}} &= \sum_{i=1}^n \sum_{j=1}^l \sum_{q=1}^{\gamma_{i,j}} c_{i,j,q} s_i x_{i,j,q} + \sum_{k=1}^m \sum_{j=1}^l \sum_{q=1}^{\gamma'_{j,k}} c'_{j,k,q} d_k x'_{j,k,q}, \\ \text{COST}_{\text{Penalty}} &= \sum_{i=1}^n P_i s_i \left( 1 - \sum_{j=1}^l \sum_{q=1}^{\gamma_{i,j}} x_{i,j,q} \right) + \sum_{k=1}^m P'_k d_k \left( 1 - \sum_{j=1}^l \sum_{q=1}^{\gamma'_{j,k}} x'_{j,k,q} \right), \\ \text{COST}_{\text{Inventory}} &= \sum_{j=1}^l \sum_{g=1}^{\tau_j} h_j (t_{j,g} - t_{j,g-1}) y_{j,t_{j,g}}, \end{aligned} \quad (2.1)$$

s.t.

$$x_{i,j,q} \leq 1 - \theta_{i,j,q} \quad (1 \leq i \leq n, 1 \leq j \leq l, 1 \leq q \leq \gamma_{i,j}), \quad (2.2)$$

$$x'_{j,k,q} \leq 1 - \theta'_{j,k,q} \quad (1 \leq j \leq l, 1 \leq k \leq m, 1 \leq q \leq \gamma'_{j,k}), \quad (2.3)$$

$$\sum_{j=1}^l \sum_{q=1}^{\gamma_{i,j}} x_{i,j,q} \leq 1 \quad (1 \leq i \leq n), \quad (2.4)$$

$$\sum_{j=1}^l \sum_{q=1}^{\gamma'_{j,k}} x'_{j,k,q} \leq 1 \quad (1 \leq k \leq m) \quad (2.5)$$

$$\begin{aligned} s_i x_{i,j,q} &\leq \text{CAP}_{i,j,q} \quad (1 \leq i \leq n, 1 \leq j \leq l, 1 \leq q \leq \gamma_{i,j}), \\ d_k x'_{j,k,q} &\leq \text{CAP}'_{j,k,q} \quad (1 \leq k \leq m, 1 \leq j \leq l, 1 \leq q \leq \gamma'_{j,k}), \end{aligned} \quad (2.6)$$

$$\begin{aligned} y_{j,t_{j,g}} &\leq c_j (1 \leq j \leq l, 1 \leq g \leq \tau_j), \\ y_{j,t_{j,0}} &= 0 (1 \leq j \leq l, t_{j,0} = 0), \end{aligned} \quad (2.7)$$

$$y_{j,t_{j,g}} = y_{j,t_{j,g-1}} + \sum_{i=1}^n \sum_{\{q: e_{i,j,q}^r = t_{j,g}\}} s_i x_{i,j,q} - \sum_{k=1}^m \sum_{\{q: b_{j,k,q}^a = t_{j,g}\}} d_k x'_{j,k,q} \quad (1 \leq j \leq l, 1 \leq g \leq \tau_j), \quad (2.8)$$

$$\begin{aligned} x_{i,j,q} &\in \{0, 1\} \quad (1 \leq i \leq n, 1 \leq j \leq l, 1 \leq q \leq \gamma_{i,j}), \\ x'_{j,k,q} &\in \{0, 1\} \quad (1 \leq j \leq l, 1 \leq k \leq m, 1 \leq q \leq \gamma'_{j,k}), \\ y_{j,t_{j,g}} &\in \mathbb{N} \quad (1 \leq j \leq l, 1 \leq g \leq \tau_j). \end{aligned} \quad (2.9)$$

In the above formulation, the objective is to minimize total cost, including transportation cost, penalty cost, and inventory cost. Note that we impose the penalty cost on both supplier and customer sides here because unfulfilled demands have different impact on each side in general. Constraint (2.2) ensures that each available fixed transportation schedule is within the time window of suppliers. Similarly, the available fixed transportation schedule of customers is given by (2.3). Constraint (2.4) ensures that each delivery is fulfilled within each supplier specified time window at most once and (2.5) forces each customer to receive cargos within its time window for no more than one time, which is required by single shipping and single delivery constraint. The capacity constraints of fixed schedules are given by (2.6). The capacity constraint of every crossdock is restricted by (2.7) and we also set a zero initial inventory for each crossdock. The changes of inventory level of each crossdock are recorded in (2.8), which ensures cargo flow conservation.

We have the following proposition whose proof is given in the appendix.

**Proposition 2.1.** *The multiple crossdocks problem (P) is NP-hard in the strong sense, even if supply and demand time windows and crossdock and route capacities are relaxed.*

From the above proposition, we know that to find minimum cost of this problem is NP-hard in the strong sense. Hence, it is unlikely to find a polynomial or pseudopolynomial time algorithm to solve the problem unless P=NP. As a result, we focus on efficient heuristics

to solve problem (P). In the next section, we describe a heuristic that exploits the problem structure and solves the problem efficiently.

### 3. Hybrid Genetic Algorithm

Genetic algorithm (GA) has become a well-known and powerful metaheuristic approach for hard combinatorial optimization problems. Genetic algorithm is based on the ideas of natural selection and has been applied to numerous combinatorial optimization problems successfully. However, basic genetic algorithms have limitations to attain the optimal solution. We propose a hybrid genetic algorithm (HGA) integrating greedy technique and variable neighborhood search to solve problem (P) as described in the previous section. The proposed HGA simulates the natural selection process; in addition, it incorporates the special structure of problem (P). In particular, in problem (P), we expect to assign crossdocks and routes to *both the suppliers and customers* in the most cost-effective manner. Clearly, the assignments of crossdocks and routes for suppliers will affect the assignments for customers due to capacity and time windows constraints and vice versa. As a result, simultaneously assigning crossdocks and routes for suppliers and customers is not effective. Observing this fact, we propose a two-stage assigning approach as follows. In the first stage, we assign crossdocks and routes for suppliers; in the second stage, by taking advantage of the former assignments for suppliers, we then assign crossdocks and routes for customers. This principle is used throughout the process of HGA including the initial solution generation and the solution updates. We describe the overall procedure of HGA briefly as follows, the formal procedure will be presented in Section 3.2 after all the components of HGA are discussed in Section 3.1.

*Step 1.* Generate initial solutions (chromosomes) by greedy technique. As described above, there are two stages to generate initial solutions. In the first stage, we apply *cost saving priority* to the assignments of crossdocks and routes to suppliers. In the second stage, for customers, the procedure is also based on *cost saving priority*, and then adjust the solution by *time match criterion*.

*Step 2.* Evaluate the fitness of each individual with respect to the objective function.

*Step 3.* Select a group of best individuals as the population pool, which guarantees that the best genes can be preserved in offsprings.

*Step 4.* Apply two-opt strategy as the crossover operator to generate offsprings.

*Step 5.* Apply mutation to diversify the pool by changing some genes in specified chromosomes.

*Step 6.* Apply variable neighborhood search to each new generated offspring, and go to Step 2 until one of the termination conditions is satisfied.

Note that both crossover and mutation operators are only applied when assigning for suppliers and any change in assignments for suppliers will trigger changes in assignments for customers. Furthermore, in HGA, variable neighborhood search is applied to improve the solution. This evaluation-selection-reproduction-local search cycle is repeated until one

$v_1$	$\chi_1$	$\chi_2$	$\chi_3$	$\chi_4$	$\chi_5$	$\chi_6$	$\chi_7$	$\chi_8$
$v_2$	$\psi_1$	$\psi_2$	$\psi_3$	$\psi_4$	$\psi_5$	$\psi_6$	$\psi_7$	$\psi_8$

Figure 1: Two-vector chromosome for  $n = m = 4$ .

of the termination conditions is satisfied, namely, either the maximum number of iterations is reached or the best solution cannot be improved within a certain number of iterations.

### 3.1. Components of HGA

#### 3.1.1. Solution Representation

The chromosome is an important component in GA, which has a great influence on the algorithm output. In the basic GA, a chromosome is usually encoded as *one* sequence to represent a solution. However, since our problem involves assigning *both the crossdocks and routes*, we construct the chromosome by two vectors. The first vector represents the assignments of crossdocks to suppliers and customers, and the second vector represents the assignments of routes to suppliers and customers. Formally, the two vectors are as follows.

- (1) crossdocks assignment vector (hereafter called  $v_1$ ),
- (2) routes assignment vector (hereafter called  $v_2$ ).

For 1, crossdocks assignment vector is represented as  $v_1 = (\chi_1, \chi_2, \dots, \chi_{n+m})$ , where  $n$  and  $m$  are the number of suppliers and that of customers, respectively; each  $\chi_i \in \{1, 2, \dots, l\}$  (recall that there are  $l$  crossdocks) represents an assignment of crossdock  $\chi_i$  to supplier  $i$  ( $i = 1, \dots, n$ ) or customer  $i - n$  ( $i = n + 1, \dots, n + m$ ). That is, supplier  $i$  ships cargos to crossdock  $\chi_i$  ( $i = 1, \dots, n$ ), or customer  $i - n$  receives cargos from crossdock  $\chi_i$  ( $i = n + 1, \dots, n + m$ ). For 2, routes representation vector  $v_2$  is designed in the way similar to  $v_1$ ,  $v_2 = (\psi_1, \psi_2, \dots, \psi_{n+m})$ , where  $\psi_i$  means supplier  $i$  ( $i = 1, 2, \dots, n$ ) or customer  $i - n$  ( $i = n + 1, n + 2, \dots, n + m$ ) chooses route  $\psi_i$  to release or receive cargos. Note that  $\psi_i$  represents an available fixed schedule between any two points. In particular, when  $1 \leq i \leq n$ ,  $\psi_i$  means supplier  $i$  chooses  $\psi_i^{\text{th}}$  route among all the available routes  $\{1, 2, \dots, \gamma_{i,\chi_i}\}$  to ship cargos to crossdock  $\chi_i$  (which has been assigned in vector  $v_1$  already); similarly, when  $n + 1 \leq i \leq n + m$ ,  $\psi_i$  means customer  $i - n$  chooses  $\psi_i^{\text{th}}$  route among all the available routes  $\{1, 2, \dots, \gamma'_{\chi_i,i}\}$  to ship cargos to crossdock  $\chi_i$  (which again has been assigned in vector  $v_1$ ). The whole chromosome for a problem instance with four suppliers and four customers is illustrated in Figure 1.

The initial sequences are generated randomly. However, given such a chromosome sequence, we cannot guarantee the feasibility of the solution because the time windows of suppliers and customers may conflict with each other when proper transportation schedules do not exist or the capacity constraints of crossdocks or routes are violated in the cargo transferring process. In order to overcome these difficulties and find a feasible solution efficiently, a greedy technique is applied to identify a relatively better solution. More details about generation of initial solution will be given next.

### 3.1.2. Generation of Initial Solution

Common integer programming methods usually fail for large-scale problems. In view of the complexity of the crossdock problems, in order to obtain a much better solution, our approach is to attain initial solutions using greedy method. At first, we can combine the transportation cost and the penalty cost together in the objective function by some algebra as follows

$$\sum_{i=1}^n \sum_{j=1}^l \sum_{q=1}^{\gamma_{i,j}} (c_{i,j,q} - P_i) s_i x_{i,j,q} + \sum_{k=1}^m \sum_{j=1}^l \sum_{q=1}^{\gamma'_{j,k}} (c'_{j,k,q} - P'_k) d_k x'_{j,k,q} + C, \quad (3.1)$$

where constant  $C = \sum_{i=1}^n P_i s_i + \sum_{k=1}^m P'_k d_k$  and each coefficient is negative because we assume that the unit penalty cost is higher than unit transportation cost. Let  $C_{i,j,q}$  represent the cost supplier  $i$  can save if he ships cargos to crossdock  $j$  by  $(b_{i,j,q}^r, e_{i,j,q}^r)$  and  $C'_{j,k,q}$  represent the cost customer  $k$  can save if he receives cargos from crossdock  $j$  by  $(b_{j,k,q}^a, e_{j,k,q}^a)$ , where  $C_{i,j,q} = P_i - c_{i,j,q}$  and  $C'_{j,k,q} = P'_k - c'_{j,k,q}$ , respectively.

From (3.1), we can see that, for a supplier, the most important point is saving cost  $C_{i,j,q}$ , which is the primary factor that determines which crossdock to ship to and which route to be chosen between these two locations. This decision subsequently affects the holding cost in the corresponding crossdock it ships to. To reflect this fact, we set probabilities for supplier-crossdock assignments so that a higher cost saving of an assignment would result in a higher probability for that assignment to be chosen. Formally, the probability that supplier  $i$  ships cargos to crossdock  $j$  by schedule  $(b_{i,j,q}^r, e_{i,j,q}^r)$  is calculated as follows:

$$\text{Prob}_{i,j,q} = \frac{C_{i,j,q}}{\max_{j',q'} \{C_{i,j',q'}\}} \quad (1 \leq i \leq n, 1 \leq j \leq l, 1 \leq q \leq \gamma_{i,j}). \quad (3.2)$$

For customers, we know which crossdocks have cargo after the first stage, and then we assign one of these crossdocks to each customer and choose a route to deliver. The assignment strategy of crossdocks and schedules for customer is similar to that of the suppliers, and we also can calculate the probability similar to (3.2). Only one difference is that the customers just can be assigned to those crossdocks that have been already assigned to suppliers, instead of all the crossdocks. After that, we need adjust the solution according to time match criterion to guarantee feasibility. During adjustment, we need to eliminate those infeasible issues such as time conflicts, overflow of capacity, and nonconservation of cargo flows. By adjustment, infeasible solutions will scarcely be generated. However, for some infeasible solutions that are too difficult to repair, we just need to unfulfill those customers who incur infeasibility to get a feasible solution.

### 3.1.3. Crossover

In order to preserve efficient genes in a chromosome, the two-point crossover operator is applied to generate offspring, which is widely adopted in GA (see, e.g., [3, 18]). The crossover operator is illustrated by Figure 2. First, two individuals, we call *parent 1* and *parent 2*, are selected randomly from the population pool, and then two points are randomly selected between genes representing assignment for suppliers, and because crossover operators

Supplier segment of parent 1	$v_1$	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$	$X_7$	$X_8$
	$v_2$	$\psi_1$	$\psi_2$	$\psi_3$	$\psi_4$	$\psi_5$	$\psi_6$	$\psi_7$	$\psi_8$
					↓	↓	↓		
Supplier segment of offspring 1	$v_1^*$	$X'_1$	$X'_2$	$X'_3$	$X_4$	$X_5$	$X_6$	$X'_7$	$X'_8$
	$v_2^*$	$\psi'_1$	$\psi'_2$	$\psi'_3$	$\psi_4$	$\psi_5$	$\psi_6$	$\psi'_7$	$\psi'_8$
		↑	↑	↑				↑	↑
Supplier segment of parent 2	$v'_1$	$X'_1$	$X'_2$	$X'_3$	$X'_4$	$X'_5$	$X'_6$	$X'_7$	$X'_8$
	$v'_2$	$\psi'_1$	$\psi'_2$	$\psi'_3$	$\psi'_4$	$\psi'_5$	$\psi'_6$	$\psi'_7$	$\psi'_8$

Figure 2: Crossover operator.

applied among customers will generate lots of infeasible solutions, assignments for customers are determined by assignments for suppliers. The symbols outside the two crossover points are directly inherited from *parent 1* to *offspring 1*, and the other genes of *offspring 1* are transferred from the symbols of *parent 2* in corresponding positions. After crossover, crossdocks and schedules are reassigned for customers using the aforementioned greedy technique. After changing the roles of parents, the same procedure is applied to generate *offspring 2*.

### 3.1.4. Mutation

It is obvious that the initial population generated by two-stage greedy method has poor ability to carry the genetic diversity because the cost saving priority and time match priority in the greedy technique reduce the chance for crossdocks and schedules with relatively low cost saving to be chosen. As a result, the greedy technique causes population pool homogeneity. In order to overcome this limitation, we use the mutation operator with a given individual mutation probability  $P_{im}$  to mutate every individual and apply the greedy technique to mutation of some gene representing the assignment of crossdock to some supplier with gene mutation probability  $P_{gm}$  calculated by (3.2) in the selected individual. After that, we also need to adjust the new solution to be feasible by the strategy which is similar to that of initial solutions. Different from prior crossover operator slightly, the mutation operator may deteriorate the current solution in terms of fitness. However, its goal is not only to preserve the best genes but also to attain inferior genes with some probability to diversify the pool.

### 3.1.5. Variable Neighborhood Search

GA is a global search technique but is poor in local search. Therefore, we use variable neighborhood search technique to improve local search ability. A basic component of any local search is neighborhood search. A solution is said to be a neighbor of another solution  $s$  if it can be obtained from  $s$  through a neighborhood move. We develop several such moves suitable for this problem to find neighborhood solutions. These moves are key to the successful implementation of these heuristics. Next are the key moves and strategies for neighborhood search.

### *Vary Crossdock for Supplier*

In our algorithm, the crossdock assignment and route choosing problem has a special feature that a sequence may not cover all the crossdocks, which is different from other order-based problems. So single-point change strategy is applied here rather than two-opt strategy. First, we randomly select a gene in supplier segment in one chromosome and change its assigned crossdock to another that is different from the current one with probability calculated by (3.2) and repeat this procedure until a better solution is obtained. It should be emphasized that, in each search process, we use the greedy technique mentioned in Section 3.1.2 to assign crossdocks and schedules to customers after the reassignment for each supplier. This is an efficient strategy to guarantee the feasibility of the new solution.

### *Change Route for Supplier*

Routes selection is the most difficult decision in our problem, especially for suppliers. There is no good method available that can efficiently select routes, which can guarantee the feasibility besides the saving cost, not only in generating initial solution but also in generation of new population. However, routes selection has a great influence on total cost. In addition, it affects the inventory level of each possible time point in crossdocks, which determines the customers crossdock assignment and route selection in our algorithm. In order to obtain a better route, our strategy is to not only change the routes for suppliers, but also keep the crossdock assignment unchanged. In this move, we change the assigned route of a randomly selected supplier to a new one, and repeat this procedure until a better solution is obtained. It must be of concern that, for any change in suppliers route, we must conduct reassignment of crossdocks and schedules for customers to ensure the feasibility.

### *Swap Crossdocks for Customers*

In the assignment process, we assign crossdocks and routes for customers one by one, which would reduce the probability for latter customers to choose a particular crossdock. For example, suppose that the inventory at certain time of a *crossdock j* could meet the demand of *customer 1* or *customer 2* individually if there exist available routes for both of them and the probabilities for them to choose the crossdock are close to each other. If *customer 1* chooses *crossdock j* first, then *customer 2* would have small possibility to also choose *crossdock j* because the remaining inventory of *crossdock j* may not meet its demand. This move is designed to overcome this difficulty. That is, we give priority to *customer 2* if that helps reduce the total cost. Our strategy for this move is to select two customers randomly whose crossdocks are different from each other, swap their crossdocks, and then repeat this procedure until a better solution is obtained.

### *Change Route for Customer*

The goal for this move is the same as the third strategy, that is, eliminating the ordering effect for assigning crossdocks and routes for customers by the greedy method. The difference is that this move focuses on two customers whose crossdocks are the same. It is obvious that, if two customers choose the same crossdock but are assigned to receive cargos in order, for example, supposed *customer 1* and *customer 2* select the same crossdock, *customer 1* has an advantage to ship out cargos in time; however, the total cost may be much lower if *customer*

```

Generate initial solutions by two-stage greedy method.
for iter  $\leftarrow$  1 to #maximum_iter do
    for off  $\leftarrow$  1 to #crossover do
        Randomly select Parent 1 and Parent 2.
        Crossover Parent 1 and Parent 2 to produce a new Offspring.
    end for
    for each offspring do
        Mutate offspring with individual mutation probability  $P_{im}$  and gene mutation probability  $P_{gm}$ 
        Apply neighborhood search to each newly-produced Offspring.
    end for
    Select the best #pop individuals from all the Individuals including all current parents and newly produced offsprings.
    Update current best solution.
    if the best solution could not improve within #terminate_iter then
        Consider the solution as the goal optimal solution.
        break
    end if
end for
output the best solution and escaped time

```

**Algorithm 1:** HGA to solve problem (P).

2 can ship out cargos earlier than *customer 1* when the penalty cost of *customer 2* is relatively high. So the strategy aims to give a priority to *customer 2*.

### 3.2. Framework of HGA

With these components, we now outline HGA framework in Algorithm 1. In this algorithm, #*pop* denotes the number of populations, #*crossover* denotes the number of crossovers we will do, #*terminate\_iter* denotes the maximum number of iterations the current best solution cannot be improved, #*maximum\_iter* denotes the maximum number of iterations, and  $P_{im}$  and  $P_{gm}$  are defined in Section 3.1.4.

## 4. Computational Experiments

We generate a great variety of problem instances and apply HGA to solve them. For comparison purposes, we also use ILOG CPLEX 11.0 solver to solve the instances, which is widely adopted by many papers (see, e.g., [3, 21, 22]). Both HGA and the CPLEX solver were run on a personal computer with an Intel 2.4GHz Pentium 4 CPU and 1G memory. The test data generation, parameter settings of HGA, and detailed computational results are reported in the following content.

### 4.1. Test Data Generation and Experimental Parameter Setting

Because crossdocking problems are relatively new, there are no benchmarks test sets available. As a result, we generated our own data. The data sets are generated randomly

in such a way that they can represent realistic situations and can cover different scenarios, which is suggested by Chen et al. [21], Li et al. [3], and Ma et al. [22], and the parameters in HGA are based on numerous computational experiments, and they are effective to attain desirable results.

The test data generation procedure requires three basic parameters: the number of suppliers  $n$ , the number of customers  $m$ , and the number of crossdocks  $l$ . The time horizon is fixed at 48 hours (2 days) in the test sets; note that this is usually the longest-time shipments by railways between two cities. The  $n$  start points of supplier  $i$  time window  $b_i^r$  ( $1 \leq i \leq n$ ) were then randomly generated from a uniform distribution  $U[0, 12]$ . The end points of supplier  $i$  time window  $e_i^r$  were also randomly generated from a uniform distribution  $U[12, 36]$ . For customers, their time windows are generated as  $[b_k^a, e_k^a]$ , where  $b_k^a \sim U[12, 24]$  and  $e_k^a \sim U[24, 48]$ . The number of fixed transportation schedules between two points is randomly generated in the interval  $[6, 8]$ . Meanwhile, the beginning time of the first fixed transportation schedule from supplier to crossdock is generated according to penalty cost and transportation cost, so the fixed schedule is as  $[begin, end]$ , where  $begin \sim U[(b_i^r \times P_i) / (P_i - c_{i,j,1}), 12]$  and  $end \sim U[13, 24]$ , which means that a higher penalty cost provides a supplier with a higher motivation to ship out cargos. Other schedules are generated as  $[begin, end]$ , where  $begin \sim U[\text{the first schedule begin time}, 12]$  and  $end \sim U[13, 24]$ . Similarly, the arrival time of the last delivery schedule for customers is generated according to penalty cost and delivery cost, so the time window of the last delivery schedule is as  $(begin, end)$ , where  $begin \sim U[12, 35]$  and  $end \sim U[36, (e_k^a \times (c'_{j,k,1} + P'_k)) / P'_k]$ . For others, the time window is as  $[begin, end]$ , where  $begin \sim U[12, 35]$  and  $end: U[36, \text{the last schedule arrival time}]$ . Next, because pickups usually follow deliveries within short times, we take the inventory cost at crossdocks to be small relative to transportation costs. This reflects the fact that handling costs are usually smaller than transportation costs. Based on this, the transportation cost per unit cargo of each fixed scheduled route is uniformly generated in the interval  $[10, 30]$  and inventory handling cost per unit per hour is uniformly generated in the interval  $[1, 3]$ , which on average is  $1/10$  of transportation cost. The penalty cost is set to be relatively higher compared to the transportation cost, which can enforce suppliers and customers to deliver cargoes on time, so the penalty cost per unit cargo is uniformly generated in the interval  $[30, 90]$ . Lastly, the amount of supplied cargo  $s_i$  (demanded cargo  $d_k$ ) is uniformly generated in the interval  $[100, 500]$ . The capacity of each crossdock is set to  $\alpha \sum_{1 \leq i \leq n} s_i$ , where  $\alpha$  is randomly generated from a uniform distribution  $U[0.5, 0.8]$ . Also the capacity of each route is set to  $\beta \sum_{1 \leq i \leq n} (s_i / n)$ , where  $\beta$  is randomly generated from a uniform distribution  $U[2, 5]$ . The following values of parameters are used: #max\_iter =  $10^4$ , #terminate\_iter = 100, #pop = 200, and #crossover = 80. The mutation probability  $P_{im}$  is taken to be 0.02, which is proved to be effective in experiments.

## 4.2. The Results and Analysis

Based on the number of suppliers, crossdocks, and customers, we designed three categories of problem instances to test HGA: small, medium, and large scale. The results are presented in Tables 1, 2 and 3. Each category has 40 test instances, sorted into 8 groups where each group has 5 instances. The first row of each table specifies the instance size.  $n \times l \times m$  denotes that there are  $n$  suppliers,  $l$  crossdocks, and  $m$  customers for this instance group. The rest of each table provides the computational result of CPLEX and HGA. For each instance, the following key values were reported: the average objective value, the average computational

**Table 1:** Result of CPLEX and HGA on random instances with small scale.

Problem size		10×4×10	12×4×12	14×4×14	16×4×16	18×4×18	10×6×10	12×6×12	14×6×14
CPLEX	LBs	104818	105058	139556	167451	144959	114548	104734	145931
	Objective	111094	107805	142217	174498	152658	123788	108595	155670
	Time(s)	>3600	>3600	3301.1	>3600	>3600	>3600	>3600	>3600
	Gap	5.52%	2.21%	1.85%	4.14%	4.02%	6.24%	3.36%	5.71%
HGA	Objective	110834	107808	142124	173815	150214	122921	108113	154486
	Time(s)	472.93	781.9	530.79	922.69	562.30	726.26	608.8	1084.69
	Gap	5.30%	2.21%	1.80%	3.79%	2.81%	5.59%	2.97%	5.14%

**Table 2:** Result of CPLEX and HGA on random instances with medium scale.

Problem size		20×4×20	22×4×22	16×6×16	18×6×18	20×6×20	16×8×16	18×8×18	16 × 10 × 16
CPLEX	LBs	153165	197595	143037	161171	185072	131554	149797	132087
	Objective	158295	202441	154712	169924	195823	142704	161564	150179
	Time(s)	>5000	>5000	>5000	>5000	>5000	>5000	>5000	>5000
	Gap	2.64%	2.37%	7.48%	4.95%	5.17%	7.74%	6.90%	12.06%
HGA	Objective	157928	202398	153922	167337	194017	138126	159152	147352
	Time(s)	1080.17	1741.77	1139.11	677.49	1326.22	675.84	1224.98	609.23
	Gap	2.48%	2.34%	6.97%	3.69%	4.40%	4.40%	5.71%	10.22%

time, the gaps between value attained by both CPLEX and HGA, and the low bound attained by CPLEX when terminated.

- (1) Small-size instances: the results are shown in Table 1. In this category, eight small scale instance groups are generated with the size  $n$  and  $m$  ranging from 10 to 14, and  $l$  ranging from 4 to 6. We use these instances to compare the performance of the CPLEX solver and HGA. We find that, only in one group, CPLEX solver reaches the LB within time limit set as 3600 s; for other cases, CPLEX fails to get the better solutions within 3600 s comparing HGA, which gets better solutions more quickly, and of which the average gaps are apparently smaller than CPLEX solver.
- (2) Medium-size instances: the results are reported in Table 2. In this category, eight instance groups with the size  $n$  and  $m$  ranging from 16 to 22 and  $l$  ranging from 4 to 10 are tested. Time limit is set to more than 5000 s. Also CPLEX fails to get the better solutions within the time limit for all the instance groups, while HGA performs well in no more than 1200 s.
- (3) Large-size instances: the results can be found in Table 3. In this category, large-scale instance groups are generated and categorized into 8 groups with the size  $n$  and  $m$  ranging from 20 to 24 and  $l$  ranging from 8 to 12. The CPLEX solver is unable to obtain the better solutions within the time limit, which is set to 7200 s; only in one group CPLEX gets a better solution than HGA. However, HGA can attain much better solutions in no more than 1800 s in the other seven cases.

All the three categories of 24 instance groups show that HGA performs fairly well and is preferable over the commercial CPLEX solver.

The main feature of our proposed HGA is to integrate variable neighborhood search (VNS) into a general GA framework so that it has the ability to get better solutions, especially

**Table 3:** Result of CPLEX and HGA on random instances with large scale.

Problem size		20×8×20	22×8×22	24×8×24	18 × 10 × 18	20 × 10 × 20	22 × 10 × 22	22 × 12 × 22	24 × 12 × 24
CPLEX	LBs	179309	189188	196688	1495661	162114	170686	208110	192499
	Objective	195510	207942	217234	160386	186614	194127	232170	227976
	Time(s)	>7200	>7200	>7200	>7200	>7200	>7200	>7200	>7200
	Gap	7.68%	8.64%	9.20%	6.38%	13.12%	11.80%	10.36%	14.95%
HGA	Objective	194431	204953	216949	160445	183441	189251	227295	221557
	Time(s)	1159.11	1718.59	1696.08	1314.28	1298.37	1316.45	1419.08	1545.45
	Gap	7.30%	7.34%	9.07%	6.43%	11.71%	9.77%	8.44%	12.96%

**Table 4:** Gap between HGA and GA.

Problem size		30×10×30	34×10×34	36×10×36	40×10×40	36×15×36	40×15×40
HGA	Objective	261081	299320	296109	347677	306791	334069
	Time(s)	1365.87	1626.23	1801.35	1661.80	1738.46	1823.73
GA	Objective	280881	309925	314313	355154	334459	357053
	Time(s)	1338.44	1303.33	1405.94	1691.24	1332.00	1386.86
Gap		7.58%	3.54%	6.15%	2.15%	9.02%	6.88%

for large-scale problem instances. Hence, by comparing the results of HGA and GA without VNS, we can identify how much the solutions can be improved for large-size problem instances. The numerical results are reported in Table 4, where each problem size has 5 instances, and Gap is defined by

$$\frac{\text{Objective\_GA} - \text{Objective\_HGA}}{\text{Objective\_HGA}} * 100\%. \quad (4.1)$$

The results show that our proposed HGA provides better solutions without sacrificing much computational efforts compared with the GA. Specifically, the results show that HGA outperforms GA for all the cases in terms of solution quality. Although the speed of HGA is slower than GA, it is reasonable because HGA requires more time to search for a better solution by applying VNS. This gives us a clearer idea of the performance of the proposed HGA for the large-sized problems. That is, in general, HGA can provide high-quality solutions in realistic timescales for large-size problems.

Note that our problem has many characteristics (e.g., fixed transportation schedules, inventory capacity, etc.) different from the vehicle routing problem (VRP), although the VRP also considers how to find an optimal transportation scheme to satisfy customer demands. The heuristic algorithms that can be very effective for the VRP cannot be applied to our problem because these two types of problems have different structures and constraints.

## 5. Conclusions

In this paper, we consider multiple crossdocks problem through fixed transportation schedules with time windows, capacity, and penalty. The objective is to minimize the total costs including shipment costs, penalty cost, and inventory cost. Since we prove that the

problem is  $\mathcal{NP}$ -hard in the strong sense, we focus on developing an efficient heuristic algorithm. Based on the problem structure, we propose an HGA to solve the problem efficiently. In HGA, we employ two vectors (two sequences) to represent a solution including crossdock assignment and route assignment, and we use a greedy method to generate initial solutions that can help the solutions achieve feasibility. We apply variable neighborhood search to eliminate the limitations caused by greedy method and accelerate convergence rate of HGA. Experiments are conducted by using a wide range of test data sets that reflect various realistic scenarios with different problem sizes. Computational results demonstrate that HGA is preferable over the commercial CPLEX Solver.

Our main contribution is threefold. First, the problem we consider represents a class of transshipment problems that arise from real-world applications, which may help industrial practitioners to improve the transshipment decisions within multiple crossdock networks. Second, we set up an integer programming model for this special problem and show that its complexity is strongly  $\mathcal{NP}$ -hard, which implies that it is unlikely to find a polynomial or pseudopolynomial time algorithm to solve the problem unless  $\mathcal{P} = \mathcal{NP}$ . Third, we propose a hybrid genetic algorithm integrating greedy technique and variable neighborhood search method that exploits the problem structure and is able to solve the problem effectively and efficiently. The proposed heuristic sheds light on solving many other related complex multiple crossdocks problems.

There are a few directions for further research. Firstly, it may be worthwhile to consider different cost structures, for example, discounted transportation costs based on the shipping amount. Secondly, lateral transportation between crossdocks may be considered. Finally, the current problem can be extended to the multicommodity consolidation problem with repacking consideration, in which various types of goods or freight are considered in a given supply chain transshipment network, as well as the packing problem.

## Appendix

### Proof of Proposition 2.1

We provide a reduction of the strongly  $\mathcal{NP}$ -complete *3-partition* problem: given positive integers,  $w$ ,  $D$ , and  $\Gamma = \{1, 2, \dots, 3w\}$  with positive integer values  $\gamma(i)$  where, for each  $i \in \Gamma$ ,  $\sum_{i \in \Gamma} \gamma(i) = wD$  and  $D/4 < \gamma(i) < D/2$  for  $i \in \Gamma$ , can  $\Gamma$  be partitioned into  $w$  disjoint sets  $\Gamma_1, \Gamma_2, \dots, \Gamma_w$  such that  $|\Gamma_k| = 3$  and  $\sum_{i \in \Gamma_k} \gamma(i) = D$  for  $k = 1, \dots, w$ ? From an arbitrary instance of *3-partition*, we consider a polynomial reduction to an instance of our multiple crossdocks problem and ask if there exists a feasible solution whose objective value is no greater than  $2wD$ . For  $w$  suppliers given in  $\Sigma$  (let  $\Sigma = \{1, 2, \dots, w\}$ ) and  $3w$  customers in  $\Delta$  (let  $\Delta = \Gamma$ ), let  $s_i$  be the supply and  $s_i = D$  for  $i \in \Sigma$  with unit penalty cost 3, while for each  $k \in \Delta$ , let  $d_k = \gamma(k)$  be the demand also with unit penalty cost 3. Exactly one crossdock,  $\chi$ , say, with inventory holding cost 1 per unit product per time, exists linking suppliers with customers. For each supplier  $i$  ( $i \in \Sigma$ ), there is only one fixed transportation schedule  $(i, i + 1)$  with unit transportation cost 1. On the other hand, for each customer  $k$  ( $k \in \Delta$ ), there is  $w$  fixed transportation schedule  $\{(q + 1, q + 2) : q = 1, \dots, w\}$  connected with crossdock  $\chi$  also with unit shipping cost 1.

We now show that a feasible schedule exists whose objective value is no greater than  $2wD$  if and only if the *3-partition* has a feasible solution. On the one hand, if *3-partition* has a feasible solution  $\Gamma_1, \dots, \Gamma_w$ , note that we need pay attention to the single shipping and single delivery condition, and hence we should ship all goods provided by supplier  $i$  ( $i \in \Sigma$ ) to  $\chi$

through fixed schedule  $(i, i + 1)$ , respectively, and transship all of them to customer  $j$  ( $j \in \Gamma_i$ ) through  $(i + 1, i + 2)$ , which satisfies the demand  $\gamma(j)$  for customer  $j$  ( $j \in \Gamma_i$ ) exactly. It is easy to verify that such a schedule is feasible and total cost is  $2wD$ . On the other hand, if a feasible schedule exists with objective no greater than  $2wD$ , then it is optimal since it is easy to prove that  $2wD$  is the lower bound of our instance, whose reason is because the total transportation cost is  $2wD$  at least, and if any cargo is delayed in crossdock or any demand is unfulfilled, then the total cost is definitely greater than  $2wD$ . Hence, this optimal solution must satisfy the following two conditions: (1) there is no inventory in crossdock at any time; (2) no penalty cost is incurred. We can then construct a partition by setting  $\Delta_i$  to be the subset of  $k \in \Delta$  whose demand is satisfied by supplier  $i$  for  $1 \leq i \leq w$ . Because of conditions (1) and (2), the demand of customer  $k$  ( $k \in \Delta$ ) is  $\gamma(k)$  which should be satisfied immediately by fixed schedule  $(i + 1, i + 2)$ . Moreover, because of the single shipping and single delivery condition, we have  $\sum_{k \in \Delta_i} d_k = D$ . Since  $D/4 < d_k < D/2$  for  $k \in \Delta$ , we have  $|\Delta_i| = 3$ . Hence,  $\Delta_1, \dots, \Delta_w$  is a feasible partition for the instance of 3-partition and this completes the proof.

## Acknowledgments

The authors wish to acknowledge one anonymous referee whose comments have helped to greatly improve this paper. This research was supported in part by NSFC (70802052, 70701039, 71072090), MOE (NCET-10-0712, NCET-10-0847), the Fundamental Research Funds for the Central Universities (2010221025, 10wkpy20), the Academic Outstanding Youthful Research Talent Plan of Fujian Province (JA10001S), and Soft Science Projects of Fujian Province (2011R0081).

## References

- [1] D. Simchi-Levi, P. Kaminsky, and E. Simchi-Levi, *Designing and Managing the Supply Chain*, McGraw-Hill—Irwin, Boston, Mass, USA, 2nd edition, 2003.
- [2] T. Brockmann, “21 Warehousing trends in the 21st century,” *IIE Solutions*, vol. 31, pp. 36–40, 1999.
- [3] Y. Li, A. Lim, and B. Rodrigues, “Crossdocking—JIT scheduling with time windows,” *Journal of the Operational Research Society*, vol. 55, no. 12, pp. 1342–1351, 2004.
- [4] M. Gumerus and J. H. Bookbinder, “Cross-docking and its implication in location-distribution systems,” *Journal of Business Logistics*, vol. 25, no. 2, pp. 199–228, 2004.
- [5] K. Krishnan and V. Rao, “Inventory control in N warehouses,” *Journal of Industrial Engineering*, vol. 16, pp. 212–215, 1965.
- [6] U. S. Karmarkar and N. R. Patel, “The one-period n-location distribution problem,” *Naval Research Logistics*, vol. 24, no. 4, pp. 559–575, 1977.
- [7] U. S. Karmarkar, “The multilocation multiperiod inventory problem: bounds and approximations,” *Management Science*, vol. 33, no. 1, pp. 86–94, 1987.
- [8] G. Tagaras, “Effects of pooling on the optimization and service levels of two-location inventory systems,” *IIE Transactions*, vol. 21, no. 3, pp. 250–257, 1989.
- [9] L. W. Robinson, “Optimal and approximate policies in multiperiod, multilocation inventory models with transshipments,” *Operations Research*, vol. 38, no. 2, pp. 278–295, 1990.
- [10] N. Rudi, S. Kapur, and D. F. Pyke, “A two-location inventory model with transshipment and local decision making,” *Management Science*, vol. 47, no. 12, pp. 1668–1680, 2001.
- [11] J. Grahovac and A. Chakravarty, “Sharing and lateral transshipment of inventory in a supply chain with expensive low-demand items,” *Management Science*, vol. 47, no. 4, pp. 579–594, 2001.
- [12] Y. T. Herer and M. Tzur, “The dynamic transshipment problem,” *Naval Research Logistics*, vol. 48, no. 5, pp. 386–408, 2001.

- [13] Y. T. Herer, M. Tzur, and E. Yücesan, "Transshipments: an emerging inventory recourse to achieve supply chain leagility," *International Journal of Production Economics*, vol. 80, no. 3, pp. 201–212, 2002.
- [14] S. Axsäter, "A new decision rule for lateral transshipments in inventory systems," *Management Science*, vol. 49, no. 9, pp. 1168–1179, 2003.
- [15] S. Axsäter, "Evaluation of unidirectional lateral transshipments and substitutions in inventory systems," *European Journal of Operational Research*, vol. 149, no. 2, pp. 438–447, 2003.
- [16] H. Donaldson, E. L. Johnson, H. D. Ratliff, and M. Zhang, "Schedule-driven crossdocking networks," Research Report, Georgia Institute of Technology, 1999.
- [17] D. H. Ratcliff, J. van de Vate, and M. Zhang, "Network design for load-driven cross-docking systems," Research Report, Georgia Institute of Technology, 1999.
- [18] Z. Miao, A. Lim, and H. Ma, "Truck dock assignment problem with operational time constraint within crossdocks," *European Journal of Operational Research*, vol. 192, no. 1, pp. 105–115, 2009.
- [19] N. Boysen, M. Fliedner, and A. Scholl, "Scheduling inbound and outbound trucks at cross docking terminals," *OR Spectrum*, vol. 32, no. 1, pp. 135–161, 2009.
- [20] A. Lim, Z. Miao, B. Rodrigues, and Z. Xu, "Transshipment through crossdocks with inventory and time windows," *Naval Research Logistics*, vol. 52, no. 8, pp. 724–733, 2005.
- [21] P. Chen, Y. Guo, A. Lim, and B. Rodrigues, "Multiple crossdocks with inventory and time windows," *Computers and Operations Research*, vol. 33, no. 1, pp. 43–63, 2006.
- [22] H. Ma, Z. Miao, A. Lim, and B. Rodrigues, "Crossdocking distribution networks with setup cost and time window constraint," *Omega-International Journal of Management Science*, vol. 39, no. 1, pp. 64–72, 2011.
- [23] L. Hai and A. Lim, "A metaheuristic for the pickup and delivery problem with time windows," *International Journal on Artificial Intelligent Tools*, vol. 12, no. 2, pp. 173–186, 2003.
- [24] M. Gendreau, F. Guertin, J. Y. Potvin, and R. Séguin, "Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries," *Transportation Research Part C*, vol. 14, no. 3, pp. 157–174, 2006.

*Research Article*

# A Nonlinear Multiobjective Bilevel Model for Minimum Cost Network Flow Problem in a Large-Scale Construction Project

Jiuping Xu,<sup>1,2</sup> Yan Tu,<sup>2</sup> and Ziqiang Zeng<sup>2</sup>

<sup>1</sup> State Key Laboratory of Hydraulics and Mountain River Engineering, Sichuan University, Chengdu 610064, China

<sup>2</sup> Uncertainty Decision-Making Laboratory, Sichuan University, Chengdu 610064, China

Correspondence should be addressed to Jiuping Xu, xujiuping@scu.edu.cn

Received 3 January 2012; Revised 9 March 2012; Accepted 19 March 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Jiuping 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 aim of this study is to deal with a minimum cost network flow problem (MCNFP) in a large-scale construction project using a nonlinear multiobjective bilevel model with birandom variables. The main target of the upper level is to minimize both direct and transportation time costs. The target of the lower level is to minimize transportation costs. After an analysis of the birandom variables, an expectation multiobjective bilevel programming model with chance constraints is formulated to incorporate decision makers' preferences. To solve the identified special conditions, an equivalent crisp model is proposed with an additional multiobjective bilevel particle swarm optimization (MOBLPSO) developed to solve the model. The Shuibuya Hydropower Project is used as a real-world example to verify the proposed approach. Results and analysis are presented to highlight the performances of the MOBLPSO, which is very effective and efficient compared to a genetic algorithm and a simulated annealing algorithm.

## 1. Introduction

Network flow optimization is a large part of combinatorial optimization. The minimum cost network flow problem (MCNFP) is made up of a wide category of problems [1, 2]. MCNFP plays a very important role in many real-world applications such as communications [3, 4], informatics [5], and transportation [6]. Other well-known problems like the shortest path problem and the assignment problem are considered to be special MCNFP cases [7].

In recent decades, the MCNFP has been well researched with many models and algorithms being developed, for example, [8–13]. These studies, however, have not often taken carrier type selection and transportation time into account when looking at the transportation network. Yet, both cost and time control are important in construction projects, especially in

large-scale construction projects where transportation costs are largely based on the rates charged by carriers, which have a significant influence on transportation time. In real conditions, there is increasing pressure to shorter transportation time to reduce or eliminate extra project expenses, with the early arrival of materials shortening the completion time of the construction project and improving construction efficiency. In these cases, it is necessary to include both the carrier type selection or transportation time in the transportation network analysis. In this paper, a multiobjective bilevel MCNFP is studied. On the upper level, the construction contractor determines the material flow of each transportation network path with the criteria being the minimization of both direct costs and total transportation time costs. On the lower level, the transportation manager controls each carrier's flow so that total transportation costs are minimized.

The research presented previously has a common foundation in that they were all based on a deterministic transportation network. However, transportation systems are often complex, so decision makers inevitably encounter uncertain parameters when making a decision. Within the last two decades the use of multimodels with uncertain parameters in the study of network flow problems has been increasingly exploited. For example, Watling [14] studied a user equilibrium traffic network assignment problem with stochastic travel times and a late arrival penalty. Chen and Zhou [15] developed an  $\alpha$ -reliable mean-excess traffic equilibrium model with stochastic travel times. Lin [16] constructed a revised stochastic flow network to model a realistic computer network in which each arc has a lead time and a stochastic capacity. Sumalee et al. [17] dealt with a reliable network design problem which was looked at uncertain demand and total travel time reliability. In actual analyses, randomness is considered one important source of uncertainty. Yet with MCNFP randomness is seen to be increasingly complex because of the often incomplete or uncertain information. To date, there has been little research which considers multilevel twofold uncertainty coefficients for MCNFP. Therefore this research concentrates on the problem under a birandom environment with the logic behind this choice of birandom variables illustrated in Section 2.

The MCNFP proposed in this paper is a multiobjective bilevel programming problem, first introduced by Geoffrion and Hogan [18], and consequently developed by researchers such as Tarvainen and Haimes [19], Osman et al. [20], Zhang et al. [21], and Calvete and Galéb [22]. Multiobjective bilevel programming has been greatly improved in both the theoretical and practical areas. While these studies have significantly contributed to a variety of applications, to the best of our knowledge, there is still no known research considering the modeling for the MCNFP. With bilevel programming problems being intrinsically difficult, it is not surprising that most exact algorithms to date have focused on the simplest cases of bilevel programs, that is, problems with relatively easy to determine properties such as linear, quadratic, or convex objective and/or constraint functions [23]. Since the proposed bilevel MCNFP model is nonlinear, nonconvex, and nondifferentiable, it follows that the search for exact algorithms which are formally efficient is all but futile and it is necessary instead to search for effective heuristic algorithms to solve the MCNFP. Determining the global optimal solution is of great importance in MCNFP. Specifically, this paper deals with the multiple objectives by employing the concept of nondominated solutions instead of applying weighted sum scalarization. In this study, an effort is made to develop a multiobjective bilevel particle swarm optimization (MOBLPSO) to solve a real world MCNFP in the Shuibuya Hydropower Project.

The remainder of this paper is structured as follows. In Section 2, an introduction to the bilevel MCNFP is presented along with the motivation for employing birandom variables in the problem. An expectation multiobjective bilevel programming model with chance

constraints under a birandom environment is established in Section 3, for which the equivalent crisp model is derived in Section 4. In addition, an MOBLPSO is illustrated in Section 5. In Section 6, an application to earth-rock work transportation in a large construction project is given in order to show the validity and efficiency of the proposed models and algorithms. Concluding remarks and further discussion are in Section 7.

## 2. Problem Statement

In construction projects, and especially in large-scale construction projects, the MCNFP is becoming increasingly important. Here we discuss a multiobjective bilevel MCNFP in a large-scale construction project. In order to establish the model, a description is given.

### 2.1. Bilevel Problem Description

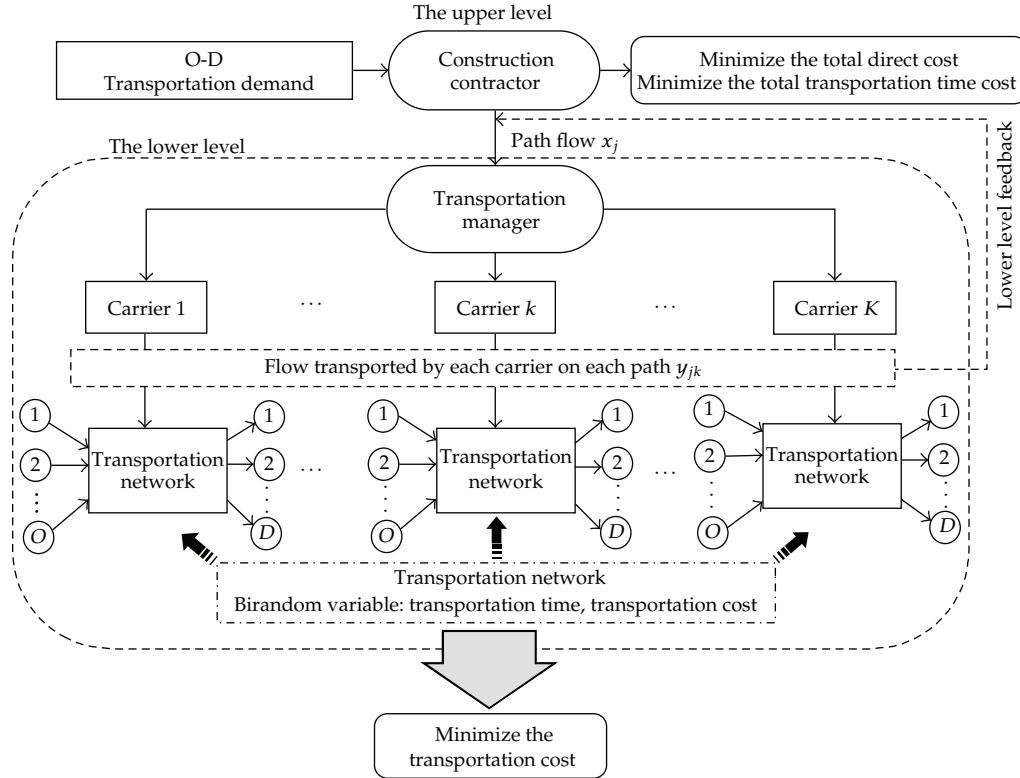
The MCNFP discussed considers both the construction contractor and the transportation manager as the two participants. In a large-scale construction project, material often has supply (origin) and receipt (destination) nodes, with the construction contractor generally assigning a specialized Transportation Company. The bilevel model considers the construction contractor and the transportation manager in the specialized Transportation Company concurrently, gives priority to contractor benefit, and considers the influence of the contractor's decision making on the flow distribution of the transportation manager's carriers. As both cost and time control are important in construction projects, their effectiveness needs to be considered. The construction contractor assigns the flow of material to each transportation path to minimize direct costs and transportation time costs, while the transportation manager aims to minimize transportation costs by making decisions about flow of material by each carrier through the transportation path based on the construction contractor's decision making, which in turn influences contractor's decision-making through adjustments to the flow of material by each carrier along the transportation path.

Therefore, the MCNFP in this paper can be abstracted as a bilevel programming problem. To model the problem conveniently, the involved transportation network is considered a bipartite network represented by a graph with sets of nodes and arcs. In the network, a node represents the facilities in the network, for instance, a station or a yard, and an arc represents a line between two adjacent facilities. The model structure of the MCNFP is in Figure 1.

### 2.2. The Motivation for Considering Birandom Environment in the MCNFP

The birandom environment has been successfully studied and applied in many areas, such as flow shop scheduling problem [24], portfolio selection [25], and vendor selection [26]. These studies show the necessity of considering birandom environment in practical problems. There is a strong motivation for considering birandom environment for the MCNFP.

In real conditions, the transportation plan is usually made before the occurrence of any transportation activity; thus the determined values of some parameters cannot be obtained in advance; so there is a strong need to consider uncertainty in transportation



**Figure 1:** Model structure for MCNFP.

problems. An MCNFP in a large-scale construction project is considered in this paper, which may be subjected to twofold randomness with incomplete or uncertain information. For example, the transportation time in an arc of the project is not fixed because of the effect of the transportation environment which includes many uncertainties such as traffic accidents, traffic congestion, vehicle breakdowns, bad weather, natural disasters, and special events [27]. Therefore, it is often subject to a stochastic distribution. Generally, transportation time approximately follows a normal distribution expressed as  $\mathcal{N}(\tilde{\mu}, \sigma^2)$  [28–30], whereas the normal distribution has to be truncated to avoid negative values. However, the expected value of transportation time  $\tilde{\mu}$  is also uncertain because it is determined by the carrier's speed, which in turn is influenced by such uncertainties as vehicle condition. It is possible to specify a realistic distribution (e.g., normal distribution) for the parameter  $\tilde{\mu}$  through statistical methods or related expertise and other knowledge. When the value of  $\tilde{\mu}$  is provided as a random variable which approximately follows a normal distribution, the pattern of overlapping randomness is said to be birandom as illustrated in Section 3.1. The flowchart of transportation time as a birandom variable is in Figure 2.

The situation is similar with transportation costs. However, the mean values of transportation costs are considered as approximately following normal distributions due to the fluctuation of gasoline prices over time, which results in the birandomness of the transportation costs. From the previous description, the birandom variable is employed to take account of the hybrid uncertainty and obtain a more feasible network flow scheme.

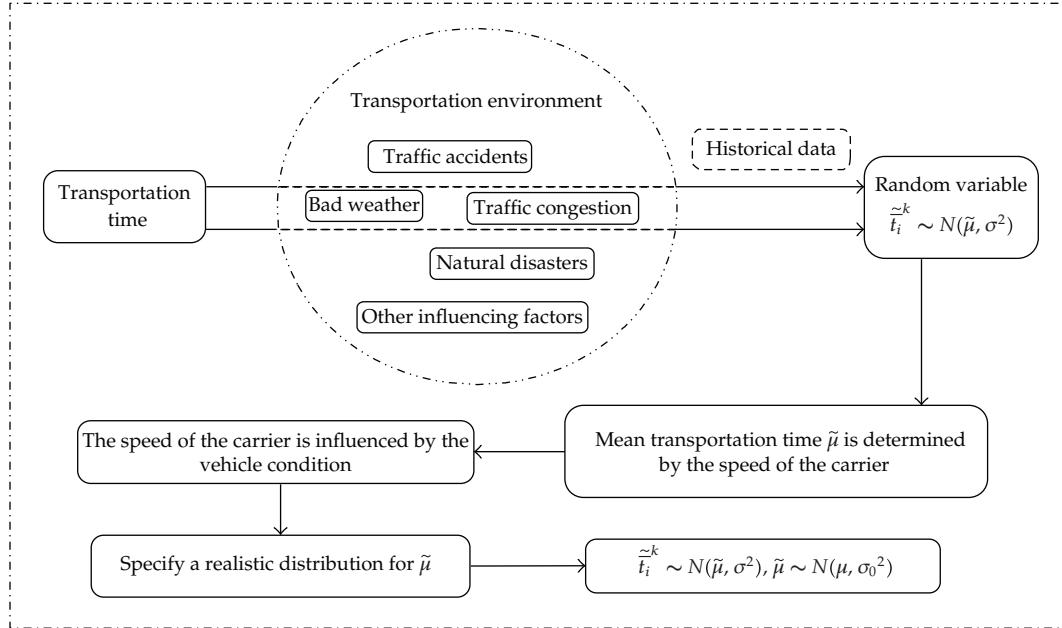


Figure 2: Flowchart of transportation time as a birandom variable.

### 3. Modelling

In this section, the relevant assumptions and notations are first outlined, some basic knowledge about the birandom variable is introduced, and then the nonlinear multiobjective bi-level MCNFP model under a birandom environment is formulated.

#### 3.1. Problem Assumptions

The mathematical model described has the following assumptions.

- (1) The proposed transportation network is a single material transportation network composed of nodes and road sections.
- (2) The capacities of the different arcs are satisfactorily independent, and the total flow of carriers on each arc cannot exceed its capacity.
- (3) Flows on all transportation paths between OD pairs satisfy the feasible flow conservation [31].
- (4) All transportation paths in the network are known.
- (5) The transportation cost of every road section and transportation time are considered birandom variables, with the attributes determined from available statistics and historical data as well as forecast transportation environments. They are considered to be independent.
- (6) The demand at every reception node must be met on schedule. The material has a given transportation duration. If the transportation time exceeds the given duration, then a delay cost is added.

### 3.2. Notations

The following mathematical notations are used to describe the MCNFP.

#### *Subscripts and Sets*

- $o$ : index of origin node;
- $d$ : index of destination node;
- $\Psi$ : set of carriers,  $k \in \Psi$  is an index;
- $\Phi$ : set of arcs in the transportation network,  $i \in \Phi$  is an index;
- $\Omega$ : set of paths in the transportation network,  $j \in \Omega$  is an index;
- $E$ : set of origin-destination (OD) pairs,  $(o, d) \in E$ ;
- $A_j$ : set of arcs in transportation path  $j$ ;
- $P_{od}$ : set of paths from origin node  $o$  to destination node  $d$ .

#### *Certain Parameters*

- $r_i$ : maximal passing capacity of arc  $i$ ;
- $w_k$ : weight for carrier  $k$  in the transportation network;
- $v_k$ : volume capacity of carrier  $k$ ;
- $T_j$ : transportation time constraint represents the time of transportation path  $j$  in  $T_j$  units in which the material demand between all OD pairs has to be transported;
- $\gamma_i^j$ : a binary variable equal to 1 if and only if arc  $i$  is a segment of transportation path  $j$  for carrier  $k$ ;
- $Q_{od}$ : transportation demand of the material from origin node  $o$  to destination node  $d$ ;
- $c_j$ : direct cost of unit volume material using transportation  $j$ ;
- $\lceil \cdot \rceil$ : ceiling operator rounding upward to integer.

#### *Uncertain Parameters*

- $\tilde{e}_i^k$ : unit transportation cost of material flow on arc  $a_i$  for carrier  $k$ ;
- $\tilde{t}_{i0}^k$ : free transportation time of material flow on arc  $i$  for carrier  $k$ ;
- $\tilde{t}_i^k$ : transportation time of material flow on arc  $a_i$  for carrier  $k$ .

#### *Decision Variables*

- $x_j$ : volume of material flow on transportation path  $j$ , which is the decision variable of the upper level;
- $y_{kj}$ : volume of material flow transported by carrier  $k$  through path  $j$ , which is the decision variable of the lower level.

### 3.3. Birandom Variable

The birandom variable, proposed by Peng and Liu [32], can be used to explain the proposed problem. Some basic knowledge about birandom variable is as follows.

*Definition 3.1* (see [32]). A birandom variable  $\xi$  is a mapping from a probability space  $(\Omega, \mathcal{A}, \Pr)$  to a collection  $S$  of random variables such that for any Borel subset  $B$  of the real line  $\mathbb{R}$  the induced function  $\Pr\{\xi(\omega) \in B\}$  is a measurable function with respect to  $\xi$ .

For each given Borel subset  $B$  of the real line  $\mathbb{R}$ , the function  $\Pr\{\xi(\omega) \in B\}$  is a random variable defined on the probability space  $(\Omega, \mathcal{A}, \Pr)$ .

From Definition 3.1, a birandom variable is a mapping from a probability space to a collection of random variables. Roughly speaking, a birandom can be seen as a random variable with random parameter(s). Here we give three examples of birandom variable.

*Example 3.2.* Let  $\Omega = \{\omega_1, \omega_2, \dots, \omega_m\}$ , and  $\Pr\{\omega_1\} + \Pr\{\omega_2\} + \dots + \Pr\{\omega_{m-1}\} + \Pr\{\omega_m\} = 1$ . Assume that  $\xi$  is a function on  $(\Omega, \mathcal{A}, \Pr)$  as follows:

$$\xi(\omega) = \begin{cases} \tilde{\xi}_1, & \text{if } \omega = \omega_1, \\ \tilde{\xi}_2, & \text{if } \omega = \omega_2, \\ \vdots \\ \tilde{\xi}_{m-1}, & \text{if } \omega = \omega_{m-1}, \\ \tilde{\xi}_m, & \text{if } \omega = \omega_m, \end{cases} \quad (3.1)$$

where  $\tilde{\xi}_1$  is a uniformly distributed random variable on  $[0, 1]$ ,  $\tilde{\xi}_2, \dots, \tilde{\xi}_{m-1}$  are normally distributed random variables with a mean 1 and a standard variance 0.5, and  $\tilde{\xi}_m$  is a standard normally distributed random variable with a mean 0 and a standard variance 1, that is,  $\tilde{\xi}_1 \sim U[0, 1]$ ,  $\tilde{\xi}_2 \sim \mathcal{N}[1, 0.5]$ ,  $\dots$ ,  $\tilde{\xi}_{m-1} \sim \mathcal{N}[1, 0.5]$ , and  $\tilde{\xi}_m \sim \mathcal{N}[0, 1]$ . From Definition 3.1,  $\xi$  is clearly a birandom variable as shown in Figure 3.

*Example 3.3.* Assume that  $\bar{a}$  and  $\bar{b}$  are two random variables defined on  $(\Omega', \mathcal{A}', \Pr')$ , and for any  $\omega^* \in \Omega'$ ,  $\bar{b}(\omega^*) \geq \bar{a}(\omega^*)$  holds; then random variable  $\xi \sim \mathcal{N}[\bar{a}(\omega^*), \bar{b}(\omega^*)]$  is a birandom variable.

*Example 3.4.* Let  $\xi$  be a random variable defined on the probability space  $(\Omega, \mathcal{A}, \Pr)$  satisfying  $\xi \in \mathcal{N}(\tilde{\mu}, \sigma^2)$ , where  $\tilde{\mu}$  is also a normally distributed random variable on  $(\Omega', \mathcal{A}', \Pr')$  with the mean  $\mu$  and variance  $\sigma^2$ . Then  $\xi$  is a birandom variable.

*Definition 3.5.* A birandom variable  $\xi$  is said to be normal, if for each  $\omega$ ,  $\xi(\omega)$  is a random variable with the following probability density function:

$$\phi(x) = \frac{1}{\sigma(\omega)\sqrt{2\pi}} \exp\left(-\frac{(x - \mu(\omega))^2}{2\sigma(\omega)^2}\right), \quad (3.2)$$

where the number of random variable of  $\mu(\omega)$  and  $\sigma(\omega)$  is not less than one. The normal birandom variable is denoted by  $\mathcal{N}(\mu(\omega))$  and  $\sigma(\omega)$ .

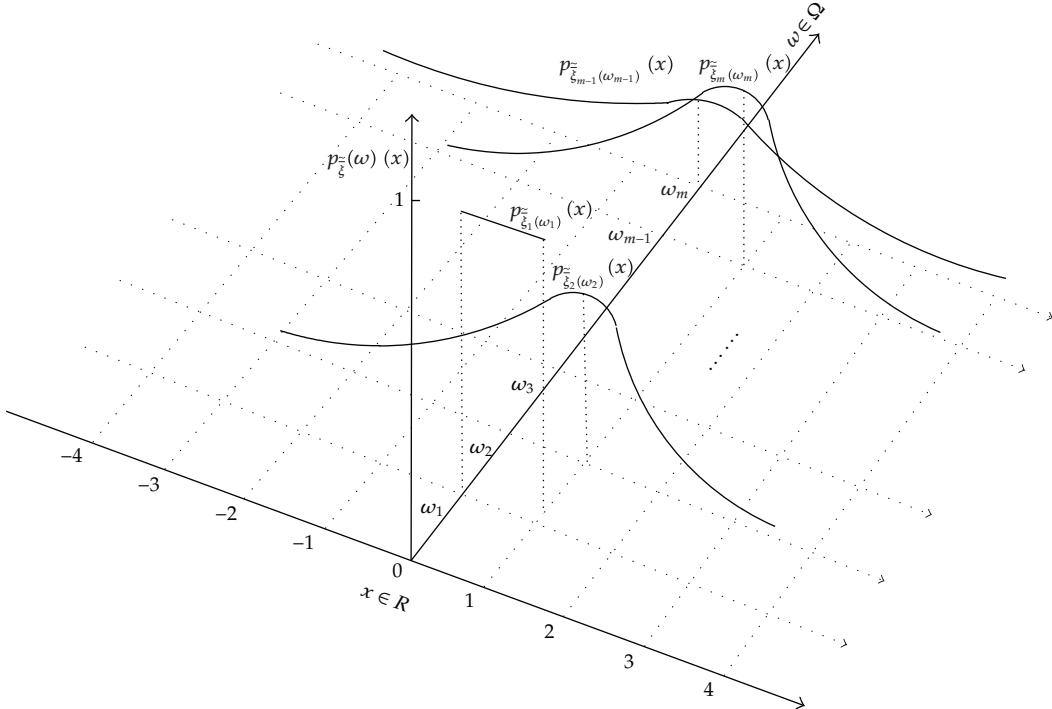


Figure 3: Representation of a Birandom variable in Example 3.2.

### 3.4. Bilevel Model Formulation

In this section, the expectation multiobjective bilevel programming model with chance constraints under a birandom environment based on the philosophy is proposed: decisions are selected by optimizing the expected values of the objective functions subject to chance constraints with some predetermined confidence levels given by the actual decision makers.

#### 3.4.1. Lower Level Model for the Bilevel MCNFP

The problem posed on the lower level is how to make decisions on material flow by each type of carrier on transportation path  $y_{kj}$  while satisfying all capacity constraints, with the main objective being to minimize expected total transportation cost.

##### (1) Objective Functions of the Lower Level

The total transportation cost of the material is calculated by taking the sum of the carriers of each arc's transportation cost and the number of carriers needed to transport the material across the network. In real conditions, it is desirable that each service carrier be fully loaded, so the numbers of carrier  $k$  through path  $j$  can be denoted by  $[y_{kj}/v_k]$ . Since  $\tilde{e}_i^k$  is considered a birandom variable, the total transportation cost of material is considered under a birandom environment. Generally, it is difficult to completely minimize total transportation costs because of the birandom variables. Because decision-makers expect minimal cost,

the expected value of the total transportation cost is the objective of the lower level. Denote the expected total transportation cost of material as  $C(y_{kj})$ ; then the objective function of the lower level model can be formulated as

$$\min C(y_{kj}) = E \left[ \sum_{k \in \Psi} \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \tilde{t}_i^k \left[ \frac{y_{kj}}{v_k} \right] \right]. \quad (3.3)$$

Generally, a path can be represented by a sequence of adjacent arcs. A binary variable  $\gamma_i^j$  is introduced to determine whether an arc  $i$  is a segment of path  $j$  for carrier  $k$ :

$$\gamma_i^j = \begin{cases} 1, & \text{if } i \in A_j, j \in \Omega, \\ 0, & \text{otherwise.} \end{cases} \quad (3.4)$$

## (2) Constraints of the Lower Level

For transportation time, each carrier requires the transport of material from the source to the destination on schedule  $T_j$ . If not, a delay cost is applied.  $\sum_{i \in \Phi} \gamma_i^j \tilde{t}_i^k$  represents the total travel time of carrier  $k$  on transportation path  $j$ , in which  $\tilde{t}_i^k$  is usually represented by a non-decreasing function (i.e., Bureau of Public Roads (BPR) function) [33] as follows:

$$\tilde{t}_i^k = \tilde{t}_{i0}^k \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right], \quad i \in \Phi, k \in \Psi, \quad (3.5)$$

where  $\alpha$  and  $\beta$  are user-defined parameters and, in this problem, are set to 0.15 and 2.0, respectively.

Technically, it is not possible to strictly ensure that the random event  $\sum_{i \in \Phi} \gamma_i^j \tilde{t}_i^k$  does not exceed  $T_j$  because of the birandom variable  $\tilde{t}_{i0}^k$ . In practical problem, the decision makers often provide an appropriate budget  $T_j$  in advance, to ensure that the restriction is, to a certain extent, satisfied, that is to maximize the probability of the random event  $\Pr\{\sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k(\omega) [1 + \alpha (\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]/r_i)^\beta] \leq T_j\}$  under a given confidence level, which can be written as follows:

$$\Pr \left\{ \omega \mid \Pr \left\{ \sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] \leq T_j \right\} \geq \theta \right\} \geq \delta, \quad j \in \Omega, k \in \Psi. \quad (3.6)$$

Here the decision makers' aspiration level is indicated as  $\theta$ , so we use a "Pr" to ensure that the constraint holds at the predetermined confidence level. Additionally, based on probability theory, a further "Pr" is needed to describe the random elements presented in Section 2,

which guarantee the establishment of a certain confidence level  $\delta$ , resembling the P-model (probability maximization model) presented in [34].

The transportation flow may exceed some arcs' capacity because of uncertainties such as the condition of the construction project road. Such conditions may require the manager to select another path. Thus, the total amount of capacity on arc  $i$  cannot exceed the maximal capacity of the arc  $i$ , which produces the following constraint:

$$\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j \left[ \frac{y_{kj}}{v_k} \right] \leq r_i, \quad i \in \Phi. \quad (3.7)$$

Actually, it is difficult to ensure that each service carrier is fully loaded; so the sum of all flows transported by all kinds of carriers through each path cannot be less than the material flow assigned to it; thus the following constraint is obtained:

$$\sum_{k \in \Psi} y_{kj} \geq x_j, \quad j \in \Omega. \quad (3.8)$$

The path flow in path  $j$  used by carrier  $k$  should not be negative, such that

$$y_{kj} \geq 0, \quad j \in \Omega, k \in \Psi. \quad (3.9)$$

### 3.4.2. Upper Level Model for the Bilevel MCNFP

The problem the construction contractor on the upper level faces is how to assign material flow among the transportation paths across the complete transportation network, that is, how to decide the material flow  $x_j$  through transportation path  $j$ . Thus, the decision variable on the upper level is  $x_j$ .

#### (1) Objective Functions of the Upper Level

For large-scale construction projects, cost and time control are both important, so minimizing total direct costs and total transportation time costs is the two objectives of the upper level model. The two objectives of the upper level can be described as follows.

Firstly, the upper level decision maker attempts to minimize the direct costs of the complete network by assigning the flow of the material to each transportation path to achieve a system optimized flow pattern; thus, the total direct cost is the sum of all transportation costs from different transportation paths. The first objective function of the upper level model can be formulated as follows

$$C(x_j, y_{kj}) = \sum_{(o,d) \in E} \sum_{j \in P_{od}} c_j x_j. \quad (3.10)$$

In real conditions, there is increasing pressure to shorter transportation time to reduce or eliminate extra project expenses, with the early arrival of materials shortening the completion time of the construction project and improving construction efficiency. Thus, the total transportation time for carrier  $k$  in each path can be described as  $\sum_{i \in \Phi} \gamma_i^{jk} \tilde{t}_i$ , and, since  $\tilde{t}_i$

is a birandom variable,  $\sum_{i \in \Phi} \gamma_i^j \tilde{t}_i^k$  can be regarded as a special birandom variable. Similarly, the expected value of the total transportation time cost is one of the objectives on the upper level. Different carriers are given different weights, and  $\tilde{t}_i^k = \tilde{t}_{i0}^k [1 + \alpha (\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]/r_i)^\beta]$ , so the second objective function of the upper level can be described as follows:

$$T(x_j, y_{kj}) = E \left[ \sum_{k \in \Psi} w_k \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] \right]. \quad (3.11)$$

## (2) Constraints of the Upper Level

According to the basic assumptions in Section 3.1, it is stipulated that the demands between all OD pairs should be satisfied. Thus,

$$\sum_{j \in P_{od}} x_j = Q_{od}, \quad (o, d) \in E. \quad (3.12)$$

The following constraint ensures that the sum of the weights is equal to 1:

$$\sum_{k \in \Psi} w_k = 1. \quad (3.13)$$

In order to describe the nonnegative variables, the constraints in (3.14) are presented:

$$x_j \geq 0, \quad j \in \Omega. \quad (3.14)$$

### 3.4.3. Global Model for the Bilevel MCNFP

Based on the previous discussion, by integrating (3.3)–(3.14), the following global model for the nonlinear multiobjective bilevel programming with birandom variables is formulated for the MCNFP in a large-scale construction project:

$$\begin{aligned} \min C(x_j, y_{kj}) &= \sum_{(o,d) \in E} \sum_{j \in P_{od}} c_j x_j, \\ \min T(x_j, y_{kj}) &= E \left[ \sum_{k \in \Psi} w_k \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] \right], \end{aligned}$$

$$\begin{aligned}
& \left. \begin{array}{l} \sum_{j \in P_{od}} x_j = Q_{od}, \quad (o, d) \in E, \\ \sum_{k \in \Psi} w_k = 1, \\ x_j \geq 0, \quad j \in \Omega, \\ \min C(y_{kj}) = E \left[ \sum_{k \in \Psi} \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \tilde{e}_i^k \left[ \frac{y_{kj}}{v_k} \right] \right], \\ \text{s.t. } \Pr \left\{ \omega \mid \Pr \left\{ \sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] \leq T_j \right\} \geq \theta \right\} \geq \delta, \\ \text{s.t. } \left\{ \begin{array}{l} \sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j \left[ \frac{y_{kj}}{v_k} \right] \leq r_i, \quad i \in \Phi, \\ \gamma_i^j = \begin{cases} 1, & \text{if } i \in A_j, \ j \in \Omega, \\ 0, & \text{otherwise,} \end{cases} \\ \sum_{k \in \Psi} y_{kj} \geq x_j, \quad j \in \Omega, \\ y_{kj} \geq 0, \quad j \in \Omega, \ k \in \Psi. \end{array} \right. \end{array} \right. \\
& \quad (3.15)
\end{aligned}$$

#### 4. Equivalent Crisp Model

Problem (3.15) is an expectation multiobjective bilevel programming model with chance constraints which has clear nonlinear objectives and constraints. However, this is also difficult to solve with complete confidence because the measures  $E\{\cdot\}$  and  $\Pr\{\cdot\}$  are difficult to obtain. In the following section, the normal birandom variables are focused on and the equivalent crisp model of (3.15) is presented.

*Definition 4.1* (see [35]). Let  $\xi$  be a random variable on the probability space  $(\Omega, \mathcal{A}, \Pr)$ . Then the expected value of  $\xi$  is defined by

$$E[\xi] = \int_0^{+\infty} \Pr\{\xi \geq r\} dr - \int_{-\infty}^0 \Pr\{\xi \leq r\} dr. \quad (4.1)$$

Note that the terms expected value, expectation, and mean value can be used interchangeably.

*Definition 4.2* (see [36]). Let  $\xi$  be a birandom variable on the probability space  $(\Omega, \mathcal{A}, \Pr)$ ; then the expected value of birandom variable  $\xi$  can be defined as follows:

$$E[\xi] = \int_0^{+\infty} \Pr\{\omega \in \Omega \mid E[\xi(\omega)] \geq t\} dt - \int_{-\infty}^0 \Pr\{\omega \in \Omega \mid E[\xi(\omega)] \leq t\} dt, \quad (4.2)$$

provided that at least one of the aforementioned two integrals is finite.

**Lemma 4.3** (see [32]). Let  $\xi$  be a birandom variable on the probability space  $(\Omega, \mathcal{A}, Pr)$ . If the expected value  $E[\xi(\omega)]$  of the random variable  $\xi(\omega)$  is finite for each  $\omega$ , then  $E[\xi(\omega)]$  is a random variable on  $(\Omega, \mathcal{A}, Pr)$ .

*Remark 4.4.* It should be noted that the expected value operator  $E$ , which appears on both sides of the previous definition of  $E[\xi]$ , is overloaded. In fact, symbol  $E$  represents different meanings. That is to say, overloading allows us to use the same symbol  $E$  for different expected value operators, because we can deduce the meaning from the type of argument.

**Lemma 4.5.** Let  $\tilde{\tilde{t}}_{i0}^k(\omega)$  be a normal birandom variable, subject to a normal distribution  $\mathcal{N}(\tilde{\mu}_{t_{i0}^k}(\omega), \sigma_{1t_{i0}^k}^2(\omega))$  where  $\tilde{\mu}_{t_{i0}^k} \sim \mathcal{N}(\mu_{t_{i0}^k}, \sigma_{2t_{i0}^k}^2)$ . Then the objective function in (3.15)

$$T(x_j, y_{kj}) = E \left[ \sum_{k \in \Psi} w_k \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \tilde{\tilde{t}}_{i0}^k \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] \right] \quad (4.3)$$

is transformed into the following equivalent objective function:

$$T(x_j, y_{kj}) = \sum_{k \in \Psi} w_k \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \mu_{t_{i0}^k} \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right]. \quad (4.4)$$

*Proof.* From the assumption, for any  $\omega \in \Omega$ ,  $\tilde{\tilde{t}}_{i0}^k(\omega) \sim \mathcal{N}(\tilde{\mu}_{t_{i0}^k}(\omega), \sigma_{1t_{i0}^k}^2(\omega))$  is an independent random variable, where  $\tilde{\mu}_{t_{i0}^k} \sim \mathcal{N}(\mu_{t_{i0}^k}, \sigma_{2t_{i0}^k}^2)$ . From Definition 4.2,

$$E[\tilde{\tilde{t}}_{i0}^k] = \int_0^{+\infty} \Pr \left\{ \omega \in \Omega \mid E[\tilde{\tilde{t}}_{i0}^k(\omega)] \geq t \right\} dt - \int_{-\infty}^0 \Pr \left\{ \omega \in \Omega \mid E[\tilde{\tilde{t}}_{i0}^k(\omega)] \leq t \right\} dt, \quad (4.5)$$

since  $\tilde{\tilde{t}}_{i0}^k(\omega) \sim \mathcal{N}(\tilde{\mu}_{t_{i0}^k}(\omega), \sigma_{1t_{i0}^k}^2(\omega))$ , and by Definition 4.1, function (4.5) is transformed as follows:

$$E[\tilde{\tilde{t}}_{i0}^k] = \int_0^{+\infty} \Pr \left\{ \tilde{\mu}_{t_{i0}^k}(\omega) \geq t \right\} dt - \int_{-\infty}^0 \Pr \left\{ \tilde{\mu}_{t_{i0}^k}(\omega) \leq t \right\} dt = \mu_{t_{i0}^k}, \quad (4.6)$$

and then

$$\begin{aligned}
T(x_j, y_{kj}) &= E \left[ \sum_{k \in \Psi} w_k \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] \right] \\
&= T(x_j, y_{kj}) = \sum_{k \in \Psi} w_k \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j E \left[ \tilde{t}_{i0}^k \right] \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] \\
&= \sum_{k \in \Psi} w_k \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \mu_{t_{i0}^k} \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right]. \tag{4.7}
\end{aligned}$$

This completes the proof.  $\square$

Let  $\tilde{e}_i^k(\omega)$  be a normal birandom variable subject to a normal distribution  $\mathcal{N}(\tilde{\mu}_{e_i^k}(\omega), \sigma_{1e_i^k}^2(\omega))$  where  $\tilde{\mu}_{e_i^k} \sim \mathcal{N}(\mu_{e_i^k}, \sigma_{2e_i^k}^2)$ . Then similarly the objective function of the lower level is transformed into crisp equivalents:

$$C(y_{kj}) = \sum_{k \in \Psi} \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \mu_{e_i^k} \left[ \frac{y_{kj}}{v_k} \right]. \tag{4.8}$$

**Lemma 4.6.** Assume that  $\tilde{t}_{i0}^k(\omega) \sim \mathcal{N}(\tilde{\mu}_{t_{i0}^k}(\omega), \sigma_{1t_{i0}^k}^2(\omega))$  is a normal birandom variable, where  $\tilde{\mu}_{t_{i0}^k}$  is a normal distributed random variable characterized by  $\tilde{\mu}_{t_{i0}^k} \sim \mathcal{N}(\mu_{t_{i0}^k}, \sigma_{2t_{i0}^k}^2)$ ; then  $\sum_{i \in \Phi} \gamma_i^j \tilde{\mu}_{t_{i0}^k}(\omega) [1 + \alpha (\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]/r_i)^\beta] - T_j \in \mathcal{N}(\mu_r, \sigma_r^2)$  is also a random variable, where

$$\begin{aligned}
\mu_r &= \sum_{i \in \Phi} \gamma_i^j \mu_{t_{i0}^k}(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] - T_j, \\
\sigma_r &= \sqrt{\sum_{i \in \Phi} (\gamma_i^j)^2 \sigma_{2t_{i0}^k}^2 \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right]^2}. \tag{4.9}
\end{aligned}$$

Then the following constraint for the first constraint is derived in (3.15):

$$\Pr \left\{ \omega \mid \Pr \left\{ \sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] \leq T_j \right\} \geq \theta \right\} \geq \delta, \tag{4.10}$$

being equivalent to the following equation:

$$\Phi^{-1}(\theta) \sqrt{\sum_{i \in \Phi} (\gamma_i^j)^2 \sigma_{1t_{i0}^k}^2(\omega)} \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right]^2 + \mu_r + \sigma_r \Phi^{-1}(\delta) \leq 0. \quad (4.11)$$

*Proof.* From the assumption, it is known that for any  $\omega \in \Omega$ ,  $\tilde{t}_{i0}^k(\omega) \sim \mathcal{N}(\tilde{\mu}_{t_{i0}^k}(\omega), \sigma_{1t_{i0}^k}^2(\omega))$  is an independent random variable, so it follows that

$$\sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] - T_j \sim \mathcal{N}(\tilde{\mu}_r(\omega), \tilde{\sigma}_r^2(\omega)) \quad (4.12)$$

is also a normally distributed random variable, where

$$\begin{aligned} \tilde{\mu}_r(\omega) &= \sum_{i \in \Phi} \gamma_i^j \tilde{\mu}_{t_{i0}^k}(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] - T_j, \\ \tilde{\sigma}_r^2(\omega) &= \sum_{i \in \Phi} (\gamma_i^j)^2 \sigma_{1t_{i0}^k}^2(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right]^2. \end{aligned} \quad (4.13)$$

Then

$$\begin{aligned} &\Pr \left\{ \sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] \leq T_j \right\} \geq \theta \\ &\iff \Pr \left\{ \frac{\left( \sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k(\omega) \left[ 1 + \alpha \left( \sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k] / r_i \right)^\beta \right] - T_j \right) - \tilde{\mu}_r(\omega)}{\sigma_r(\omega)} \leq \frac{-\tilde{\mu}_r(\omega)}{\tilde{\sigma}_r(\omega)} \right\} \geq \theta \\ &\iff \Phi \left( \frac{-\tilde{\mu}_r(\omega)}{\tilde{\sigma}_r(\omega)} \right) \geq \theta \iff \tilde{\mu}_r(\omega) \leq -\Phi^{-1}(\theta) \tilde{\sigma}_r(\omega) \\ &\iff \sum_{i \in \Phi} \gamma_i^j \tilde{\mu}_{t_{i0}^k}(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] - T_j \\ &\leq -\Phi^{-1}(\theta) \sqrt{\sum_{i \in \Phi} (\gamma_i^j)^2 \sigma_{1t_{i0}^k}^2(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right]^2}. \end{aligned} \quad (4.14)$$

Since  $\tilde{\mu}_{t_{i0}^k} \sim \mathcal{N}(\mu_{t_{i0}^k}, \sigma_{2t_{i0}^k}^2)$ , then  $\sum_{i \in \Phi} \gamma_i^j \tilde{\mu}_{t_{i0}^k}(\omega) [1 + \alpha (\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k] / r_i)^\beta] - T_j \in \mathcal{N}(\mu_r, \sigma_r^2)$ , where

$$\begin{aligned} \mu_r &= \sum_{i \in \Phi} \gamma_i^j \mu_{t_{i0}^k}(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] - T_j, \\ \sigma_r &= \sqrt{\sum_{i \in \Phi} (\gamma_i^j)^2 \sigma_{2t_{i0}^k}^2 \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right]^2}, \end{aligned} \quad (4.15)$$

for given confidence levels  $\theta, \delta \in [0, 1]$ , so,

$$\begin{aligned} &\Pr \left\{ \omega \mid \Pr \left\{ \sum_{i \in \Phi} \gamma_i^j \tilde{t}_{i0}^k(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] \leq T_j \right\} \geq \theta \right\} \geq \delta \\ &\iff \Pr \left\{ \omega \mid \sum_{i \in \Phi} \gamma_i^j \tilde{\mu}_{t_{i0}^k}(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right] - T_j \leq M \right\} \geq \delta \\ &\iff \Pr \left\{ \omega \mid \frac{\sum_{i \in \Phi} \gamma_i^j \tilde{\mu}_{t_{i0}^k}(\omega) \left[ 1 + \alpha \left( \sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k] / r_i \right)^\beta \right] - T_j - \mu_r}{\sigma_r} \leq \frac{M - \mu_r}{\sigma_r} \right\} \leq \delta \\ &\iff M \geq \mu_r + \sigma_r \Phi^{-1}(\delta) \iff \Phi^{-1}(\theta) \sqrt{\sum_{i \in \Phi} (\gamma_i^j)^2 \sigma_{1t_{i0}^k}^2(\omega) \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right]^2} \\ &\quad + \mu_r + \sigma_r \Phi^{-1}(\delta) \leq 0, \end{aligned} \quad (4.16)$$

where  $M = -\Phi^{-1}(\theta) \sqrt{\sum_{i \in \Phi} (\gamma_i^j)^2 \sigma_{1t_{i0}^k}^2(\omega) \sum_i [1 + \alpha (\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k] / r_i)^\beta]^2}$ .

The proof is completed.  $\square$

Based on Lemmas 4.3 and 4.5, it is determined that the expectation multiobjective bilevel programming model with chance constraints (3.15) is equivalent to the following crisp nonlinear multiobjective bilevel programming problem:

$$\begin{aligned}
\min C(x_j, y_{kj}) &= \sum_{(o,d) \in E} \sum_{j \in P_{od}} c_j x_j, \\
\min T(x_j, y_{kj}) &= \sum_{k \in \Psi} w_k \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \mu_{t_{i0}^k} \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right], \\
\text{s.t. } &\left\{ \begin{array}{l} \sum_{j \in P_{od}} x_j = Q_{od}, \quad (o, d) \in E, \\ \sum_{k \in \Psi} w_k = 1, \\ x_j \geq 0, \quad j \in \Omega, \\ \min C(y_{kj}) = \sum_{k \in \Psi} \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \mu_{e_i^k} \left[ \frac{y_{kj}}{v_k} \right], \\ \text{s.t. } \left\{ \begin{array}{l} \Phi^{-1}(\theta) \sqrt{\sum_{i \in A_j} (\gamma_i^j)^2 \sigma_{1t_{i0}^k}^2(\omega)} \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]}{r_i} \right)^\beta \right]^2 \\ + \mu_r + \sigma_r \Phi^{-1}(\delta) \leq 0, \quad j \in \Omega, k \in \Psi, \\ \sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j \left[ \frac{y_{kj}}{v_k} \right] \leq r_i, \quad i \in \Phi, \\ \gamma_i^j = \begin{cases} 1, & \text{if } i \in A_j, i \in \Omega, \\ 0, & \text{otherwise,} \end{cases} \\ \sum_{k \in \Psi} y_{kj} \geq x_j, \quad j \in \Omega, \\ y_{kj} \geq 0, \quad j \in \Omega, k \in \Psi, \end{array} \right. \end{array} \right. \end{aligned} \tag{4.17}$$

where  $\mu_r = \sum_{i \in \Phi} \gamma_i^j \mu_{t_{i0}^k}(\omega) [1 + \alpha (\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]/r_i)^\beta] - T_j$ ,  $\sigma_r = \sqrt{\sum_{i \in \Phi} (\gamma_i^j)^2 \sigma_{2t_{i0}^k}^2 [1 + \alpha (\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj}/v_k]/r_i)^\beta]^2}$  and  $\theta$  and  $\delta$  are predicted confidence levels specified by the DM.

When (4.17) is used to make decisions, the upper level decision maker first makes a decision  $x_j$ , then the lower decision maker reacts  $y_{kj}$  according to the cost, and the upper level decision maker makes a proper adjustment based on the lower level feedback, finally making the upper level objective optimal. Thus, the upper programming and the lower programming influence and restrict each other in bilevel programming.

## 5. Solution Approach

To obtain an analytical optimal solution for a bilevel programming problem (BLPP) is difficult, yet there is theoretical evidence supporting these observations since BLPP is in fact NP-hard even in its linear form [37]. Moreover, this problem is nonlinear and nondifferentiable, and the MCNFP in a large-scale construction project has various nodes and links. On the other hand, the nondifferentiable piecewise objective functions and constraints

presented in the MCNFP bring computational difficulties. Thus, the possibility of finding a solution to the complexity is increased, and it is difficult to solve with any exact algorithm. Therefore, here, an MOBLPSO is proposed to solve the MCNFP in a large-scale construction project. Because particle swarm optimization (PSO) is computationally tractable compared with other heuristic algorithms, it is easy to implement, and does not require any gradient information for an objective function but the value.

PSO is a population-based self-adaptive search stochastic optimization technique proposed by Kennedy and Eberhart [38], which was inspired by the social behaviour of animals such as fish schooling and birds flocking. Similar to other population-based algorithms, such as evolutionary algorithms, PSO can solve a variety of difficult optimization problems but has shown a faster convergence rate than other evolutionary algorithms on some problems [39]. PSO is influenced little by the continuity of the objective function; it just uses the primary math operators and receives good results in static, noisy, and continuously changing environments [40]. Another advantage of PSO is that it has very few parameters to adjust, which makes it particularly easy to implement. Since PSO can be implemented easily and effectively, it has been applied in solving real-world optimization problems in recent years, such as [41–45]. In PSO, the following formulas [38] are applied to update the position and velocity of each particle:

$$\begin{aligned} v_d^l(\tau_1 + 1) &= w(\tau)v_d^l(\tau_1) + c_p r_1 \left[ p_d^{l,\text{best}}(\tau) - p_d^l(\tau) \right] + c_g r_2 \left[ g_d^{l,\text{best}}(\tau) - p_d^l(\tau) \right], \\ p_d^l(\tau_1 + 1) &= p_d^l(\tau) + v_d^l(\tau + 1), \end{aligned} \quad (5.1)$$

where  $v_d^l$  is the velocity of  $l$ th particle at the  $d$ th dimension in the  $\tau$ th iteration,  $w(\tau)$  is the inertia weight,  $p_d^l(\tau)$  is the position of  $l$ th particle at the  $d$ th dimension,  $r_1$  and  $r_2$  are random numbers in the range  $[0, 1]$ ,  $c_p$  and  $c_g$  are personal and global best position acceleration constants, respectively, while,  $p_d^{l,\text{best}}(\tau)$  is personal best position of the  $l$ th particle at the  $d$ th dimension, and  $g_d^{l,\text{best}}(\tau)$  is the global best position at the  $d$ th dimension.

As mentioned before, it is very difficult to solve the bilevel model, especially when the model is nonlinear. The contribution of this paper is that a universal effective algorithm for solving the bilevel model is put forward, which is based on the hierarchical iteration. The key idea of the algorithm is that the optimum of the bilevel model can be approached step by step through repeatedly iterative calculations between the upper and lower models. The main body of the proposed approach is a type of PSO—multiobjective PSO (MOPSO) with Pareto-Archived Evolution Strategy (PAES)—which is designed only to cope with the upper level programming problem based on the follower's optimal response. Another type of improved PSO—PSO with passive congregation (PSOPC)—is embedded to deal with the lower level programming problem and obtain the optimal response of the follower for each given decision variable of the upper level programming. The follower's optimal reaction is then returned to upper level programming problem as the implementation base for the MOPSO. The algorithm is called the MOBLPSO, the notations of which for the MCNFP are listed as follows:

$\tau_1, \tau_2$ : iteration index of the upper and lower level,  $\tau_1 = 1, 2, \dots, T_1$  and  $\tau_2 = 1, 2, \dots, T_2$ ;

$l_1, l_2$ : particle index of the upper and lower level,  $l_1 = 1, 2, \dots, L_1$  and  $l_2 = 1, 2, \dots, L_2$ ;

$j$ : index of transportation path,  $j = 1, 2, \dots, J$ ;

- $k$ : index of carrier,  $k = 1, 2, \dots, K$ ;
- $r_1, r_2, r_3, r_4, r_5$ : uniform distributed random number within  $[0,1]$ ;
- $w_1(\tau_1)$ : inertia weight in the  $\tau_1$ th iteration of the upper level;
- $w_2(\tau_2)$ : inertia weight in the  $\tau_2$ th iteration of the lower level;
- $R_{kj}^{l_2}(\tau_2)$ : particle selected randomly from the swarm at the  $kj$ th dimension in the  $\tau_2$ th iteration;
- $c_{1p}, c_{1g}$ : personal and global best position acceleration constant of the upper level;
- $c_{2p}, c_{2g}$ : personal and global best position acceleration constant of the lower level;
- $c_{pc}$ : passive congregation coefficient of the lower level;
- $p_{1\max}, p_{1\min}$ : maximum and minimum position value of the upper level;
- $P^{l_1}(\tau_1)$ : vector position of the  $l_1$ th particle in the  $\tau_1$ th iteration,  $P^{l_1}(\tau_1) = [p_1^{l_1}(\tau_1), p_2^{l_1}(\tau_1), \dots, p_J^{l_1}(\tau_1)]$ ;
- $P^{l_2}(\tau_2)$ : vector position of the  $l_2$ th particle in the  $\tau_2$ th iteration,  $P^{l_2}(\tau_2) = [p_1^{l_2}(\tau_2), p_2^{l_2}(\tau_2), \dots, p_{KJ}^{l_2}(\tau_2)]$ ;
- $V^{l_1}(\tau_1)$ : vector velocity of the  $l_1$ th particle in the  $\tau_1$ th iteration,  $V^{l_1}(\tau_1) = [v_1^{l_1}(\tau_1), v_2^{l_1}(\tau_1), \dots, v_J^{l_1}(\tau_1)]$ ;
- $V^{l_2}(\tau_2)$ : vector velocity of the  $l_2$ th particle in the  $\tau_2$ th iteration,  $V^{l_2}(\tau_2) = [v_1^{l_2}(\tau_2), v_2^{l_2}(\tau_2), \dots, v_{KJ}^{l_2}(\tau_2)]$ ;
- $P^{l_1,\text{best}}(\tau_1)$ : vector personal best position of the  $l_1$ th particle in the  $\tau_1$ th iteration,  $P^{l_1,\text{best}}(\tau_1) = [p_1^{l_1,\text{best}}(\tau_1), p_2^{l_1,\text{best}}(\tau_1), \dots, p_J^{l_1,\text{best}}(\tau_1)]$ ;
- $P^{l_2,\text{best}}(\tau_2)$ : vector personal best position of the  $l_2$ th particle in the  $\tau_2$ th iteration,  $P^{l_2,\text{best}}(\tau_2) = [p_1^{l_2,\text{best}}(\tau_2), p_2^{l_2,\text{best}}(\tau_2), \dots, p_{KJ}^{l_2,\text{best}}(\tau_2)]$ ;
- $G^{\text{best}}(\tau_2)$ : vector global best position in the  $\tau_2$ th iteration,  $G^{\text{best}}(\tau_2) = [g_1^{\text{best}}(\tau_2), g_2^{\text{best}}(\tau_2), \dots, g_{KJ}^{\text{best}}(\tau_2)]$ ;
- ARC: the positions of the particles that represent nondominated vectors in the repository;
- $F(P^{l_1}(\tau_1))$ : fitness value of  $P^{l_1}(\tau_1)$ ; and
- $F(P^{l_2}(\tau_2))$ : fitness value of  $P^{l_2}(\tau_2)$ .

### 5.1. Multiobjective Methods for the Upper Level Programming

Researchers are also seeing PSO as a very strong competitor to other algorithms in solving multiobjective optimal problems [46] and it has been proved to be especially suitable for multiobjective optimization [47]. The method applied here to deal with the upper level problem is a multiobjective method which combines a MOPSO with PAES. The PAES [48] is one of the Pareto-based approaches to update the best position. The methods use a truncated archive to store the elite individuals. This approach uses leader selection techniques based on Pareto dominance. The basic idea is to select as leaders to the particles that are nondominated with respect to the swarm.

**Procedure** The updating of the best position in the subsequent iterations generate an initial random solution  $p_j^{l_1,\text{best}}(\tau_1)$  and add it to the archive; update  $p_j^{l_1,\text{best}}(\tau_1)$  to produce  $p_j^{l_1}(\tau_1 + 1)$  and evaluate  $p_j^{l_1}(\tau_1 + 1)$ ; **If** ( $p_j^{l_1,\text{best}}(\tau_1)$  dominates  $p_j^{l_1}(\tau_1 + 1)$ ) **then** discard  $p_j^{l_1}(\tau_1 + 1)$ ; **else if** ( $p_j^{l_1}(\tau_1 + 1)$  dominates  $p_j^{l_1,\text{best}}(\tau_1)$ ) **then** replace  $p_j^{l_1,\text{best}}(\tau_1)$  with  $p_j^{l_1}(\tau_1 + 1)$ , and add  $p_j^{l_1}(\tau_1 + 1)$  to the archive  $p_j^{l_1,\text{best}}(\tau_1 + 1) = p_j^{l_1}(\tau_1 + 1)$ ; **else if** ( $p_j^{l_1}(\tau_1 + 1)$  is dominated by any member of the archive) then discard  $p_j^{l_1}(\tau_1 + 1)$ ; **else** apply test ( $p_j^{l_1,\text{best}}(\tau_1)$ ,  $p_j^{l_1}(\tau_1 + 1)$ , archive) to determine which becomes the new current solution and whether to add  $p_j^{l_1}(\tau_1 + 1)$  to the archive; until a termination criterion has been reached, return to line 2.

**Algorithm 1**

**Procedure** Test  
**if** the archive is not full;  
  add  $p_j^{l_1}(\tau_1 + 1)$  to the archive;  
  **if** ( $p_j^{l_1}(\tau_1 + 1)$  is in a less crowded region of the archive than  $p_j^{l_1,\text{best}}(\tau_1)$ );  
    accept  $p_j^{l_1,\text{best}}(\tau_1 + 1) = p_j^{l_1}(\tau_1 + 1)$ ;  
  **else** maintain  $p_j^{l_1,\text{best}}(\tau_1 + 1) = p_j^{l_1,\text{best}}(\tau_1)$ ;  
**else**  
  **If** ( $p_j^{l_1}(\tau_1 + 1)$  is in a less crowded region of the archive than  $p_j^{l_1,\text{best}}(\tau_1)$  for some member  $p_j^{l_1,\text{best}}(\tau_1)$  on the archive);  
    add  $p_j^{l_1}(\tau_1 + 1)$  to the archive, and remove a member of the archive from the most crowded region;  
    **if** ( $p_j^{l_1}(\tau_1 + 1)$  is in a less crowded region of the archive than  $p_j^{l_1,\text{best}}(\tau_1)$ );  
      accept  $p_j^{l_1,\text{best}}(\tau_1 + 1) = p_j^{l_1}(\tau_1 + 1)$ ;  
      **else** maintain  $p_j^{l_1,\text{best}}(\tau_1 + 1) = p_j^{l_1,\text{best}}(\tau_1)$ ;  
  **else**  
    **if** ( $p_j^{l_1}(\tau_1 + 1)$  is in a less crowded region of the archive than  $p_j^{l_1,\text{best}}(\tau_1)$ );  
      accept  $p_j^{l_1,\text{best}}(\tau_1 + 1) = p_j^{l_1}(\tau_1 + 1)$ ;  
      **else** maintain  $p_j^{l_1,\text{best}}(\tau_1 + 1) = p_j^{l_1,\text{best}}(\tau_1)$ .

**Algorithm 2**

The details for the PAES procedure, test procedure, and selection procedure are stated hereinafter (Algorithms 1 and 2), in which  $P_j^{l_1,\text{best}}(\tau_1)$  is initially set equal to the initial position of particle  $l_1$ .

The repository that stores the positions of the particles that represent nondominated vectors is denoted by ARC; then the velocity of each  $l_1$ th particle of the upper level is updated using the following equations:

$$v_j^{l_1}(\tau_1 + 1) = w_1(\tau_1)v_j^{l_1}(\tau_1) + c_{1p}r_1 \left[ p_j^{l_1,\text{best}}(\tau_1) - p_j^{l_1}(\tau_1) \right] + c_{1g}r_2 \left[ \text{ARC}_h(\tau_1) - p_j^{l_1}(\tau_1) \right], \quad (5.2)$$

where  $\text{ARC}_h(\tau_1)$  is a solution randomly selected from the repository in iteration  $\tau_1$ , which can improve significantly the ability of global convergence by avoiding being trapped in

a stagnant state in finite iterations [49, 50]. The index  $h$  is selected in the following way: the hypercubes containing more than one particle are assigned a fitness equal to the result of dividing any number into the number of particles they contain. This aims to decrease the fitness of those hypercubes that contain more particles which is seen as a form of fitness sharing [51]. Then, a roulette-wheel selection is applied using these fitness values to select the hypercube from which the corresponding particle is taken. Once the hypercube has been selected, a particle is selected randomly within the hypercube.

## 5.2. PSOPC for the Lower Level Programming

From the previous description, because there are many constraints in the lower level, the standard PSO can easily fall into premature convergence, so a PSOPC based on the standard PSO is adopted to solve the lower level problem. He et al. [52] proved that PSOPC can avoid the premature convergence problem, in the running of the PSO algorithm by adding a passive congregation coefficient to the standard PSO, as this helps the algorithm move out of the local optimal solution improving the convergence of the algorithm, and thus improving the global search ability. The following equation is adopted to update the velocity of each  $l_2$ th particle of the lower level:

$$\begin{aligned} v_{kj}^{l_2}(\tau_2 + 1) = & w_2(\tau_2)v_{kj}^{l_2}(\tau_2) + c_{2p}r_3 \left[ p_{kj}^{l_2,\text{best}}(\tau_2) - p_{kj}^{l_2}(\tau_2) \right] \\ & + c_{2g}r_4 \left[ g_{kj}^{\text{best}}(\tau_2) - p_{kj}^{l_2}(\tau_2) \right] + c_{\text{pc}}r_5 \left( R_{kj}^{l_2}(\tau_2) - p_{kj}^{l_2}(\tau_2) \right). \end{aligned} \quad (5.3)$$

## 5.3. Framework of the Proposed MOBLPSO for the MCNFP

In our proposed algorithm, solving multiobjective bilevel programming is transformed to solve the upper level and lower level programming problem interactively while determining, respectively, the decision variable of the upper level or the lower level. To be mentionable, the PSOPC for the lower level is nested in the MOPSO for the upper level, and the MOPSO is the main body of the MOBLPSO. The solution information is exchanged between the two types of PSO, and the output of one algorithm is the input of another algorithm, namely,  $y$ , the output of PSOPC for lower level programming is the input of the MOPSO for upper level programming; and  $x$ , the output of the MOPSO for the upper level programming is the input of PSOPC for the lower level programming. These form a hierarchical and sequential framework.

### Parameter Selection

In order to guarantee the convergence of MOBLPSO, the parameters are selected on the basis of empirical results that are carried out to observe the behaviour of the algorithm in different parameter settings. By comparing several sets of parameters, including population size, iteration number, acceleration coefficients, and inertia weight, the empirical results have shown that the constant acceleration coefficients with  $c_{1p} = c_{1g} = 1.5$  for the upper level,  $c_{2p} = 1$ ,  $c_{2g} = 2$ , and  $c_{\text{pc}} = 1.5$  (i.e., passive congregation coefficient [52]) for the lower level, and the adaptive inertia weights [53] provide good convergent behaviour in this study, which is in accordance with the results provided by Eberhart and Shi [54]. The adaptive inertia

weights for the upper level (i.e.,  $e = 1$ ) and lower level (i.e.,  $e = 2$ ) are set to be varying with iteration as follows:

$$w_e(\tau_e) = w_e(T_e) - \frac{\tau_e - T_e}{1 - T_e} [w_e(1) - w_e(T_e)], \quad (5.4)$$

where the iteration numbers are  $T_1 = 500$  (i.e., for the upper level) and  $T_2 = 100$  (i.e., for the lower level), and  $w_e(1) = 0.9$  and  $w_e(T_e) = 0.1$  (for  $e = 1, 2$ ). Since the probability of becoming trapped in the stagnant state can be reduced dramatically by using a large number of particles [49], the population sizes are set to be 300 for the upper level and 100 for the lower level.

### Initialize

In the upper level, set iteration  $\tau_1 = 1$ . For  $l_1 = 1, 2, \dots, L_1$ , generate the position of the particle  $l_1$  with an integer random position (note that every particle in the upper level consists of  $j$  dimensions in this study). In the lower level, set iteration  $\tau_2 = 1$ . For  $l_2 = 1, 2, \dots, L_2$ , generate the position of the particle  $l_2$  with an integer random position (note that every particle in the lower level consists of  $k \times j$  dimensions in this study).

### Decode Particles into Solutions

Decode particles into solutions: for  $l_1 = 1, 2, \dots, L_1$ , decode  $p_j^{l_1}(\tau_1)$  to a solution as  $p_j^{l_1}(\tau_1) = x_j(\tau_1)$ . For  $l_2 = 1, 2, \dots, L_2$ , decode  $p_{kj}^{l_2}(\tau_2)$  to a solution as  $p_{kj}^{l_2}(\tau_2) = y_{kj}(\tau_2)$ . Mapping between one potential solution for the upper and lower level of the MCNFP and particle representation is shown in Figure 4.

### Check the Feasibility

All particles of the upper level satisfy the constraints of the upper level. All particles of the lower level satisfy the constraints of the lower level. Then, the particles are feasible.

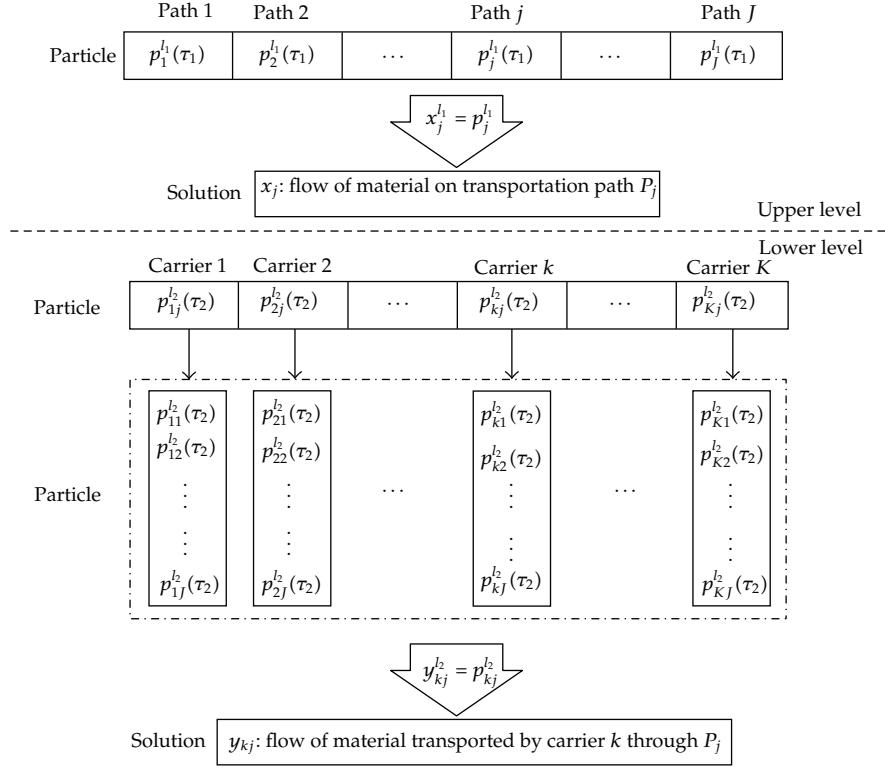
### Fitness Value

The fitness value used to evaluate the particle is the value of objective function in each level. There are two objectives in the upper level; particle  $P^{l_1}(\tau_1)$ 's fitness value  $F(P^{l_1}(\tau_1))$  is a  $1 \times 2$  matrix, namely,

$$F(P^{l_1}(\tau_1)) = \left[ \sum_{(o,d) \in E} \sum_{j \in P_{od}} c_j x_j, \sum_{k \in \Psi} w_k \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \mu_{e_i^k} \left[ 1 + \alpha \left( \frac{\sum_{j \in \Omega} \sum_{k \in \Psi} \gamma_i^j [y_{kj} / v_k]}{r_i} \right)^\beta \right] \right]. \quad (5.5)$$

The fitness value of the particle in the lower level is

$$F(P^{l_2}(\tau_2)) = \sum_{k \in \Psi} \sum_{j \in \Omega} \sum_{i \in \Phi} \gamma_i^j \mu_{e_i^k} \left[ \frac{y_{kj}}{v_k} \right]. \quad (5.6)$$



**Figure 4:** Decoding method and mapping between PSO particles and solutions of two levels.

Figure 5 shows the schematic procedure for the MOBLPSO to generate solutions for the proposed multiobjective bilevel model. Such a repeated interaction and cooperation between two types of PSO reflects and simulates the decision process of multiobjective bilevel programming and is able to solve multiobjective bilevel programming sequentially.

## 6. Practical Application

### 6.1. Project Description

In this section, an earth-rock work transportation project in a large-scale water conservancy and hydropower construction project is taken as an example for our optimization method. The Shuibuya Hydropower Project was conducted in Badong County, which is located in the middle reaches of Qingjiang River in Sichuan province, China. The project is the first cascaded project on the Qingjiang main stream and the third important project following Geheyuan and Gaobazhou in China. Once completed, it will provide a major power source to meet the peak load demand in the Central China Power Grid. The installed capacity and annual output of Shuibuya Power Plant are 1,600 MW and 3.92 GWh, respectively. The project has a powerful regulating ability with a normal pool level of 400 m and reservoir capacity of  $4.58 \times 10^9$  m<sup>3</sup>. The project consists of a concrete-faced rock fill dam (CFRD), underground power house, a chute spillway on the left bank, and the sluice tunnel on the right bank. The dam is 233 m high and is the tallest of its kind in the world at present with a total volume of  $15.64 \times 10^6$  m<sup>3</sup>.

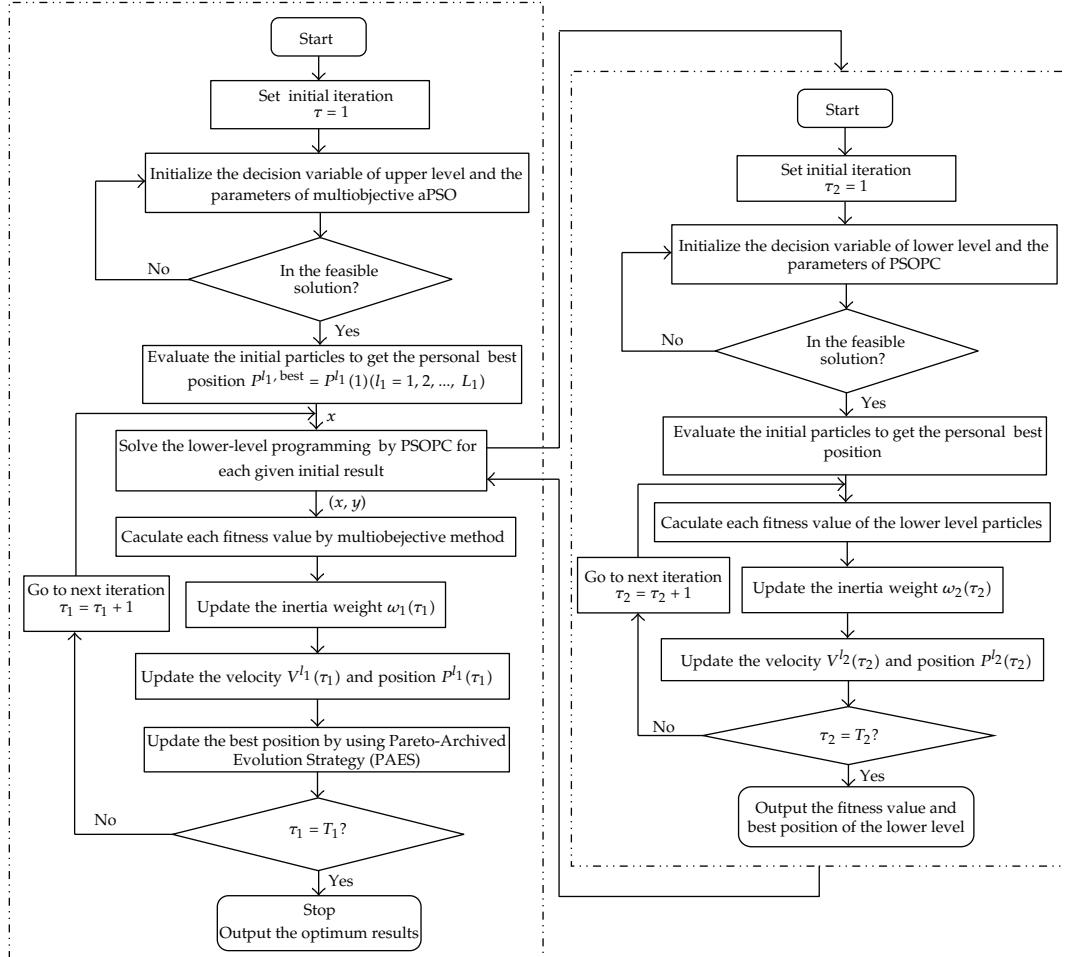
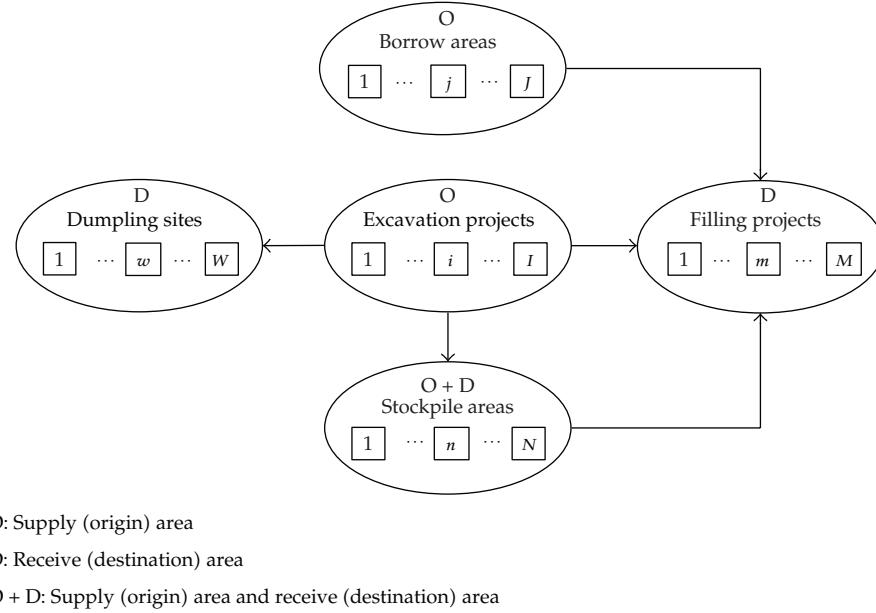


Figure 5: Schematic diagram of MOBLPSO for MCNFP.

## 6.2. Data Collection

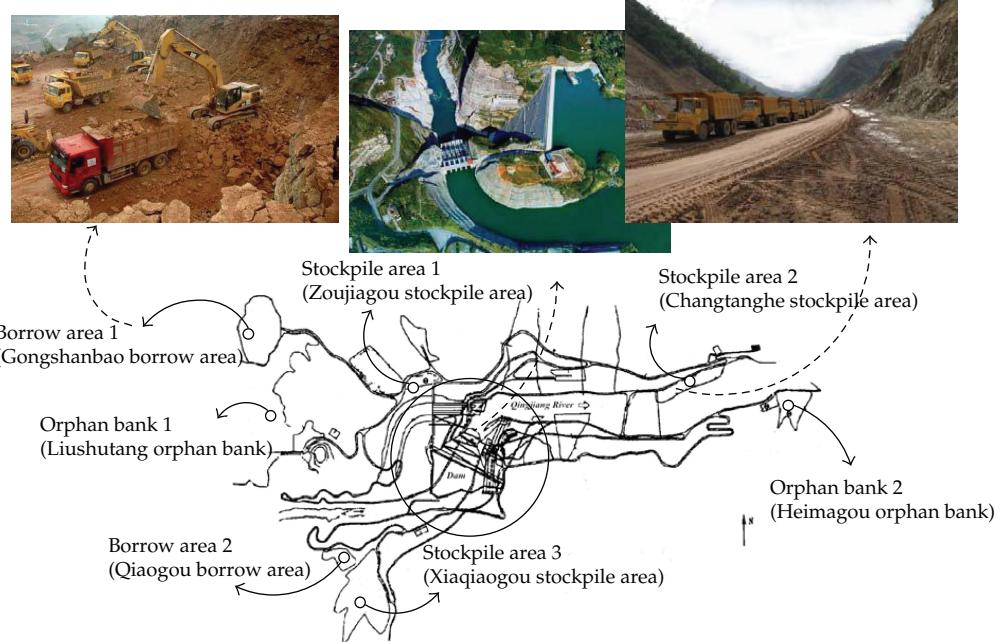
All detailed data for the Shuibuya Hydropower Project were obtained from the Hubei Qingjiang Shuibuya Project Construction Company. In a large-scale construction project, especially in a large water conservancy and hydropower construction project, earth-rock work is usually a primary material, and earth-rock work transportation occurs every day in excavation projects, borrow areas, filling projects, dumpling sites, and stockpile areas as it is turned over and needs to be replaced frequently (see Figure 6). In the Shuibuya Hydropower Project, there are 4 excavation projects, 2 borrow areas, 3 stockpile areas, and 2 dumpling sites. The location and detailed information of borrow areas, dumpling sites, and stockpile areas, of the Shuibuya Hydropower Project is illustrated in Figure 7.

Three types of dump trucks (carriers) in the construction project are considered, which transport earth-rock work along different paths connecting the OD pairs, with the destination nodes having the practical demand of timeliness. All necessary data for every kind of carrier were calculated as shown in Table 1, and Table 2 shows the details of the paths in the whole road network.



**Figure 6:** Earth-rock work transportation relations in large-scale construction project.

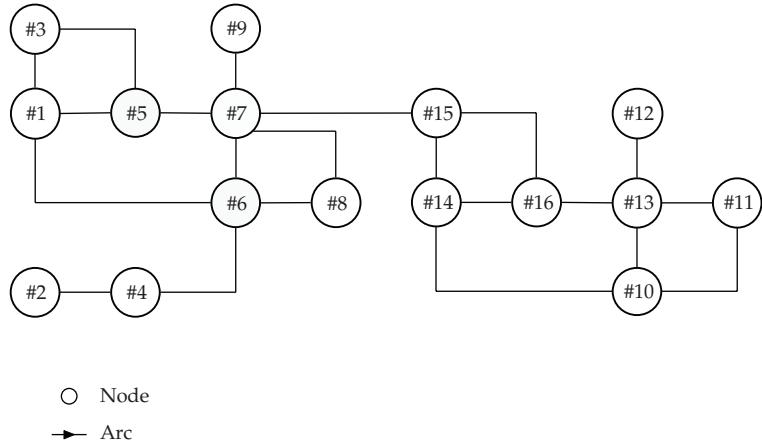
The transportation network in the project includes an internal road network and an external road network. In this case, only the internal road network is considered. The internal road network is composed of 20 trunk roads located on the left and right banks, with 11 on the left and 9 on the right, with a cross-river bridge connecting the left and right banks. Therefore, 21 links are considered in this paper. In order to apply the proposed methods more conveniently, adjacent roads of the same type have been combined and road shapes have been ignored. An abstracted transportation network is illustrated in Figure 8. For each link in the transportation network, there are a free flow birandom travel time  $\tilde{t}_{i0}^k$  and birandom transportation cost  $\tilde{e}_i^k$ . The corresponding data of which are stated in Table 3. In order to collect the data of transportation time and costs, investigations and surveys were made to obtain historical data from the financial department and experienced engineers of construction team in Hubei Qingjiang Shuibuya Project Construction Company. Since the transportation time and costs for each arc of the path change over time, the data are classified into different parts based on different periods. Both transportation time and costs are assumed to approximately follow normal distributions for each period, and the two parameters (expected value and variance) for the normal distributions are estimated by using maximum likelihood estimation, which justified by a chi-square goodness-of-fit test. By comparing the normal distributions for the same transportation time and costs in different periods, it can be found that the expected values of the aforementioned normal distributions also approximately follow a similar random distribution pattern, which has also been justified using a chi-square goodness-of-fit test. It should be noted that since the variance fluctuations are quite insignificant, the median values of the variances in different periods are selected as the variance for the previous normal distribution. The predicted confidence levels given by the decision maker are, respectively,  $\theta = 0.9$  and  $\delta = 0.85$ .



**Figure 7:** Layout of borrow areas, dumping sites and stockpile areas of Shuibuya Hydropower Project.

**Table 1:** Information of carriers in Shuibuya Hydropower Project.

Carrier	Kind index $k$	Type (heaped capacity; maximum payload)	Weight $w_k$
Dump truck	1	K30N-8.4 (16 m <sup>3</sup> ; 26 t)	0.3
	2	Terex TA28 (17 m <sup>3</sup> ; 28 t)	0.3
	3	Perlini DP366 (20 m <sup>3</sup> ; 36 t)	0.4



**Figure 8:** Illustrations of road network of Shuibuya Hydropower Project.

**Table 2:** Related data about information of paths in earth-rock work transportation network.

OD pairs	Demand $Q_{od}$ ( $\text{m}^3$ )	Transportation path $j$	Direct cost (CNY/ $\text{m}^3$ )	Constraint time (h)
Excavation project 1,2 → dumping site 1	950	1 ⇔ #5 → #7 → #9 2 ⇔ #5 → #3 → #1 → #6 3 ⇔ #5 → #1 → #6 4 ⇔ #5 → #7 → #8 → #6 5 ⇔ #5 → #7 → #6 6 ⇔ #5 → #7	1.80 2.50 1.70 2.40 1.90 1.20	1.45 2.15 1.35 2.20 1.40 1.25
Excavation project 1,2 → filling project 1,2	2650	7 ⇔ #11 → #10 → #14 → #16 8 ⇔ #11 → #13 → #16 9 ⇔ #11 → #10 → #13 → #16 10 ⇔ #11 → #10 → #14 → #15 → #16	2.30 1.90 2.70 3.00	2.25 1.45 2.30 3.50
Excavation project 1,2 → stockpile area 1	1700	11 ⇔ #11 → #13 → #12 12 ⇔ #11 → #10 → #13 → #12	1.60 2.33	1.50 2.50
Excavation project 3,4 → filling project 3,4	2300	13 ⇔ #11 → #10 → #13 14 ⇔ #11 → #13	2.60 1.10	1.75 1.25
Excavation project 3,4 → dumping site 2	1650	15 ⇔ #11 → #10 → #14 → #15 16 ⇔ #2 → #4 → #6 17 ⇔ #14 → #15 → #16	2.10 1.50 1.80	2.33 1.50 1.55
Excavation project 3,4 → stockpile area 2	1480	18 ⇔ #14 → #16 19 ⇔ #7 → #8 → #6 20 ⇔ #13 → #16 21 ⇔ #15 → #7 → #8 → #6 22 ⇔ #15 → #7 → #6	1.20 1.90 1.40 2.40 2.20	1.20 1.45 1.17 2.17 1.60
Borrow area 1 → filling project 1,2	1700			
Borrow area 2 → filling project 3,4	1250			
Stockpile area 1 → filling project 1,2	1150			
Stockpile area 2 → filling project 3,4	1300			
Stockpile area 3 → filling project 1,2	1260			

**Table 3:** Birandom free flow travel time  $\tilde{t}_0^k$  and birandom transportation cost  $\tilde{e}_i^k$ .

Arc $i$	Corresponding nodes	Arc capacity $r_i$ (n/h)	Transportation cost $e_i^k$ (CNY/unit)			Free transportation time $t_{t_0}^k$ (h)
			Carrier 1	Carrier 2	Carrier 3	
1	#1, #3	105	$\mathcal{N}(\mu, 0.64)$ , with $\mu \sim \mathcal{N}(5.2, 0.10)$	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(6.0, 0.09)$	$\mathcal{N}(\mu, 0.25)$ , with $\mu \sim \mathcal{N}(6.8, 1.00)$	$\mathcal{N}(\mu, 0.16)$ , with $\mu \sim \mathcal{N}(0.24, 0.09)$
2	#1, #5	110	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(3.2, 0.64)$	$\mathcal{N}(\mu, 0.49)$ , with $\mu \sim \mathcal{N}(3.5, 0.16)$	$\mathcal{N}(\mu, 0.36)$ , with $\mu \sim \mathcal{N}(3.8, 0.25)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.18, 0.01)$
3	#1, #6	135	$\mathcal{N}(\mu, 4.00)$ , with $\mu \sim \mathcal{N}(7.2, 1.00)$	$\mathcal{N}(\mu, 0.81)$ , with $\mu \sim \mathcal{N}(7.5, 0.64)$	$\mathcal{N}(\mu, 0.25)$ , with $\mu \sim \mathcal{N}(7.8, 0.49)$	$\mathcal{N}(\mu, 0.16)$ , with $\mu \sim \mathcal{N}(0.38, 0.09)$
4	#2, #4	115	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(3.7, 0.49)$	$\mathcal{N}(\mu, 4.00)$ , with $\mu \sim \mathcal{N}(4.0, 0.81)$	$\mathcal{N}(\mu, 0.81)$ , with $\mu \sim \mathcal{N}(4.3, 0.01)$	$\mathcal{N}(\mu, 0.36)$ , with $\mu \sim \mathcal{N}(0.19, 0.01)$
5	#3, #5	108	$\mathcal{N}(\mu, 0.36)$ , with $\mu \sim \mathcal{N}(2.2, 0.01)$	$\mathcal{N}(\mu, 0.81)$ , with $\mu \sim \mathcal{N}(2.4, 0.09)$	$\mathcal{N}(\mu, 0.64)$ , with $\mu \sim \mathcal{N}(2.6, 0.16)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.22, 0.01)$
6	#4, #6	112	$\mathcal{N}(\mu, 0.81)$ , with $\mu \sim \mathcal{N}(3.2, 0.16)$	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(4.0, 0.09)$	$\mathcal{N}(\mu, 0.49)$ , with $\mu \sim \mathcal{N}(4.8, 0.09)$	$\mathcal{N}(\mu, 0.25)$ , with $\mu \sim \mathcal{N}(0.19, 0.01)$
7	#5, #7	180	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(5.0, 0.01)$	$\mathcal{N}(\mu, 0.64)$ , with $\mu \sim \mathcal{N}(5.2, 0.25)$	$\mathcal{N}(\mu, 0.36)$ , with $\mu \sim \mathcal{N}(5.4, 0.16)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.28, 0.01)$
8	#6, #7	145	$\mathcal{N}(\mu, 0.49)$ , with $\mu \sim \mathcal{N}(4.5, 0.01)$	$\mathcal{N}(\mu, 0.36)$ , with $\mu \sim \mathcal{N}(4.2, 0.16)$	$\mathcal{N}(\mu, 0.49)$ , with $\mu \sim \mathcal{N}(3.4, 0.16)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.18, 0.01)$
9	#6, #8	160	$\mathcal{N}(\mu, 0.16)$ , with $\mu \sim \mathcal{N}(3.6, 0.25)$	$\mathcal{N}(\mu, 0.81)$ , with $\mu \sim \mathcal{N}(4.0, 1.00)$	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(4.4, 0.09)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(0.32, 0.01)$
10	#7, #8	165	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(4.0, 0.81)$	$\mathcal{N}(\mu, 0.64)$ , with $\mu \sim \mathcal{N}(4.4, 0.49)$	$\mathcal{N}(\mu, 0.64)$ , with $\mu \sim \mathcal{N}(4.8, 0.25)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(0.17, 0.01)$
11	#7, #9	100	$\mathcal{N}(\mu, 4.00)$ , with $\mu \sim \mathcal{N}(7.2, 0.09)$	$\mathcal{N}(\mu, 0.16)$ , with $\mu \sim \mathcal{N}(7.6, 0.81)$	$\mathcal{N}(\mu, 0.25)$ , with $\mu \sim \mathcal{N}(8.0, 0.16)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.42, 0.01)$
12	#7, #15	130	$\mathcal{N}(\mu, 0.81)$ , with $\mu \sim \mathcal{N}(8.0, 0.81)$	$\mathcal{N}(\mu, 0.64)$ , with $\mu \sim \mathcal{N}(8.5, 0.50)$	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(8.7, 0.01)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(0.36, 0.01)$
13	#10, #11	225	$\mathcal{N}(\mu, 4.00)$ , with $\mu \sim \mathcal{N}(5.0, 0.81)$	$\mathcal{N}(\mu, 0.64)$ , with $\mu \sim \mathcal{N}(6.0, 0.16)$	$\mathcal{N}(\mu, 0.25)$ , with $\mu \sim \mathcal{N}(6.5, 0.09)$	$\mathcal{N}(\mu, 0.01)$ , with $\mu \sim \mathcal{N}(0.36, 0.01)$
14	#10, #13	155	$\mathcal{N}(\mu, 0.81)$ , with $\mu \sim \mathcal{N}(5.5, 0.36)$	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(5.6, 0.49)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(5.8, 0.16)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(0.30, 0.09)$
15	#10, #14	150	$\mathcal{N}(\mu, 0.64)$ , with $\mu \sim \mathcal{N}(4.5, 0.49)$	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(4.6, 0.25)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(4.8, 0.09)$	$\mathcal{N}(\mu, 0.01)$ , with $\mu \sim \mathcal{N}(0.27, 0.01)$
16	#11, #13	170	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(3.6, 0.16)$	$\mathcal{N}(\mu, 0.16)$ , with $\mu \sim \mathcal{N}(4.0, 0.01)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(4.4, 0.25)$	$\mathcal{N}(\mu, 0.01)$ , with $\mu \sim \mathcal{N}(0.20, 0.01)$

Table 3: Continued.

Arc $i$	Corresponding nodes	Arc capacity $r_i$ (n/h)	Transportation cost $e_i^k$ (CNY/unit)	Carrier 1	Carrier 2	Carrier 3	Free transportation time $t_0^k$ (h)	Carrier 2	Carrier 3
17	#12, #13	125	$\mathcal{N}(\mu, 0.25)$ , with $\mu \sim \mathcal{N}(4.2, 0.16)$	$\mathcal{N}(\mu, 0.49)$ , with $\mu \sim \mathcal{N}(4.3, 0.09)$	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(4.4, 0.16)$	$\mathcal{N}(\mu, 0.01)$ , with $\mu \sim \mathcal{N}(0.22, 0.01)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(0.24, 0.09)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.26, 0.01)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.26, 0.01)$
18	#13, #16	145	$\mathcal{N}(\mu, 0.100)$ , with $\mu \sim \mathcal{N}(4.5, 0.16)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(4.7, 0.81)$	$\mathcal{N}(\mu, 0.49)$ , with $\mu \sim \mathcal{N}(4.9, 1.00)$	$\mathcal{N}(\mu, 0.01)$ , with $\mu \sim \mathcal{N}(0.21, 0.64)$	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(0.23, 0.01)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.25, 0.16)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.25, 0.16)$
19	#14, #15	140	$\mathcal{N}(\mu, 0.25)$ , with $\mu \sim \mathcal{N}(5.6, 0.49)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(6.0, 0.01)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(6.5, 0.04)$	$\mathcal{N}(\mu, 0.01)$ , with $\mu \sim \mathcal{N}(0.30, 0.01)$	$\mathcal{N}(\mu, 0.16)$ , with $\mu \sim \mathcal{N}(0.33, 0.01)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.35, 0.04)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.35, 0.04)$
20	#14, #16	135	$\mathcal{N}(\mu, 0.36)$ , with $\mu \sim \mathcal{N}(4.6, 0.25)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(4.8, 0.16)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(5.0, 0.01)$	$\mathcal{N}(\mu, 0.09)$ , with $\mu \sim \mathcal{N}(0.18, 0.04)$	$\mathcal{N}(\mu, 0.16)$ , with $\mu \sim \mathcal{N}(0.22, 0.09)$	$\mathcal{N}(\mu, 0.25)$ , with $\mu \sim \mathcal{N}(0.25, 0.49)$	$\mathcal{N}(\mu, 0.25)$ , with $\mu \sim \mathcal{N}(0.25, 0.49)$
21	#15, #16	140	$\mathcal{N}(\mu, 1.00)$ , with $\mu \sim \mathcal{N}(5.8, 0.09)$	$\mathcal{N}(\mu, 0.81)$ , with $\mu \sim \mathcal{N}(6.2, 0.16)$	$\mathcal{N}(\mu, 0.01)$ , with $\mu \sim \mathcal{N}(6.5, 0.04)$	$\mathcal{N}(\mu, 0.16)$ , with $\mu \sim \mathcal{N}(0.30, 0.09)$	$\mathcal{N}(\mu, 0.25)$ , with $\mu \sim \mathcal{N}(0.32, 0.25)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(0.35, 0.01)$	$\mathcal{N}(\mu, 0.04)$ , with $\mu \sim \mathcal{N}(0.35, 0.01)$

### 6.3. Computational Results

To verify the practicality and efficiency of the MCNFP optimization model under a birandom environment presented previously, the proposed MOBLPSO was implemented to determine the flow assignment amongst the transportation paths and amongst the carriers over a certain period using actual data from the Hubei Qingjiang Shuibuya Project Construction Company. After running the proposed MOBLPSO using MATLAB 7.0, the computational results were obtained and the efficiency of the proposed algorithm was proven.

The computer running environment was an intel core 2 Duo 2.26GHz clock pulse with 2048 MB memory. The problem was solved using the proposed algorithm with satisfactory solutions within 21 minutes on average, and the optimal solutions for lower level programming and the Pareto optimal solution set for upper level programming were worked out.

The red dots in Figure 9 show the Pareto optimal solutions, while the blue dots show the best position of particles in this iteration. The decision maker can choose a plan from these Pareto optimal solutions depending on their preference. For example, if the decision makers feel that the total direct cost objective is more important, they may sacrifice transportation time for more economical scheme. Thus they choose the absolute left of the Pareto optimal solution in the minimum cost network flow plan in Table 4. On the contrary, if decision makers feel that the transportation time objective is more important, they may choose minimum total transportation time cost and sacrifice more costs for the MCNFP, choosing the lowest of the Pareto optimal solutions. The minimum transportation time plan is in Table 5.

### 6.4. Model Comparison to an MCNFP with Single Randomness

To highlight the advantages of our mathematical model (4.17), additional computational work was done using the proposed MOBLPSO to solve a similar MCNFP under a different uncertain environment, that is, the random environment. To carry out comparisons under a similar circumstance, analyses were conducted based on results from running the test problem 10 times. A detailed analysis follows.

In order to guarantee a fair comparison between the MCNFP model with birandomness (denoted by MCNFP-birm) and a model just considering single randomness (denoted by MCNFP-rm), the random distribution for each related uncertain parameter in the MCNFP-rm was selected in the following way. Take the transportation cost  $\tilde{\bar{e}}_1^1$  (i.e.,  $\tilde{\bar{e}}_1^1 \sim \mathcal{N}(\mu, 0.64)$ , with  $\mu \sim \mathcal{N}(5.2, 0.10)$ ), for example. The stochastic nature of the expected value  $\mu$  in the normal distribution  $\mathcal{N}(\mu, 0.64)$  was ignored by using its expectation 5.2 as a representation while the variance 0.64 was retained. Thus, for the MCNFP-rm, the birandomness of the transportation cost  $\tilde{\bar{e}}_1^1$  degenerated to a single randomness, in which the distribution of the transportation cost could be expressed as  $\mathcal{N}(5.2, 0.64)$ . Since the variance of the random variable  $\mu$  was sufficiently small (i.e.,  $0.1 \leq 1$ ), the expectation of the random variable  $\mu$  essentially reflected the most possible value over time. Thus, it was reasonable to select  $\mathcal{N}(5.2, 0.64)$  as the normal distribution for the transportation cost in the MCNFP-rm to compare with that in the MCNFP-birm. The transformation of the other related uncertain parameters followed a similar pattern. Thus the model for the MCNFP-rm was formulated and solved using the proposed MOBLPSO 10 times.

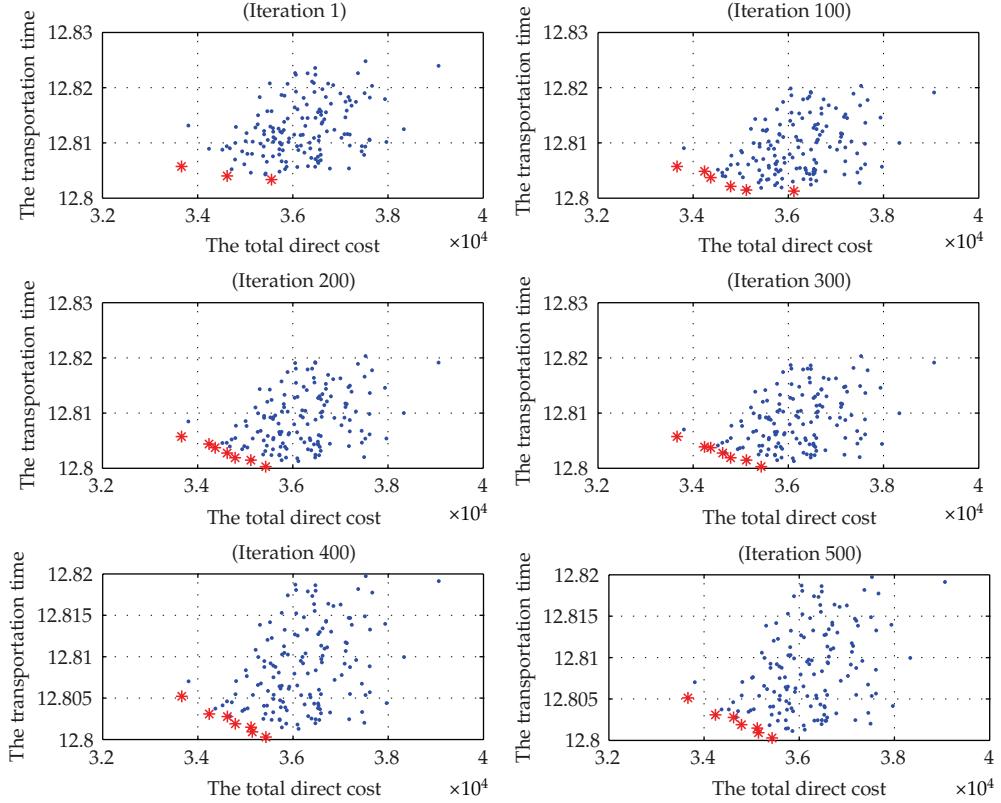
As shown in Table 6, the best, the worst, and the average results for the random type are higher than their counterparts for the MCNFP-birm. It is worth noting that the gaps

**Table 4:** The most cost-effective plan for MCNFP in Shuibuya case.

Transportation path $j$	1	2	3	4	5	6	7	8	9	10	11
Material volume $x_j$	950.000	543.763	1672.488	256.317	177.432	1700.000	766.646	1411.958	57.966	63.430	1279.675
Material volume $y_{1j}$	426.432	309.723	576.206	71.020	93.874	505.161	371.885	449.943	36.224	46.715	520.477
Material volume $y_{2j}$	413.345	107.514	587.051	92.521	44.360	620.743	267.585	521.378	42.172	48.536	364.225
Material volume $y_{3j}$	369.912	159.467	509.230	52.741	127.025	574.096	285.792	440.637	34.154	48.542	394.974
Transportation path $j$	12	13	14	15	16	17	18	19	20	21	22
Material volume $x_j$	370.325	161.357	1318.643	1380.000	1700.000	346.384	903.616	1150.000	1300.000	747.349	512.651
Material volume $y_{1j}$	144.947	43.373	487.119	398.457	650.263	158.874	427.931	381.462	382.515	363.827	63.158
Material volume $y_{2j}$	195.901	42.686	421.532	476.339	598.634	111.618	200.388	430.262	446.545	264.172	80.990
Material volume $y_{3j}$	155.662	46.672	409.992	505.204	451.103	171.539	325.558	338.276	470.940	373.284	484.426

**Table 5:** The most time-saving plan for MCNFP in Shuibuya case.

Transportation path $j$	1	2	3	4	5	6	7	8	9	10	11
Material volume $x_j$	950.000	1013.517	758.507	820.247	57.735	1700.000	948.762	1056.185	253.392	41.661	647.444
Material volume $y_{1j}$	265.783	354.500	308.790	329.115	45.111	573.575	492.092	422.106	157.707	21.322	363.277
Material volume $y_{2j}$	427.952	324.088	400.510	476.403	39.031	523.953	354.168	405.054	48.491	28.744	222.921
Material volume $y_{3j}$	293.735	334.929	390.005	432.165	38.571	602.472	398.627	315.831	56.696	39.224	199.717
Transportation path $j$	12	13	14	15	16	17	18	19	20	21	22
Material volume $x_j$	1002.556	200.435	1279.565	1380.000	1700.000	1057.049	192.951	1150.000	1300.000	500.306	759.694
Material volume $y_{1j}$	489.996	100.614	337.177	417.547	552.807	274.888	127.423	337.068	377.977	274.467	369.143
Material volume $y_{2j}$	402.297	109.078	535.589	454.315	539.657	493.531	131.399	451.603	525.488	255.822	299.626
Material volume $y_{3j}$	265.074	31.314	406.799	508.137	607.536	439.660	47.452	361.329	396.535	65.138	331.525



**Figure 9:** Pareto optimal solutions of upper level programming for Shuibuya Hydropower Project.

between the best and the worst and between the best and the average solutions for MCNFP-birm are wider than the gaps of their counterparts for the birandom type. This shows that randomness creates a much larger solution space when uncertainty is introduced. Fortunately, the widened solution space with a further stochastic nature in the MCNFP-birm provides better solutions and they were successfully located by MOBLPSO, evidenced by the narrower gaps between the best and the worst and between the best and the average solutions found for the MCNFP-birm. This suggests that MOBLPSO is an effective and relatively efficient approach for solving the MCNFP under a birandom environment.

### 6.5. Algorithm Evaluation

Since the PSOPC for the lower level is nested in the MOPSO for the upper level, and the MOPSO is the main body of the proposed MOBLPSO, the evaluation of the MOPSO was mainly paid attention to. In the MOPSO, a multiobjective method is introduced to derive the Pareto optimal solution set for the upper level programming. This provides effective and nondominated alternate schemes for the construction contractor. Compared to the weight-sum method dealing with multiobjectives in [21], the solutions here are confirmed to be more practical.

**Table 6:** Comparisons of two types of model.

	The direct cost objective ( $\times 10^3$ )		The transportation time cost objective ( $\times 10^3$ )		The transportation cost objective ( $\times 10^3$ )	
	Birandom type	Random type	Birandom type	Random type	Birandom type	Random type
Best result	33.662	33.832	12.8003	12.7997	18.036	18.067
Average result	34.546	35.022	12.8027	12.8045	18.239	18.285
Worst result	35.429	36.211	12.8051	12.8092	18.441	18.503

**Table 7:** Algorithm evaluation by metrics of performance for Pareto optimal sets.

Iteration	The average distance metric	The distribution metric	The extent metric
1	0.0781	0.1474	48.6227
100	0.0738	0.3460	58.1423
200	0.0878	0.4031	61.1722
300	0.0526	0.6967	69.2970
400	0.0554	0.8082	64.7773
500	0.0435	0.8938	68.8648

Comparing different optimization techniques experimentally always involves the notion of performance. In the case of multiobjective optimization, the definition of quality is substantially more complex than for single-objective optimization problems. There are many metrics of performance to measure the distance of the resulting nondominated set to the Paretooptimal front, the distribution of the solution found, and the extent of the obtained nondominated front [55].

To gain further insight into the performance of the multiobjective method in the proposed algorithm, the procedure was run different times and the results are summarized in Figure 9 and Table 7. As is shown in Figure 9, the amount and the distribution of Pareto optimal solutions in each iteration are satisfactory. For further expression of the efficiency of the convergence, three metrics of performance proposed by Zitzler et al. [55] were introduced: (1) the average distances of the resulting nondominated set to the Pareto-optimal front: the value of which decreases with an increase in the iterations, meaning that the program results come in toward the Pareto optimal front, which expresses the convergence of the algorithm; (2) the distribution in combination with the number of non-dominated solutions found: the higher the value, the better the distribution for an appropriate neighbourhood parameter; (3) the extent of the obtained nondominated fronts: it uses a maximum extent in each dimension to estimate the range to which the fronts spread out, which in this paper equals the distance of the two outer solutions.

The metrics of performance for the Pareto optimal set is shown in Table 7 to provide a satisfactory result for the efficiency of the convergence. Although there are some fluctuations in the three metrics in the 500 iterations, these do not affect the final result.

To asses the efficiency and effectiveness of the MOBLPSO for the proposed MCNFP, the MOBLPSO results for the MCNFP in the Shuibuya Hydropower Project are compared with two other state-of-the-art heuristic algorithms, that is, a genetic algorithm for a multi-objective bilevel model (denoted by MOBLGA) [56] and a simulated annealing algorithm for a multiobjective bilevel model (denoted by MOBLSA) [57].

In order to carry out the comparisons under a similar circumstance, the parameter selections for the MOBLGA and MOBLSA refer to those of the MOBLPSO, and nondominated alternate schemes are also employed for both. To measure the quality of the results obtained by the three algorithms, a weight sum method was introduced to detrmine one minimal weight sum for the objectives from the nondominated solutions. Thus the comparison could be implemented based on the unique measured criterion (i.e., the minimal weight sum of the objectives). To ensure the conformity validity of the multiobjectives, the division of the dimensions and a unifying of the order of magnitude need to be performed before the weight-sum procedure.

Table 8 shows the comparison results, that is, the minimal weight sum value of the two objectives, and the average computation times, obtained using the preceding approaches

**Table 8:** Computation time and memory used by MOBLPSO, MOBLGA and MOBLSA.

Type	Combination of weights		Minimal weight sum value of the two objectives		Average computation time (min)	
	$\omega_1$	$\omega_2$	MOBLPSO	MOBLGA	MOBLPSO	MOBLGA
1	$\omega_1 = 0.1$	$\omega_2 = 0.9$	1152.027	1160.989	1152.531	20.856
2	$\omega_1 = 0.2$	$\omega_2 = 0.8$	1024.272	1032.419	1024.273	20.926
3	$\omega_1 = 0.3$	$\omega_2 = 0.7$	896.238	903.848	897.214	21.052
4	$\omega_1 = 0.4$	$\omega_2 = 0.6$	769.472	775.277	769.555	21.158
5	$\omega_1 = 0.5$	$\omega_2 = 0.5$	641.786	646.706	641.897	20.963
6	$\omega_1 = 0.6$	$\omega_2 = 0.4$	514.148	518.136	514.216	20.678
7	$\omega_1 = 0.7$	$\omega_2 = 0.3$	386.486	388.529	386.521	21.203
8	$\omega_1 = 0.8$	$\omega_2 = 0.2$	258.792	260.836	258.792	20.734
9	$\omega_1 = 0.9$	$\omega_2 = 0.1$	131.081	132.143	131.081	21.134
						26.189
						34.815

for the different combination of weights (i.e.,  $\omega_1$  and  $\omega_2$  represent the weights of the two objectives, resp.). It is demonstrated that the MOBLPSO for the MCNFP can perform optimizing better than the MOBLGA, since MOBLGA may lead to a local search and need more computation time. On the other hand, the MOBLSA could get similar results to those from MOBLPSO, but computation was much slower than the MOBLPSO.

## 7. Conclusions and Future Research

In this paper, a multiobjective bilevel programming model under birandom environment for an MCNFP in a large-scale construction project was formulated. The contributions of this paper to the literature are as follows. Firstly, the multiobjective bilevel model for the minimum cost network flow problem in a large-scale construction project focused on here was found to provide a more reasonable expression of the proposed problem, where the upper level aims at optimizing the material flow assignment along the transportation paths and the lower level decides on the flow of each carrier transports on the paths. Secondly, because of the complicated realistic decision systems, this study employs birandom variables to characterize the hybrid uncertain environment. The application of birandom variables makes the proposed programming model more suitable for describing a vague and uncertain situation in the real world. Further, the birandom uncertainty model was converted into an expectation multiobjective bilevel programming model with chance constraints. Thirdly, in order to solve the NP-hard multiobjective bilevel problem, a very effective and relatively efficient algorithm (i.e., MOBLPSO) was developed by employing both a MOPSO and a PSOPC. Finally, the Shuibuya Hydropower Project was used here as a practical application example. The MOBLPSO results for the preceding project example were compared with MOBLGA and MOBLSA methods, which demonstrated the validity of the proposed mathematical model and the effectiveness of the proposed MOBLPSO method in handling complex problems.

Further research is necessary to identify further properties to develop a more effective method for solving other practical problems: (1) the formulation of an MCNFP for manifold materials rather than only one type of material transportation network in large-scale construction projects, (2) the investigation of other new approaches such as an automated design methodology and dependent chance programming to handle the birandom variables more reasonably and effectively, (3) the development of more efficient solution methods to solve multiobjective bilevel programming problems. Each of these areas is very important and equally worthy of attention. It should be mentioned that there are several commercial solvers that can efficiently solve large-scale nonlinear problems such as MINOS, CONOPT and SNOPT. However, when solving bilevel programming with nonlinear and non-differentiable piecewise objective functions and constraints like the MCNFP discussed in this paper, these solvers may face difficulties to deal with the nondifferentiability and nonconvexity by employing the exact techniques such as enumeration method, Karush-Kuhn-Tucker method, and penalty function approach. The future research may seek to address this issue with alternative exact techniques.

## Acknowledgments

This paper was supported by the Key Program of NSFC (Grant no. 70831005) and “985” Program of Sichuan University “Innovative Research Base for Economic Development and Management.” The authors would like to give their great appreciates to the editors and

anonymous referees for their helpful and constructive comments and suggestions, which have helped to improve this paper.

## References

- [1] R. K. Ahuja, T. L. Magnanti, J. B. Orlin, and M. R. Reddy, "Applications of network optimization," in *Network Models*, vol. 7, pp. 1–83, North-Holland, Amsterdam, The Netherlands, 1995.
- [2] F. Glover, D. Klingman, and N. Phillips, *Network Models in Optimization and Their Applications in Practice*, Wiley, New York, NY, USA, 1st edition, 1992.
- [3] E. L. Hahne and R. G. Gallager, "Round robin scheduling for fair flow control in data communication networks," *NASA STI/Recon Technical Report N*, vol. 86, pp. 30–47, 1986.
- [4] M. Pióro and D. Medhi, *Routing, Flow, and Capacity Design in Communication and Computer Networks*, Morgan Kaufmann, 2004.
- [5] A. S. Avestimehr, S. N. Diggavi, and D. N. C. Tse, "Wireless network information flow: a deterministic approach," *IEEE Transactions on Information Theory*, vol. 57, no. 4, pp. 1872–1905, 2011.
- [6] W. S. Lee, W. I. Lim, and P. H. Koo, "Transporter scheduling based on a network flow model under a dynamic block transportation environment," in *Proceedings of the International Conference on Computers & Industrial Engineering*, pp. 311–316, 2009.
- [7] K. Paparrizos, N. Samaras, and A. Sifaleras, "An exterior simplex type algorithm for the minimum cost network flow problem," *Computers & Operations Research*, vol. 36, no. 4, pp. 1176–1190, 2009.
- [8] C. Dang, Y. Sun, Y. Wang, and Y. Yang, "A deterministic annealing algorithm for the minimum concave cost network flow problem," *Neural Networks*, vol. 24, no. 7, pp. 699–708, 2011.
- [9] D. B. M. M. Fontes, E. Hadjiconstantinou, and N. Christofides, "A dynamic programming approach for solving single-source uncapacitated concave minimum cost network flow problems," *European Journal of Operational Research*, vol. 174, no. 2, pp. 1205–1219, 2006.
- [10] D. Goldfarb and Z. Jin, "A new scaling algorithm for the minimum cost network flow problem," *Operations Research Letters*, vol. 25, no. 5, pp. 205–211, 1999.
- [11] A. V. Goldberg, "An efficient implementation of a scaling minimum-cost flow algorithm," *Journal of Algorithms*, vol. 22, no. 1, pp. 1–29, 1997.
- [12] A. Sedeño-Noda and C. González-Martín, "An algorithm for the biobjective integer minimum cost flow problem," *Computers & Operations Research*, vol. 28, no. 2, pp. 139–156, 2001.
- [13] X. Zhu, Q. Yuan, A. García-Díaz, and L. Dong, "Minimal-cost network flow problems with variable lower bounds on arc flows," *Computers & Operations Research*, vol. 38, no. 8, pp. 1210–1218, 2011.
- [14] D. Watling, "User equilibrium traffic network assignment with stochastic travel times and late arrival penalty," *European Journal of Operational Research*, vol. 175, no. 3, pp. 1539–1556, 2006.
- [15] A. Chen and Z. Zhou, "The  $\alpha$ -reliable mean-excess traffic equilibrium model with stochastic travel times," *Transportation Research B*, vol. 44, no. 4, pp. 493–513, 2010.
- [16] Y. K. Lin, "Reliability evaluation of a revised stochastic flow network with uncertain minimum time," *Physica A*, vol. 389, no. 6, pp. 1253–1258, 2010.
- [17] A. Sumalee, D. P. Watling, and S. Nakayama, "Reliable network design problem: case with uncertain demand and total travel time reliability," *Transportation Research Record*, vol. 1964, no. 1, pp. 81–90, 2006.
- [18] A. M. Geoffrion and W. W. Hogan, "Coordination of two-level organizations with multiple objectives," in *Techniques of Optimization*, pp. 455–466, Academic Press, New York, NY, USA, 1972.
- [19] K. Tarvainen and Y. Y. Haimes, "Coordination of hierarchical multiobjective systems: theory and methodology," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 12, no. 6, pp. 751–764, 1982.
- [20] M. S. Osman, M. A. Abo-Sinna, A. H. Amer, and O. E. Emam, "A multi-level non-linear multi-objective decision-making under fuzziness," *Applied Mathematics and Computation*, vol. 153, no. 1, pp. 239–252, 2004.
- [21] G. Zhang, J. Lu, and T. Dillon, "Decentralized multi-objective bilevel decision making with fuzzy demands," *Knowledge-Based Systems*, vol. 20, no. 5, pp. 495–507, 2007.
- [22] H. I. Calvete and C. Galé, "Linear bilevel programs with multiple objectives at the upper level," *Journal of Computational and Applied Mathematics*, vol. 234, no. 4, pp. 950–959, 2010.
- [23] B. Colson, P. Marcotte, and G. Savard, "Bilevel programming: a survey," *4OR: A Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, vol. 3, no. 2, pp. 87–107, 2005.

- [24] J. Xu and X. Zhou, "A class of multi-objective expected value decision-making model with birandom coefficients and its application to flow shop scheduling problem," *Information Sciences*, vol. 179, no. 17, pp. 2997–3017, 2009.
- [25] L. Yan, "One type of optimal portfolio selection in birandom environments," *Modern Applied Science*, vol. 3, no. 6, pp. 126–131, 2009.
- [26] J. Xu and C. Ding, "A class of chance constrained multiobjective linear programming with birandom coefficients and its application to vendors selection," *International Journal of Production Economics*, vol. 131, no. 2, pp. 709–720, 2011.
- [27] S. Gao and I. Chabini, "Optimal routing policy problems in stochastic time-dependent networks," *Transportation Research B*, vol. 40, no. 2, pp. 93–122, 2006.
- [28] N. E. El Faouzi and M. Maurin, "Reliability Metrics for path travel time under log-normal distribution," in *Proceedings of the 3rd International Symposium on Transportation Network Reliability*, Delft, The Netherlands, July 2007.
- [29] H. Rakha, I. El-Shawarby, and M. Arafah, "Trip travel-time reliability: issues and proposed solutions," *Journal of Intelligent Transportation Systems*, vol. 14, no. 4, pp. 232–250, 2010.
- [30] E. Mazloumi, G. Currie, and G. Rose, "Using GPS data to gain insight into public transport travel time variability," *Journal of Transportation Engineering*, vol. 136, no. 7, Article ID 006007QTE, pp. 623–631, 2010.
- [31] L. R. Ford Jr. and D. R. Fulkerson, *Flows in Networks*, Princeton University Press, Princeton, NJ, USA, 1962.
- [32] J. Peng and B. Liu, "Birandom variables and birandom programming," *Computers and Industrial Engineering*, vol. 53, no. 3, pp. 433–453, 2007.
- [33] Y. Sheffi, *Urban Transportation Network*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1984.
- [34] A. Charnes and W. W. Cooper, "Deterministic equivalents for optimizing and satisficing under chance constraints," *Operations Research*, vol. 11, pp. 18–39, 1963.
- [35] K. Krickeberg, *Probability Theory*, Addison-Wesley, Reading, UK, 1965.
- [36] J. Xu and L. Yao, *Random-Like Multiple Objective Decision Making*, Springer, 2009.
- [37] P. Hansen, B. Jaumard, and G. Savard, "New branch-and-bound rules for linear bilevel programming," *Society for Industrial and Applied Mathematics*, vol. 13, no. 5, pp. 1194–1217, 1992.
- [38] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE Conference on Neural Networks*, Piscataway, NJ, USA, 1995.
- [39] J. Kennedy and R. Eberhart, *Swarm Intelligence*, Morgan Kaufmann Publishers, 2001.
- [40] M. P. Song and G. C. Gu, "Research on particle swarm optimization: a review," in *Proceedings of International Conference on Machine Learning and Cybernetics*, pp. 2236–2241, chn, August 2004.
- [41] D. Y. Sha and C. Y. Hsu, "A hybrid particle swarm optimization for job shop scheduling problem," *Computers & Industrial Engineering*, vol. 51, no. 4, pp. 791–808, 2006.
- [42] S. H. Ling, H. H. C. Iu, K. Y. Chan, H. K. Lam, B. C. W. Yeung, and F. H. Leung, "Hybrid particle swarm optimization with wavelet mutation and its industrial applications," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 38, no. 3, pp. 743–763, 2008.
- [43] J. Xu, F. Yan, and S. Li, "Vehicle routing optimization with soft time windows in a fuzzy random environment," *Transportation Research E*, vol. 47, no. 6, pp. 1075–1091, 2011.
- [44] R.-M. Chen and C.-M. Wang, "Project scheduling heuristics-based standard PSO for task-resource assignment in heterogeneous grid," *Abstract and Applied Analysis*, vol. 2011, Article ID 589862, 20 pages, 2011.
- [45] J. Xu and Z. Zeng, "A discrete time optimal control model with uncertainty for dynamic machine allocation problem and its application to manufacturing and construction industries," *Applied Mathematical Modelling*, vol. 36, no. 8, pp. 3513–3544, 2012.
- [46] C. C. Coello and M. S. Lechuga, "MOPSO: a proposal for multiple objective particle swarm optimization," in *Proceedings of the Congress on Evolutionary Computation*, 2002.
- [47] C. A. Coello Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 256–279, 2004.
- [48] J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the Pareto Archived Evolution Strategy," *Evolutionary Computation*, vol. 8, no. 2, pp. 149–172, 2000.
- [49] F. van den Bergh and A. P. Engelbrecht, "A convergence proof for the particle swarm optimiser," *Fundamenta Informaticae*, vol. 105, no. 4, pp. 341–374, 2010.
- [50] Z. Zhigang and G. Xinyi, "Particle swarm optimization based algorithm for bilevel programming problems," in *Proceedings of the 6th ISDA International Conference on Intelligent Systems Design and Applications*, pp. 951–956, October 2006.

- [51] K. Deb and D. E. Goldberg, "An investigation of niche and species formation in genetic function optimization," in *Proceedings of the 3rd International Conference of Genetic Algorithms*, pp. 42–50, 1989.
- [52] S. He, Q. H. Wu, J. Y. Wen, J. R. Saunders, and R. C. Paton, "A particle swarm optimizer with passive congregation," *BioSystems*, vol. 78, no. 1–3, pp. 135–147, 2004.
- [53] Y. Shi and R. C. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, 1998.
- [54] R. C. Eberhart and Y. Shi, "Comparing inertia weights and constriction factors in particle swarm optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '00)*, pp. 84–88, San Diego, Calif, USA, July 2000.
- [55] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: empirical results," *Evolutionary Computation*, vol. 8, no. 2, pp. 173–195, 2000.
- [56] Y. Yin, "Multiobjective bilevel optimization for transportation planning and management problems," *Journal of Advanced Transportation*, vol. 36, no. 1, pp. 93–105, 2002.
- [57] H. Liu, L. Wang, and H. Song, "Bi-level model of railway transportation existing network and simulated annealing algorithm," in *Proceedings of the 2nd International Conference on Transportation Engineering (ICTE '09)*, vol. 4, pp. 3314–3319, July 2009.

*Research Article*

# A Selection Approach for Optimized Problem-Solving Process by Grey Relational Utility Model and Multicriteria Decision Analysis

**Chih-Kun Ke<sup>1</sup> and Mei-Yu Wu<sup>2</sup>**

<sup>1</sup> Department of Information Management, National Taichung University of Science and Technology, No. 129, Section 3, Sanmin Road, Taichung 404, Taiwan

<sup>2</sup> Department of Information Management, Chung Hua University, No. 707, Section 2, WuFu Road, Hsinchu 300, Taiwan

Correspondence should be addressed to Mei-Yu Wu, mywu@chu.edu.tw

Received 21 January 2012; Accepted 17 April 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 C.-K. Ke and M.-Y. 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.

In business enterprises, especially the manufacturing industry, various problem situations may occur during the production process. A situation denotes an evaluation point to determine the status of a production process. A problem may occur if there is a discrepancy between the actual situation and the desired one. Thus, a problem-solving process is often initiated to achieve the desired situation. In the process, how to determine an action need to be taken to resolve the situation becomes an important issue. Therefore, this work uses a selection approach for optimized problem-solving process to assist workers in taking a reasonable action. A grey relational utility model and a multicriteria decision analysis are used to determine the optimal selection order of candidate actions. The selection order is presented to the worker as an adaptive recommended solution. The worker chooses a reasonable problem-solving action based on the selection order. This work uses a high-tech company's knowledge base log as the analysis data. Experimental results demonstrate that the proposed selection approach is effective.

## 1. Introduction

Problem solving is an important process that enables corporations to create competitive advantages. In manufacturing industries, various problem situations may occur during the production process [1, 2]. A situation denotes an evaluation point to determine the status of a production process. A problem may occur if there is a discrepancy between the actual situation and the desired one [3]. Thus, a problem-solving process is often initiated to achieve the desired situation. In the process, workers determine what action needs to be taken to resolve the situation. For a given problem, a situation may occur with various features

according to the context at that time. Due to the uncertain characteristics of situations, several causes and possible actions may exist for a specific situation. Workers may observe a problem situation, collect relevant information from the enterprise knowledge repository, explore possible causes, and identify operational conditions in order to decide appropriate action [3, 4].

Quality of Service (QoS) is an important consideration in evaluating a problem-solving solution. Worker feedback of an evaluating process can be represented as a utility model reflecting the satisfaction a worker observes from taking an action. The worker provides such a utility model [5] before committing to take an action. Grey relational analysis [6, 7] can quantify all influences of various factors and their relation to consolidate the utility model. Therefore, the worker's grey relational utility model can be applied to the monitoring information in order to evaluate the action's QoS. The worker will get the expected value of the issue of interest from taking an action. Based on various issues of interest, how to select the reasonable action from a large number of candidate actions requires a multicriteria decision analysis. A multicriteria decision analysis [8] is concerned with structuring and solving decision and planning problems involving multiple criteria. The purpose is to support decision makers facing such problems. Typically, there does not exist a unique optimal action for such problems, and it is necessary to use decision maker's preferences to differentiate between actions. Therefore, a multicriteria decision analysis is required to discover the selection order of the various actions for a specific situation. The discovered selection order helps worker to solve the situation.

This work uses a selection approach to candidate actions to assist the worker in acquiring a reasonable problem-solving action. Action formalization, a grey relational utility model, and a multicriteria decision analysis are used to obtain an optimal selection order for candidate actions. Then, the selected action for a specific situation is taken through a problem-solving process. The result is considered a reasonable problem-solving solution for the worker. This work explores a high-tech company's knowledge base log as the analysis data. The prototype system and use cases are proposed in the previous research [4]. In this work, we have an experiment to demonstrate that the proposed approach is effective. The contribution of this research is in demonstrating a method which is easy to implement in a problem solving knowledge recommendation system for selecting a reasonable solution.

The remainder of this paper is organized as follows. Section 2 reviews related works on problem solving, grey relational analysis, multicriteria decision analysis, knowledge management, and retrieval. Section 3 introduces a selection approach for optimized problem-solving process by a grey relational utility model and a multicriteria decision analysis. Section 4 describes an experimental data of a high-tech company's knowledge base. An experiment of the knowledge base log and discussions are showed in Section 5. Finally, Section 6 presents our conclusions.

## 2. Related Works

The related literature covers problem solving, grey relational analysis and utility model, multicriteria decision analysis, knowledge management, and retrieval.

### 2.1. Problem-Solving Process

In business enterprises, especially the manufacturing industry, various problem situations may occur during the production process [1, 2], for example, poor production performance,

system overload, and low machine utilization. A situation denotes an evaluation point to determine the status (i.e., desirable or undesirable) of a production process. A problem may occur if there is a discrepancy between the actual situation and the desired one [3]. For example, when the current production output is below the desired level, the production line may have some problems. Thus, a problem solving process is often initiated to achieve the desired situation. Problem solving is the thought process that resolves various difficulties and obstacles spread in the gap between the current problem and its desired solution [4].

Various approaches have been proposed to support problem solving. Allen et al. enforced a problem-solving process based on the knowledge gained from solving previous similar problems [9]. Chang et al. implemented a self-improvement helpdesk service system [1], and Park et al. developed a decision support system for problem solving in a complex production process [2]. More recently, Yang et al. proposed integrating the case-based reasoning (CBR) approach with ART-Kohonen neural networks (ART-KNNs) to enhance fault diagnosis in electric motors [10]. Moreover, Guardati introduced RBCShell as a tool for constructing knowledge-based systems, whereby previously solved problems are stored in the case memory to support problem solving in new cases [11].

In a complex production process, problem solving is usually knowledge intensive. Past experience or knowledge, routine problem-solving procedures, and previous decisions can be used to enhance problem solving. The types of knowledge are investigated to use for problem solving and suggest the circulation of knowledge to avoid knowledge inertia [12]. In the problem-solving process, workers take several problem-solving steps to determine what action needs to be taken to resolve the situation [3]. Such action involves both human wisdom and enterprise knowledge. Workers may observe a problem situation, collect relevant information from the enterprise knowledge repository, explore possible causes, and identify operational conditions in order to decide appropriate action [4].

## **2.2. Grey Relational Analysis and Utility Model**

The grey relational analysis (GRA) is an important method in the grey system theory [7]. The GRA has been widely used in a number of areas, such as manufacturing [13], transportation [14], and the building trade [6]. In the grey system theory, the GRA is essentially believed to have captured the similarity measurements or relations in a system. Generally, the procedure of grey relation analysis includes grey relation generation and grey relational grade calculation steps. The grey relation generation step removes anomalies associated with different measurement units and scales by the normalization of raw data. The grey relational grade calculation step uses the grey relational coefficient to describe the trend relationship between an objective series and a reference series at a given point in a system [15].

Quality of Service (QoS) is an important consideration in evaluating a problem-solving action. Worker feedback of an evaluating process can be represented as a utility model reflecting the satisfaction a worker observes from taking an action. The worker provides such a utility model [5] before committing to take an action. GRA can quantify all influences of various factors and their relation to consolidate the utility model. Therefore, the worker's grey relational utility model can be applied to the monitoring information in order to evaluate the action's QoS. The worker will get the expected value of the issue of interest from taking an action.

### **2.3. Multicriteria Decision Analysis Method, ELECTRE**

Multicriteria decision-making (MCDM) approach has played an important role in solving multidimensional and complicated problems. ELECTRE (Elimination Et Choix Translating Reality) is a family of multicriteria decision analysis methods [8, 16]. ELECTRE methods are developed in two main phases. In the first phase, the outranking relations are constructed for a comprehensive comparison of each pair of actions. In the second phase, the recommendations are elaborated from the results obtained by an exploitation procedure from the first phase. The nature of the recommendation depends on the following problems: choosing, ranking, or sorting. The evolutions of ELECTRE methods include ELECTRE I, ELECTRE Iv, ELECTRE IS, ELECTRE II, ELECTRE III, ELECTRE IV, ELECTRE-SS, and ELECTRE TRI. Each method is characterized by its construction and exploitation procedure. ELECTRE I, ELECTRE Iv, and ELECTRE IS were designed to solve choice problem. ELECTRE II, ELECTRE III, ELECTRE IV, and ELECTRE-SS were designed for solving ranking problems. ELECTRE TRI was designed for solving sorting problems. This work uses a modified version of the ELECTRE method [17] to discover an optimal selection order of candidate actions. The selection order is presented to the worker as a recommended solution.

### **2.4. Knowledge Management and Retrieval**

A repository of structured, explicit knowledge, especially in document form, is a codified strategy for managing knowledge [18, 19]. However, with the growing amount of information in organization memories, knowledge management systems (KMSs) face the challenge of helping users find pertinent information. Accordingly, knowledge retrieval is considered a core component in accessing information in knowledge repositories [20, 21]. Translating users' information needs into queries is not easy. Most systems use information retrieval (IR) techniques [22] to access organizational codified knowledge [23]. The use of information filtering (IF) with a profiling method to model users' information needs is an effective approach that proactively delivers relevant information to users. The technique has been widely used in the areas of information retrieval and recommender systems [24–26]. The profiling approach has also been adopted by some KMSs to enhance knowledge retrieval [27–29], whereby information is delivered to task-based business environments to support proactive delivery of task-relevant knowledge [20, 27, 30].

This work explores a high-tech company's knowledge base log [4] as the analysis data. Knowledge management and retrieval techniques are used to enforce an experiment to demonstrate that the proposed selection approach is effective.

## **3. The Selection Approach for Optimized Problem-Solving Process**

This section describes a selection approach to candidate actions in terms of grey relational utility model and multicriteria decision analysis, including problem-solving action formalization, grey relational utility model for candidate actions, and selection order discovery by a modified version of the ELECTRE method [17].

### **3.1. Problem-Solving Action Formalization**

Problem-solving action formalization is an essential and initial task in our proposed selection approach. This work refers to the use of a utility-based reputation model [5] to formalize an action's QoS items in order to enforce the utility model.

Let  $A = \{a_1, a_2, \dots, a_n\}$  denote the set of actions, and let  $a \in A$ . Let  $AP$  denote the set of actions providers, let  $b \in AP$ , and let function  $S : AP \rightarrow P(A)$  denote the actions provided by an action provider, where  $P$  represents the power set operator. Let  $SW$  denote the set of worker of the system, and let  $w \in SW$ . Each action has associated issues of interest, denoted by set  $I$ , whose workers are interested in monitoring, and  $i \in I$ . Function  $IS$  represents the set of issues of interest for an action:  $IS : A \rightarrow P(I)$ . Function  $O^w : A \times AP \times I \rightarrow R$  denotes the expectation of the worker  $w$  for the actions he takes, where  $R$  denotes the real numbers. Notation  $v_{a,i}^{w,b}$  represents the expectation of worker  $w$  on issue  $i$  of action  $a$  supplied by provider  $b$ .

In a problem-solving environment, a potential issue of interest could be the QoS. Based on the expectations, a worker can develop a utility model which reflects the satisfaction he perceives from taking an action.

### 3.2. Grey Relational Utility Model for Candidate Actions

After the expectation formalization process of a problem-solving action's specific interest issue, a grey relational utility model is developed to represent worker satisfaction with action acquisition.

Let  $U_{a,i}^{w,b}(v)$  denote the utility that worker  $w$  gets by obtaining the actual value  $v \in R$  on issue  $i$  from action  $a$  of provider  $b$ . Each expected value  $v$  of specific interest issue  $i$  of an action used as a QoS item to build a comparative vector  $a_i = (U_{a,i_1}^{w,b}(v_1), U_{a,i_2}^{w,b}(v_2), \dots, U_{a,i_n}^{w,b}(v_n))$ ,  $i = \{1, 2, \dots, m\}$ . This work sets the  $a_0$  as a desired action with expected utility values of specific interest issues. It forms a reference vector  $a_0 = (U_{a_0,i_1}^{w,b}(v_1), U_{a_0,i_2}^{w,b}(v_2), \dots, U_{a_0,i_n}^{w,b}(v_n))$ . Utilities are normalized and scaled to  $[0, 1]$  by grey relation consideration [31, 32], as shown in (3.1) and (3.2). Larger-the-better means that the larger target value is better, and smaller-the-better means that the smaller target value is better. Therefore,  $U_{a,i}^{w,b} R \rightarrow [0, 1]$ .

Larger-the-better is as follows:

$$U_{a,i}^{*w,b}(v) = \frac{U_{a,i}^{w,b}(v) - \min_i U_{a,i}^{w,b}(v)}{\max_i U_{a,i}^{w,b}(v) - \min_i U_{a,i}^{w,b}(v)}. \quad (3.1)$$

Smaller-the-better is as follows:

$$U_{a,i}^{*w,b}(v) = \frac{\max_i U_{a,i}^{w,b}(v) - U_{a,i}^{w,b}(v)}{\max_i U_{a,i}^{w,b}(v) - \min_i U_{a,i}^{w,b}(v)}. \quad (3.2)$$

Then the grey relation equation [25, 32, 33] is used to calculate the grey relational grade between reference vector and comparative vectors, partial equation as shown in (3.3). The  $U_{a_0,i}^{w,b}(v)$  is a partial utility of reference vector, and  $U_{a_k,i}^{w,b}(v)$  is a partial utility of comparative vector. If the grey relational grade value  $\Gamma_{0k}$  is closer to 1, it means that  $U_{a_0,i}^{w,b}(v)$  and  $U_{a_k,i}^{w,b}(v)$

have high correlation. If the grey relational grade value  $\Gamma_{0k}$  is closer to 0, it means that  $U_{a_0,i}^{w,b}(v)$  and  $U_{a_k,i}^{w,b}(v)$  have low correlation:

$$\Gamma_{0k} = \Gamma(U_{a_0,i}^{w,b}(v), U_{a_k,i}^{w,b}(v)) = \frac{\bar{\Delta}_{\max} - \bar{\Delta}_{0k}}{\bar{\Delta}_{\max} - \bar{\Delta}_{\min}}, \quad \text{where } \bar{\Delta}_{0k} = \left( \sum_{k=1}^n [\Delta_{0k}(v)]^\rho \right)^{1/\rho}, \quad (3.3)$$

where  $\rho = \{1, 2, \dots, m\}$ .  $\bar{\Delta}_{\max}$  is the largest value of  $\bar{\Delta}_{0k}$  and  $\bar{\Delta}_{\min}$  is the smallest value of  $\bar{\Delta}_{0k}$ . Based on the grey relational grade value  $\Gamma$ , a threshold value is used to filter out the low correlation actions, and the remainders are considered as candidate actions for solving a specific problem situation.

In Sections 3.1 and 3.2, the worker will get the expected value of the issue of interest from taking an action. Based on various issues of interest, how to select the reasonable action from a large number of candidate actions requires a multicriteria decision analysis.

### 3.3. Discover a Selection Order of Actions by a Modified ELECTRE Method

For the second task, this work uses a modified version of the ELECTRE method to discover the selection order for candidate actions. If there are  $m$  candidate actions which involve  $n$  QoS items, the matrix  $Q$  of expected values can be shown as (3.4). We use 8 steps to discover the optimal selection order of action using a modified version of the ELECTRE method [17]:

$$Q = [Q_{ij}]_{m \times n} = \begin{bmatrix} v_{1,1}^{c,b} & \cdots & v_{1,n}^{c,b} \\ \vdots & \ddots & \vdots \\ v_{m,1}^{c,b} & \cdots & v_{m,n}^{c,b} \end{bmatrix}. \quad (3.4)$$

*Step 1.* To calculate the weighted normalization decision matrix, a weight for each QoS item must be set to form a weighted matrix ( $W$ ), as shown in:

$$W = \begin{bmatrix} W_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & W_n \end{bmatrix}_{n \times n}. \quad (3.5)$$

The multiplication of a normalization matrix  $Q$  by a weighted matrix  $W$  gets the weighted normalization decision matrix ( $V = QW$ ), as shown in:

$$V = [v_{ij}]_{m \times n}. \quad (3.6)$$

*Step 2.* Compare arbitrary different row  $i$  and row  $j$  in the weighted normalization decision matrix  $V$  to make sure of the concordance and discordance set. If value  $v$  of row  $i$  is higher

than value  $v$  of row  $j$ , the component  $k$  can be classified as the concordance set  $C_{ij}$  or the discordance set  $D_{ij}$ . The concordance set  $C_{ij}$  or the discordance set  $D_{ij}$  is shown as:

$$C_{ij} = \{k \mid v_{ik} \geq v_{jk}\}, \quad D_{ij} = \{k \mid v_{ik} < v_{jk}\}. \quad (3.7)$$

*Step 3.* The sum of each component's weight forms a concordance matrix  $C$ , as shown in:

$$C = [c_{ij}]_{m \times m}, \quad c_{ij} = \frac{\sum_{k \in C_{ij}} w_k}{\sum_{k \in S} w_k}. \quad (3.8)$$

*Step 4.* We use a formula to get the discordance matrix.  $S$  is the set including all QoS items,  $S = \{1, 2, \dots, n\}$ , as shown in:

$$d_{ij} = \frac{\max_{k \in D_{ij}} \{|v_{ik} - v_{jk}| \}}{\max_{k \in S} \{|v_{ik} - v_{jk}| \}}. \quad (3.9)$$

Therefore, a discordance matrix can be presented as  $D = [d_{ij}]_{m \times m}$ .

*Step 5.* The reverse complementary value is used to modify  $D$  to get the modified discordance matrix  $D'$ . The calculation of  $D'$  is shown as:

$$D' = [d'_{ij}]_{m \times m}, \quad d'_{ij} = 1 - d_{ij}. \quad (3.10)$$

*Step 6.* To show the large component value of the candidate solution, when the expected value is larger, we combine each component  $c_{ij}$  of the concordance set with the discordance matrix to calculate the production and get the modified total matrix  $A$  (*Hadamard product of  $c_{ij}$  and  $d'_{ij}$* ), as shown in:

$$A = [a_{ij}]_{m \times m}, \quad a_{ij} = c_{ij} \circ d'_{ij}. \quad (3.11)$$

*Step 7.* Get the maximum value  $a_j$  of each column from modified total matrix. The purpose is to determine the modified superiority matrix, as shown in:

$$a_j = \max\{a_{ij} \mid i = 1, 2, \dots, m\}, \quad j = 1, 2, \dots, m. \quad (3.12)$$

To make sure to get a reasonable solution, we have to rank  $a_j$  from small to large:  $a_1, a_2, \dots, a_m$ . The threshold  $\bar{a}$  is set behind the smallest value  $a'_1$  and the next smallest value  $a'_2$ . If the value  $a_{ij}$  is smaller than threshold  $\bar{a}$ , it is replaced as 0 or 1. Then we get the modified total superiority matrix, as shown in:

$$E' = [e'_{ij}], \quad e'_{ij} = \begin{cases} 1, & a_{ij} \geq \bar{a}, \\ 0, & a_{ij} < \bar{a}. \end{cases} \quad (3.13)$$

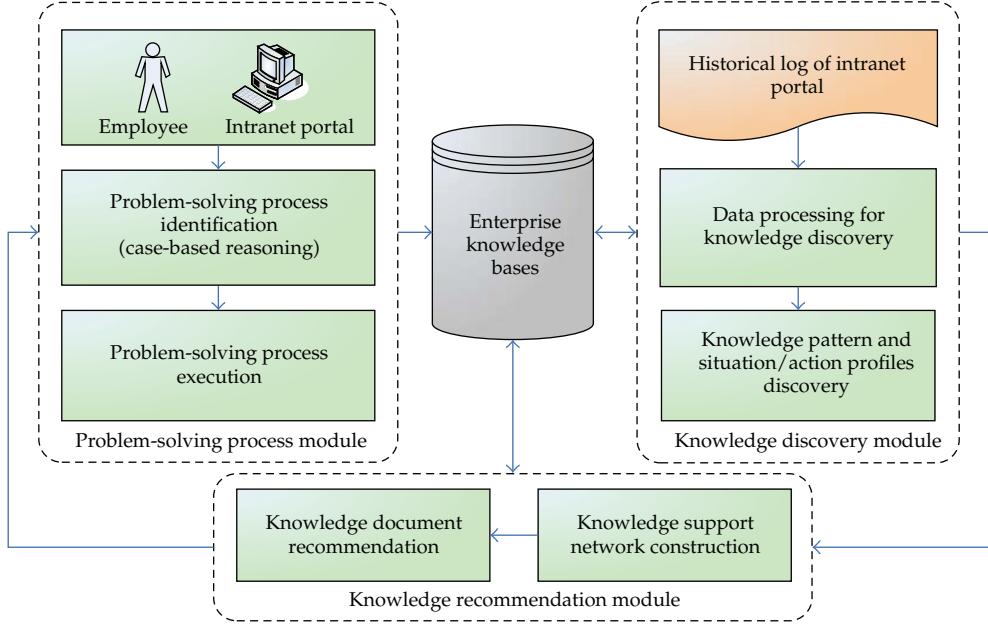


Figure 1: Knowledge support framework for problem solving [4].

*Step 8.* Finally, we get  $e'_{ij} = 1$  from the matrix  $E'$ . It indicates that solution  $i$  is better than solution  $j$ . We can eliminate solution  $j$  and show it as  $A_i \rightarrow A_j$ .

From Steps 1 to 8, we get the relationship among the QoS items of the candidate actions and get the optimal selection order for all candidate actions. The candidate action is the action provided by an action provider. The worker can follow the selection order to take a reasonable action.

#### 4. Experimental Knowledge Base Log

In this section, we fetch a high-tech company's knowledge base log as the analysis data. In the previous research [4], the proposed knowledge support framework for problem-solving is as shown in Figure 1. The proposed framework records the problem-solving steps, including the situations and actions as well as the corresponding knowledge documents accessed in the historical log. The knowledge discovery module employs mining technology to extract hidden knowledge from the historical problem solving log. The extracted knowledge, including situation/action profiles, decision making, and dependency knowledge, is used to provide knowledge support. The knowledge base comprises historical logs, discovered knowledge patterns, situation/action profiles, and enterprise knowledge documents. This component acts as an information hub to provide knowledge support for problem solving.

For specific situations or actions, relevant information (documents) accessed by workers is recorded in the problem-solving log. Historical codified knowledge (textual documents) can also provide valuable knowledge for solving the target problem. Information retrieval (IR) and text mining techniques are used to extract the key terms of relevant documents for a specific situation or action. The extracted key terms form the situation/action profile, which is used to model the information needs of the workers. This work assumes that

a generic problem-solving process is specified by experts to solve a problem or a set of similar problems encountered on a production line. When the production line encounters a problem, a problem-solving process is initiated. The situations that occurred in a problem may vary due to the uncertainty of the constantly changing business environment. Moreover, different workers may take different actions to solve a problem according to their skills and experience. The problem-solving log records historical problem solving instances.

## 5. The Experiments and Results Discussion

This section presents the experiment, results, and relevant discussion of the wafer manufacturing problem use case.

### 5.1. Experiments

This paper uses the wafer manufacturing problem [4] as a useful example to illustrate the experiment. A wafer manufacturing process in a semiconductor foundry is used to illustrate the proposed approach. The process comprises the following steps: crystal growing, wafer cutting, edge rounding, lapping, etching, polishing, cleaning, final inspection, packaging, and shipping. The wafer cleaning step mainly uses DI (deionized, ultrapure) water to remove debris left over from the mounting wax and/or polishing agent. A stable water supply system to deliver ultrapure water for wafer cleaning is therefore vital in semiconductor manufacturing. The knowledge retrieval technique is used to explore the knowledge base log which includes 1,077 relevant data records of wafer cleaning step in a wafer manufacturing process. The discovered data records involve with 72 situations from 7 interdatabases, 23 workers, and 238 actions. The 5 domain experts assist this experiment to carry out and evaluate.

#### 5.1.1. Problem-Solving Action Formalization and Grey Relational Utility Model

When the worker suffers from a specific problem situation, there are various suppliers providing the problem-solving actions. The problem-solving action formalization and a grey relational utility model are used to precompute the worker's expected list of supplied action QoS items and facilitate a multicriteria decision analysis to discover an optimal selection order of candidate actions.

First, the problem-solving action formalization process identifies the worker, action, and action providers. Then, the worker can decide the indicators (quality of service items, QoS items) of current problem situation. We use abnormal situation of wafer cleaning as a simple example. The worker sets performance, quality, and duration time as the QoS items for abnormal situation. Then the relevant values of QoS items and actions are recorded in a table, as shown in Table 1. For example, action A sets the QoS item, abnormal situation, where the performance degree is high, quality degree is middle, and duration time is evaluated as slow.

After the problem-solving action formalization process, a grey relational utility model is developed to represent worker satisfaction with the action acquisition. The action C is filtered out because of its low grey relational grade, and the reminder actions A, B, D are considered as the candidate actions for solving a specific problem situation. Each QoS item is normalized and scaled to [0, 1]. Then, Table 1 is transformed into Table 2.

**Table 1:** Property of QoS item and action for abnormal situation.

	Performance	Quality	Duration time
Action A	High	Middle	Slow
Action B	Low	High	Normal
Action C	Low	Low	Slow
Action D	Middle	Low	Quick

**Table 2:** Transformed property of QoS item and action for abnormal situation.

	Performance	Quality	Duration time
Action A	0.34	0.32	0.22
Action B	0.31	0.35	0.25
Action D	0.32	0.28	0.30

### 5.1.2. The Selection Order Discovery of Candidate Actions

A modified version of the ELECTRE method [17] is used to determine the optimal selection order of candidate actions to solve a specific problem situation. The decision matrix  $Q$  of expected values can be shown as follows:

$$Q = \begin{bmatrix} 0.34 & 0.32 & 0.22 \\ 0.31 & 0.35 & 0.25 \\ 0.32 & 0.28 & 0.30 \end{bmatrix}. \quad (5.1)$$

The weighted matrix ( $W$ ) for each QoS item is shown as follows:

$$W = \begin{bmatrix} 0.4 & 0 & 0 \\ 0 & 0.35 & 0 \\ 0 & 0 & 0.25 \end{bmatrix}. \quad (5.2)$$

The multiplication of a normalization matrix  $Q$  and a weighted matrix  $W$  which gets the weighted normalization decision matrix  $V$  ( $V = QW$ ) is shown as follows:

$$V = \begin{bmatrix} 0.136 & 0.112 & 0.055 \\ 0.124 & 0.1225 & 0.0625 \\ 0.128 & 0.098 & 0.075 \end{bmatrix}. \quad (5.3)$$

The concordance set  $C_{ij}$  or the discordance set  $D_{ij}$  is shown as follows:

$$\begin{aligned} C_{12} &= \{1\}, & D_{12} &= \{2, 3\}, & C_{13} &= \{1, 2\}, & D_{13} &= \{3\}, \\ C_{21} &= \{2, 3\}, & D_{21} &= \{1\}, & C_{23} &= \{2\}, & D_{23} &= \{1, 3\}, \\ C_{31} &= \{3\}, & D_{31} &= \{1, 2\}, & C_{32} &= \{1, 3\}, & D_{32} &= \{2\}. \end{aligned} \quad (5.4)$$

The sum of each component's weight forms a concordance matrix  $C$ :

$$C_{13} = \frac{\sum_{k \in C_{13}} w_k}{\sum_{k=1}^3 w_k} = \frac{W_1 + W_2}{W_1 + W_2 + W_3} = 0.75,$$

$$C = \begin{bmatrix} - & 0.4 & 0.75 \\ 0.6 & - & 0.35 \\ 0.25 & 0.65 & - \end{bmatrix}. \quad (5.5)$$

A discordance matrix can be presented as  $D$ :

$$D_{13} = \frac{\max_{k \in D_{13}} \{|v_{1k} - v_{3k}|\}}{\max_{k \in S} \{|v_{1k} - v_{3k}|\} \max} = \frac{\max\{0.02\}}{\max\{0.008, 0.014, 0.02\}} = 1,$$

$$D = \begin{bmatrix} - & 0.875 & 1 \\ 1 & - & 0.51 \\ 0.7 & 1 & - \end{bmatrix}. \quad (5.6)$$

A modified discordance matrix can be presented as  $D'$ :

$$D' = \begin{bmatrix} - & 0.125 & 0 \\ 0 & - & 0.49 \\ 0.3 & 0 & - \end{bmatrix}. \quad (5.7)$$

A modified total matrix can be presented as  $A$ :

$$A = \begin{bmatrix} - & 0.05 & 0 \\ 0 & - & 0.1715 \\ 0.075 & 0 & - \end{bmatrix}. \quad (5.8)$$

A modified total superiority matrix is shown as  $E'$ :

$$E' = \begin{bmatrix} - & 1 & 0 \\ 0 & - & 1 \\ 1 & 0 & - \end{bmatrix}. \quad (5.9)$$

Finally, the optimal selection order is determined for all candidate actions. The experiment results show that action  $B$  is better than action  $D$  and action  $D$  is better than action  $A$ . The worker can follow the selection order to get a reasonable action.

**Table 3:** The experimental results of abnormal situation of wafer cleaning step.

Method	Item		
	Candidate actions	Precision	Recall
The method in [4]	32	62.8% (86/137)	75.4% (86/114)
This paper's method	26	68.3% (86/126)	83.5% (86/103)

### 5.2. Experimental Results and Relevant Discussions

This work used an actual abnormal situation of wafer cleaning step in a wafer manufacturing process use case of a high-tech company to demonstrate that the proposed approach is effective. Supplying an adaptive problem-solving solution to a worker will help the business enterprise improve the service and quality. In the experiment of an actual abnormal situation of wafer cleaning step in a wafer manufacturing process use case, the experimental results are shown in Table 3.

In knowledge base log of the wafer cleaning step, a method proposed in [4] and this paper's method are enforced to the experiments. The method proposed in [4] means that worker follows the experiential rules from the knowledge discovery process to take a problem-solving action. The experimental result shows that precision is 62.8% and recall is 75.4%. This paper's method uses a grey relational utility model to filter out low correlation actions. The candidate actions for an abnormal situation decrease from 32 to 26 actions. The experimental result shows that precision is 68.3% and recall 83.5%. The selection method used in this work seems to be more effective than the method proposed in [4].

In the experiment process and result analysis, this research found that weight value in multicriteria decision analysis tasks and worker feedbacks are the critical factors that influenced the experimental results. For example, the weight and normalization values are indistinguishable. These situations prevent the system from identifying the best solution for recommendation. This study checks and adjusts the weight and normalization values to enhance the distinguishing ability. The worker feedback influences how to decide the QoS items. The QoS item is the critical factor for the grey relational utility model and the multicriteria decision analysis processing.

## 6. Conclusion

In business enterprises, especially the manufacturing industry, various problem situations may occur during the production process. In a problem-solving process, how to determine an action needed to be taken to resolve the situation becomes an important issue. This work proposes a selection approach for optimized problem-solving process to assist workers in taking a reasonable action. A grey relational utility model and a multicriteria decision analysis are used to determine the optimal selection order of candidate actions. The selection order is presented to the worker as an adaptive recommended solution. The worker fetches a reasonable problem-solving action based on the selection order. The contribution of this research is in demonstrating a method which is easy to implement in a problem-solving knowledge recommendation system for selecting a reasonable solution.

A high-tech company's knowledge base log is used for an experiment. In the experiment process and result analysis, this research found that weight value in a multicriteria decision analysis task and worker feedback influenced the experimental results. Future work should pay more attention to designing a worker feedback mechanism for

QoS item identification. The worker feedback would help the proposed selection approach by intelligent tuning and learning to improve the service quality incrementally. The recommended technique is to consider combining with more intelligent methods to enhance the effect.

## References

- [1] K. H. Chang, P. Raman, W. H. Carlisle, and J. H. Cross, "A self-improving helpdesk service system using case-based reasoning techniques," *Computers in Industry*, vol. 30, no. 2, pp. 113–125, 1996.
- [2] M. K. Park, I. Lee, and K. M. Shon, "Using case based reasoning for problem solving in a complex production process," *Expert Systems with Applications*, vol. 15, no. 1, pp. 69–75, 1998.
- [3] J. S. Heh, "Evaluation model of problem solving," *Mathematical and Computer Modelling*, vol. 30, no. 11-12, pp. 197–211, 1999.
- [4] D. R. Liu and C. K. Ke, "Knowledge support for problem-solving in a production process: a hybrid of knowledge discovery and case-based reasoning," *Expert Systems with Applications*, vol. 33, no. 1, pp. 147–161, 2007.
- [5] G. C. Silaghi, A. E. Arenas, and L. M. Silva, "A Utility-based reputation model for service-oriented computing," *Towards Next Generation Grids*, pp. 63–72, 2007.
- [6] W. C. Chang, K. L. Wen, H. S. Chen, and T. C. Chang, "Selection model of pavement material via Grey relational grade," in *Proceeding of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 3388–3391, October 2000.
- [7] J. Deng, "Grey information space," *The Journal of Grey System*, vol. 1, pp. 103–117, 1989.
- [8] A. R. Afshari, M. Mojahed, R. M. Yusuff, T. S. Hong, and M. Y. Ismail, "Personnel selection using ELECTRE," *Journal of Applied Sciences*, vol. 10, no. 23, pp. 3068–3075, 2010.
- [9] J. Allen, N. Blaylock, and G. Ferguson, "A problem solving model for collaborative agents," in *Proceedings of the 1st International Joint Conference on: Autonomous Agents adn Multiagent Systems*, pp. 774–781, Bologna, Italy, July 2002.
- [10] B. S. Yang, T. Han, and Y. S. Kim, "Integration of ART-Kohonen neural network and case-based reasoning for intelligent fault diagnosis," *Expert Systems with Applications*, vol. 26, no. 3, pp. 387–395, 2004.
- [11] S. Guardati, "RBCShell: a tool for the construction of systems with case-based reasoning," *Expert Systems with Applications*, vol. 14, no. 1-2, pp. 63–70, 1998.
- [12] S. H. Liao, "Problem solving and knowledge inertia," *Expert Systems with Applications*, vol. 22, no. 1, pp. 21–31, 2002.
- [13] B. C. Jiang, S. L. Tasi, and C. C. Wang, "Machine vision-based gray relational theory applied to IC marking inspection," *IEEE Transactions on Semiconductor Manufacturing*, vol. 15, no. 4, pp. 531–539, 2002.
- [14] Y. T. Hsu, C. B. Lin, and S. F. Su, "High noise vehicle plate recognition using grey system," *The Journal of Grey System*, vol. 10, pp. 193–208, 1998.
- [15] S. J. Huang, N. H. Chiu, and L. W. Chen, "Integration of the grey relational analysis with genetic algorithm for software effort estimation," *European Journal of Operational Research*, vol. 188, no. 3, pp. 898–909, 2008.
- [16] L. Antonieta and S. S. Vasco, "Multi criteria decision making models: an overview on electre methods," Universidade Portucalense, Centro de Investigação em Gestão e Economia (CIGE), Working paper, 2011.
- [17] Y. L. Chi, C. W. Lee, and C. Y. Chen, "A selection approach for optimized web services compositions," *Electronic Commerce Studies*, vol. 2, no. 3, pp. 297–314, 2004.
- [18] T. H. Davenport and L. Prusak, *Working Knowledge: how Organizations Manage What They Know*, Harvard Business School Press, Boston, Mass, USA, 1998.
- [19] P. H. Gray, "The impact of knowledge repositories on power and control in the workplace," *Information Technology & People*, vol. 14, no. 4, pp. 368–384, 2001.
- [20] K.D. Fenstermacher, "Process-aware knowledge retrieval," in *Proceedings of the 35th Hawaii International Conference on System Sciences*, pp. 209–217, Big Island, Hawaii, USA, 2002.
- [21] M. M. Kwan and P. Balasubramanian, "KnowledgeScope: managing knowledge in context," *Decision Support Systems*, vol. 35, no. 4, pp. 467–486, 2003.
- [22] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, The ACM Press, New York, NY, USA, 1999.

- [23] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.
- [24] J. L. Herlocker and J. A. Konstan, "Content-independent task-focused recommendation," *IEEE Internet Computing*, vol. 5, no. 6, pp. 40–47, 2001.
- [25] S. E. Middleton, N. R. Shadbolt, and D. C. De Roure, "Ontological user profiling in recommender systems," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 54–88, 2004.
- [26] M. Pazzani and D. Billsus, "Learning and revising user profiles: the identification of interesting web sites," *Machine Learning*, vol. 27, no. 3, pp. 313–331, 1997.
- [27] A. Abecker, A. Bernardi, H. Maus, M. Sintek, and C. Wenzel, "Information supply for business processes: coupling workflow with document analysis and information retrieval," *Knowledge-Based Systems*, vol. 13, no. 5, pp. 271–284, 2000.
- [28] A. Agostini, S. Albolino, R. Boselli, G. De Michelis, F. De Paoli, and R. Dondi, "Stimulating knowledge discovery and sharing," in *Proceedings of the International ACM Conference on Supporting Group Work*, pp. 248–257, Sanibel Island, Fla, USA, November 2003.
- [29] N. J. Davies, A. Duke, and A. Stonkus, "OntoShare: evolving ontologies in a knowledge sharing system," in *Towards the Semantic Web—Ontology-Based Knowledge Management*, N. J. Davies, D. Fensel, and F. van Harmelen, Eds., pp. 161–176, Wiley, London, UK, 2003.
- [30] D. R. Liu, I. C. Wu, and K. S. Yang, "Task-based script K sign-Support system: disseminating and sharing task-relevant knowledge," *Expert Systems with Applications*, vol. 29, no. 2, pp. 408–423, 2005.
- [31] M. Nagai, D. Yamaguchi, and G. D. Li, "Grey structural modeling," *Journal of Grey System*, vol. 8, no. 2, pp. 119–130, 2005.
- [32] D. Yamaguchi, G. D. Li, and M. Nagai, "Verification of effectiveness for grey relational analysis models," *Journal of Grey System*, vol. 10, no. 3, pp. 169–182, 2007.
- [33] D. Yamaguchi, G. D. Li, K. M. T. Akabane, and M. Kitaoka, "A realization algorithm of grey structural modeling," *Journal of Grey System*, vol. 10, no. 1, pp. 33–40, 2007.

*Research Article*

## **Predictor-Corrector Primal-Dual Interior Point Method for Solving Economic Dispatch Problems: A Postoptimization Analysis**

**Antonio Roberto Balbo,<sup>1</sup> Márcio Augusto da Silva Souza,<sup>2</sup> Edméa Cássia Baptista,<sup>1</sup> and Leonardo Nepomuceno<sup>3</sup>**

<sup>1</sup> Department of Mathematics, Universidade Estadual Paulista (UNESP), 17033-360 Bauru, SP, Brazil

<sup>2</sup> Graduate Program in Electrical Engineering, Universidade Estadual Paulista (UNESP), 17033-360 Bauru, SP, Brazil

<sup>3</sup> Department of Electrical Engineering, UNESP—Universidade Estadual Paulista, 17033-360 Bauru, SP, Brazil

Correspondence should be addressed to Leonardo Nepomuceno, leo@feb.unesp.br

Received 9 November 2011; Accepted 3 April 2012

Academic Editor: Jianming Shi

Copyright © 2012 Antonio Roberto Balbo 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 proposes a predictor-corrector primal-dual interior point method which introduces line search procedures (IPLS) in both the predictor and corrector steps. The Fibonacci search technique is used in the predictor step, while an Armijo line search is used in the corrector step. The method is developed for application to the economic dispatch (ED) problem studied in the field of power systems analysis. The theory of the method is examined for quadratic programming problems and involves the analysis of iterative schemes, computational implementation, and issues concerning the adaptation of the proposed algorithm to solve ED problems. Numerical results are presented, which demonstrate improvements and the efficiency of the IPLS method when compared to several other methods described in the literature. Finally, postoptimization analyses are performed for the solution of ED problems.

### **1. Introduction**

Since its introduction in 1984, the projective transformation algorithm proposed by Karmarkar in [1] has proved to be a notable interior point method for solving linear programming problems (LPPs). This pioneer study caused an upheaval in research activities in this area. Among all the variations of Karmarkar's original algorithm, the first to attract the attention of researchers was the one that uses a simple affine transformation to replace Karmarkar's original and highly complex projective transformation, enabling work with

LPP in its standard form. The affine algorithm was first introduced in [2] by Dikin, a Soviet mathematician. Later, in 1986, the work was independently rediscovered by Barnes in [3] and by Vanderbei et al. in [4]. They proposed using the primal-affine algorithm to solve LPP in standard form and also presented proof of the algorithm's convergence. A similar algorithm, called dual-affine, was developed and implemented by Adler et al. [5] to solve the LPP in the form of inequality. Compared to the relatively cumbersome projective transformation, the implementation of the primal-affine and dual-affine algorithms was simpler because of its direct relationship with the LPP. These two algorithms produced promising results when applied to large problems [6], although the theoretical proof of polynomial time complexity was not obtained from the affine transformation. Megiddo and Shub's study in [7] indicated that the trajectory that leads to the optimal solution provided by affine algorithms depends on the initial solution. A poor initial solution, which is close to a viable domain vertex, could result in an investigation that covers all vertex problems.

Nevertheless, the polynomial time complexity of primal-affine and dual-affine algorithms can be re-established by incorporating a logarithmic barrier function into the objective function of the original LPP. The purpose of this procedure is to solve the problem pointed out by Megiddo, that is, to prevent an interior solution from becoming "trapped" at the border of the problem (possibly a vertex). This procedure also provides proof of the complexity of the method. The idea of using the logarithmic barrier function method for convex programming problems was developed by Fiacco and McCormick in [8, 9], based on the method proposed by Frisch in [10]. After the introduction of Karmarkar's algorithm in 1984, the logarithmic barrier function method was reconsidered to solve linear programming problems. Gill et al. used this method in [11] to develop a projected Newton barrier method and demonstrated its equivalence with Kamarkar's projective algorithm. The methods proposed, among others, by Ye in [12], Renegar in [13], Vaidya in [14], and Megiddo in [15], as well as those of a central trajectory—called path-following methods, which were proposed by Gonzaga in [16, 17] and Monteiro and Adler in [18], use the objective function augmented by the logarithmic barrier function. A third variation, the so-called primal-dual interior point algorithm, was introduced by Monteiro et al. in [19] and by Kojima et al. in [20]. This algorithm explores a potential primal-dual function, a variation of the logarithmic barrier function, called the potential function. The polynomial time complexity theory was successfully demonstrated by Kojima et al. in [20] and Monteiro et al. in [19] based on Megiddo's work, which provided a theoretical analysis for the logarithmic barrier method and proposed the primal-dual approach.

The predictor-corrector procedure initially defined by Mehrotra and Sun in [21] and implemented in by Lustig et al. in [22] explored variant directions of the primal-affine interior point methods in the predictor step. In the corrector step, the point previously obtained (in the predictor step) was "centralized" to exploit the potential function related to the logarithmic barrier function. This procedure significantly improved the performance of the primal-dual interior point methods.

This strategy was reviewed and modified by Wu et al. in [23] and successfully applied to solve optimal power flow problems. Wu used the logarithmic barrier penalization, the Newton method, and first-order approximations in the predictor step to determine search directions and the approximate solution. Second-order approximations were considered in the corrector step to refine solutions obtained in the predictor step.

The methods related to the primal-dual interior point methodology, especially those proposed by Kojima et al. in [20], Monteiro and Adler in [18], and Monteiro et al. in [19], which were broadly investigated by Fang and Puthenpura in [24], have been explored in this past decade to solve linear, nonlinear, and integer mathematical programming problems in several fields of research.

This paper proposes a primal-dual interior point method that uses the predictor-corrector strategy described by Lustig et al. in [22] and Wu et al. in [23] and incorporates line search procedures in both the predictor and corrector steps. Theoretical aspects as well as iterative schemes and computational implementation are investigated. A Fibonacci line search procedure is carried out in the predictor step and an Armijo line search is used in the corrector step to calculate the step sizes while taking into account constraints in the variables of the problem. The line search procedures adopted are aimed at improving the overall convergence of the proposed method for solving quadratic programming problems. The method is applied to solve the economic dispatch (ED) problem, a classical quadratic problem studied in the field of electrical power systems. The results obtained with the proposed method are compared with those obtained by several others described in the literature for solving ED problems, such as the primal-dual method described in [6], evolutionary algorithms found in [25–27], genetic and coevolutionary genetic algorithms described by Samed in [27], the cultural algorithm described by Rodrigues in [26], and a hybrid atavistic genetic algorithm given in [25]. This comparative investigation demonstrates the efficiency of the proposed IPLS method.

This paper is organized as follows: Section 2 presents the ED problem; Section 3 develops the theory of the proposed IPLS method and presents its algorithm; Section 4 describes a computational implementation of the method, applying it to solve ED problems. This section also includes a postoptimization analysis. Finally, conclusions are drawn in Section 5.

## 2. The Economic Dispatch Problem

The economic dispatch problem (ED) is defined as an optimal allocation process of electricity demands among available generating units, where operational constraints must be satisfied while minimizing generation costs. Happ reported in [28] that, by 1920, several engineers had become aware of the economic allocation problem. According to the aforementioned author, one of the first methods employed to solve the ED problem was to request power from the most efficient unit (the merit order loading method). This method was based on the following idea: the next incremental active power was to be supplied by the most efficient plant, until it reached its maximum operational point, and so on, successively. Although this method failed to minimize costs, it was employed until 1930, when the equal incremental cost criterion began to produce better results.

The idea behind the method of incremental costs is that the next incremental system load increase should be allocated to the unit with the lowest incremental cost, which is determined by measuring the derivative of the cost curve. Steinberg and Smith mathematically proved in [29] the equal incremental costs criterion, which was already being used empirically. Around 1931, this method had already become well established. In theory, the method described by Steinberg and Smith in [29] also ensures that if there are no active constraints at the point of optimal operation, the incremental costs of all units should be equal. This rule is still widely used by power system operators today.

## 2.1. Optimization Model for the Classical Economic Dispatch Problem

The economic dispatch (ED) problem is concerned with minimizing active power production costs while meeting the system demand and taking into account the operational limits of all generating units. The ED problem is mathematically described in

$$\begin{aligned} \text{Minimize } & C_T = \sum_{i=1}^n \left( C_i(P_i) = a_i P_i^2 + b_i P_i + c_i \right) \\ \text{subject to } & \sum_{i=1}^n P_i = D \\ & P_i^{\min} \leq P_i \leq P_i^{\max}; \quad i = 1, \dots, n, \end{aligned} \tag{2.1}$$

where  $C_T$ : total cost function for generating units,  $n$ : number of generating units,  $C_i(P_i)$ : cost of generation unit  $i$  (without considering the valve-point effect),  $a_i, b_i, c_i$ : coefficient of the cost function for generating unit  $i$ ,  $P_i$ : power output of generating unit  $i$ ,  $D$ : total power demand,  $P_i^{\min}$ : minimum power output limit for generating unit  $i$ , and  $P_i^{\max}$ : maximum power output limit for generating unit  $i$ .

The objective function in (2.1) may also represent the so-called valve-point effects, as in [28], which are associated with the opening of pressure valves at some specific operating points. In such cases,  $C_i(P_i)$  is mathematically described as in

$$C_i(P_i) = a_i P_i^2 + b_i P_i + c_i + \left| e_i \sin(f_i(P_i^{\min} - P_i)) \right|. \tag{2.2}$$

The cost function  $C_i(P_i)$  described in (2.2) is continuous but not differentiable. Although it is more representative, function (2.2) makes ED a much more complex problem to solve due to its characteristics of nondifferentiability.

The literature describes several different methodologies to solve ED problems. In those studies, the methodology associated with evolutionary algorithms stands out, especially when issues related to nondifferentiability (such as those described in (2.2)) are involved. Evolutionary algorithms have been used to solve ED problems because they are able to find optimal solutions even when the objective function and/or constraints are not continuous or non-differentiable. Numerical problems related to evolutionary algorithms involve the inability to verify the optimality conditions associated with the solutions obtained and also the computational effort necessary to obtain the solutions, especially for large systems.

Optimal solutions to ED problems have also been investigated through traditional nonlinear programming methods, including interior point methods [6, 23, 30]. This paper proposes a predictor-corrector primal-dual interior point method for solving ED problems, which incorporates line search procedures in both the predictor and corrector steps. The results presented here demonstrate that the proposed method improves the solution of ED problems. The following section examines the application of the primal-dual interior point method to solve quadratic programming problems.

### 3. Solution Technique for the Quadratic Programming Problem

Quadratic programming problems (QPPs) represent a special class of nonlinear programming in which the objective function is quadratic and the constraints are linear [31]. The problem is expressed mathematically by

$$\begin{aligned} \text{Minimize } & \frac{1}{2}x^T Qx + c^T x \\ \text{subject to } & Ax = b \\ & l \leq x \leq u, \end{aligned} \tag{3.1}$$

where  $A \in \mathbb{R}^{m \times n}$  so that  $\text{rank}(A) = n$ ,  $b \in \mathbb{R}^m$ ,  $x \in \mathbb{R}^n$ ,  $c \in \mathbb{R}^n$ ,  $u \in \mathbb{R}^n$ ,  $l \in \mathbb{R}^n$ , and  $Q \in \mathbb{R}^{n \times n}$  is a diagonal matrix. Problem (3.1) is equivalent to (3.2), where  $z \in \mathbb{R}^n$  is a slack variable and  $r \in \mathbb{R}^n$  is a surplus variable:

$$\begin{aligned} \text{Minimize } & \frac{1}{2}x^T Qx + c^T x \\ \text{subject to } & Ax = b \\ & x + z = u; \quad z \geq 0 \\ & x - r = l; \quad r \geq 0. \end{aligned} \tag{3.2}$$

For  $\mu > 0$ , it is possible to incorporate a logarithmic barrier function to the objective function of (3.2) and eliminate inequality constraints. This procedure results in the following nonlinear optimization problem:

$$\begin{aligned} \text{Minimize } & F_\mu(x, r, z) = \frac{1}{2}x^T Qx + c^T x - \mu \sum_{j=1}^n \ln r_j - \mu \sum_{j=1}^n \ln z_j \\ \text{subject to } & Ax = b \\ & x + z = u \\ & x - r = l. \end{aligned} \tag{3.3}$$

The Lagrangian function related to (3.3) is expressed by

$$\begin{aligned} L_\mu(x, r, z, w, s, y) = & \frac{1}{2}x^T Qx + c^T x + w^T(b - Ax) + s^T(l - x + r) + y^T(x + z - u) \\ & - \mu \sum_{j=1}^n \ln s_j - \mu \sum_{j=1}^n \ln y_j, \end{aligned} \tag{3.4}$$

where the dual variables associated with the three equality constraints in (3.3) are, respectively,  $w \in \mathbb{R}^m$ ,  $y \in \mathbb{R}^n$ ,  $s \in \mathbb{R}^n$ . The optimal Karush-Kuhn-Tucker (KKT) conditions for problem (3.1) are depicted in

$$\begin{aligned}\frac{\partial L_\mu}{\partial x} = 0 &\iff -Qx - c + A^T w + s - y = 0, \\ \frac{\partial L_\mu}{\partial w} = 0 &\iff Ax = b, \\ \frac{\partial L_\mu}{\partial y} = 0 &\iff x + z = u, \\ \frac{\partial L_\mu}{\partial s} = 0 &\iff x - r = l, \\ \frac{\partial L_\mu}{\partial z} = 0 &\iff ZYe - \mu e = 0, \\ \frac{\partial L_\mu}{\partial r} = 0 &\iff RSe - \mu e = 0,\end{aligned}\tag{3.5}$$

where  $R, Z, S$  and  $Y$  are diagonal matrices whose diagonal elements are  $r_i, z_i, s_i$ , and  $y_i$ ;  $i = 1, \dots, n$ , respectively;  $e = (1, \dots, 1)^T$ ;  $\mu$  is a dual metric or adjustment parameter for the curve defined by the central trajectory (path-following parameter). The set  $\Omega^0$  given in (3.6) is defined to simplify to notation. This set describes interior points for problem (3.2) and its corresponding dual problem:

$$\begin{aligned}\Omega^0 = \Big\{ (x, w, z, r, y, s) \mid & -Qx + A^T w + s - y = c, \\ & Ax = b, \quad x + z = u, \quad x - r = l, \quad (z, r, y, s) > 0 \Big\}.\end{aligned}\tag{3.6}$$

Equations (3.4), (3.5), and (3.6) are considered in the following sections to perform the analysis of important issues concerning the proposed IPLS method, such as search directions, step sizes, stopping criteria, and updating of the barrier parameter.

### 3.1. Search Directions

In this section, the search directions used in the proposed IPLS method are investigated. In Section 3.2, the search directions for the predictor step are calculated while the search directions for the corrector step are determined in Section 3.3. The proposed strategy is a variant of the approach developed by Wu et al. in [23].

### 3.2. Search Directions—Predictor Step

Let us suppose that, in an iteration  $k$ , a point  $h^k$  satisfies the primal and dual feasibility conditions expressed by (3.5). In this case, the definition of the new point  $h^{k+1}$  depends solely on the calculation of a search direction and on a step size in such a direction. Disregarding the step size in the iteration  $k + 1$ , the new point  $h^{k+1}$  is defined by the following equation:

$$h^{k+1} = \begin{pmatrix} x^{k+1} \\ w^{k+1} \\ z^{k+1} \\ r^{k+1} \\ y^{k+1} \\ s^{k+1} \end{pmatrix} = \begin{pmatrix} x^k + d_x^k \\ w^k + d_w^k \\ z^k + d_z^k \\ r^k + d_r^k \\ y^k + d_y^k \\ s^k + d_s^k \end{pmatrix}. \quad (3.7)$$

Therefore, it is necessary to determine the direction of the movement  $d^k$  to obtain the new point  $h^{k+1}$ . Following the steps in the Newton method applied to the nonlinear system (3.5),  $d^k$  can be obtained by solving for the system (3.8), which is equivalent to (3.9):

$$J(h^{(k)})d^{(k)} = -F(h^{(k)}), \quad (3.8)$$

$$\begin{pmatrix} A & 0 & 0 & 0 & 0 & 0 \\ -Q & A^T & 0 & 0 & -I & I \\ I & 0 & I & 0 & 0 & 0 \\ I & 0 & 0 & -I & 0 & 0 \\ 0 & 0 & Y & 0 & Z & 0 \\ 0 & 0 & 0 & S & 0 & R \end{pmatrix} \begin{pmatrix} d_x^k \\ d_w^k \\ d_z^k \\ d_r^k \\ d_y^k \\ d_s^k \end{pmatrix} = \begin{pmatrix} t^k \\ g^k \\ f^k \\ o^k \\ q^k \\ v^k \end{pmatrix}, \quad (3.9)$$

where the residuals of (3.9) for the predictor step are expressed as in

$$\begin{aligned} t^k &= b - Ax^k; & g^k &= Qx^k + c - A^Tw^k - s^k + y^k; & f^k &= u - x^k - z^k; \\ o^k &= l - x^k + r^k; & q^k &= \mu_k e - Z_k Y_k e; & v^k &= \mu_k e - R_k S_k e; & e &= (1, 1, \dots, 1)^T. \end{aligned} \quad (3.10)$$

The directions  $d_x^k, d_w^k, d_z^k, d_r^k, d_y^k, d_s^k$  to be determined in the predictor step use the residuals defined in (3.10), resulting in (3.11)–(3.16), as follows:

$$Ad_x^k = t^k, \quad (3.11)$$

$$-Qd_x^k + A^T d_w^k + d_s^k - d_y^k = g^k, \quad (3.12)$$

$$d_x^k + d_z^k = f^k, \quad (3.13)$$

$$d_x^k - d_r^k = o^k, \quad (3.14)$$

$$Y_k d_z^k + Z_k d_y^k = q^k, \quad (3.15)$$

$$S_k d_r^k + R_k d_s^k = v^k. \quad (3.16)$$

From (3.13) and (3.14), we obtain (3.17) and (3.18), respectively:

$$d_z^k = -d_x^k + f^k, \quad (3.17)$$

$$d_r^k = d_x^k - o^k. \quad (3.18)$$

Isolating  $d_y^k$  and  $d_s^k$  in (3.15) and (3.16), respectively, leads to (3.19) and (3.20), as follows:

$$d_y^k = Z_k^{-1} (q^k - Y_k d_z^k), \quad (3.19)$$

$$d_s^k = R_k^{-1} (v^k - S_k d_r^k). \quad (3.20)$$

Combining the results found in (3.17), (3.18), (3.19), and (3.20) with those given in (3.11)–(3.16), and considering (3.21), yields the directions (3.22) and (3.23):

$$\theta = (R_k^{-1} S_k + Z_k^{-1} Y_k + Q)^{-1}, \quad (3.21)$$

$$d_w^k = (A\theta A^T)^{-1} [A\theta (g^k + p^k) + t^k], \quad (3.22)$$

$$d_x^k = \theta (A^T d_w^k - g^k - p^k), \quad (3.23)$$

where

$$p^k = R_k^{-1} (S_k o^k - v^k) + Z_k^{-1} (q^k - Y_k f^k). \quad (3.24)$$

Note that, after calculating (3.23), the remaining components of the direction vector  $d_z^k, d_r^k, d_y^k$ , and  $d_s^k$  in (3.17), (3.18), (3.19), and (3.20), respectively, are easily calculated. Since matrix  $A\theta A^T$  is symmetrical and positive definite in (3.22) (considering that  $Q$  is symmetrical and positive definite),  $d_w^k$  can be determined by using the Cholesky decomposition.

### 3.3. Search Directions—Corrector Step

Analogously to the procedure carried out in Section 3.2, this section describes the calculation of the search direction for the corrector step  $\tilde{d}^k$ , which is obtained by solving the linear system (3.25), as follows:

$$J(h^k)\tilde{d}^k = -\tilde{F}(h^k), \quad (3.25)$$

where  $\tilde{F}(h^k)$  is obtained by considering second-order approximations in the residuals (3.10) (calculated in the predictor step), so that (3.25) is equivalent to

$$\begin{pmatrix} A & 0 & 0 & 0 & 0 & 0 \\ -Q & A^T & 0 & 0 & -I & I \\ I & 0 & I & 0 & 0 & 0 \\ I & 0 & 0 & -I & 0 & 0 \\ 0 & 0 & Y & 0 & Z & 0 \\ 0 & 0 & 0 & S & 0 & R \end{pmatrix} \begin{pmatrix} \tilde{d}_x^k \\ \tilde{d}_w^k \\ \tilde{d}_z^k \\ \tilde{d}_r^k \\ \tilde{d}_y^k \\ \tilde{d}_s^k \end{pmatrix} = \begin{pmatrix} \tilde{t}^k \\ \tilde{g}^k \\ \tilde{f}^k \\ \tilde{o}^k \\ \tilde{q}^k \\ \tilde{v}^k \end{pmatrix}, \quad (3.26)$$

where

$$\begin{aligned} \tilde{t}^k &= b - Ax^k; & \tilde{g}^k &= Qx^k + c - A^Tw^k - s^k + y^k; & \tilde{f}^k &= u - x^k - z^k; \\ \tilde{o}^k &= l - x^k + r^k; & \tilde{q}^k &= \mu_k e - Z_k Y_k e - D_z^k D_y^k e; & \tilde{v}^k &= \mu_k e - R_k S_k e - D_x^k D_s^k e, \end{aligned} \quad (3.27)$$

$D_x^k, D_z^k, D_y^k$  and  $D_s^k$  are diagonal matrices whose diagonal components are  $(d_x^k)_i, (d_z^k)_i, (d_y^k)_i$  and  $(d_s^k)_i, i = 1, \dots, n$ , respectively.

The calculation of the residuals  $t^k, g^k, f^k, o^k, q^k, v^k$ , described in (3.10), and of  $\tilde{t}^k, \tilde{g}^k, \tilde{f}^k, \tilde{o}^k, \tilde{q}^k, \tilde{v}^k$ , described in (3.27), basically distinguishes the predictor and corrector steps in the proposed method. It is important to note that the corrector step procedure uses direction values  $d_x^k, d_z^k, d_y^k$ , and  $d_s^k$ , which have already been calculated in the predictor step, to redefine residuals  $\tilde{q}^k$  and  $\tilde{v}^k$  in (3.27). Using (3.25) and (3.26) and following the same steps

taken to determine the directions of the predictor step, as seen in Section 3.2, the components of the direction vector  $\tilde{d}^k$  can be calculated using the following:

$$\tilde{d}_w^k = (A\theta A^T)^{-1} [A\theta(g^k + \tilde{p}^k) + \tilde{l}^k], \quad (3.28)$$

$$\tilde{d}_x^k = \theta(A^T \tilde{d}_w^k - g^k - \tilde{p}^k), \quad (3.29)$$

$$\tilde{d}_z^k = -\tilde{d}_x^k + \tilde{f}^k, \quad (3.30)$$

$$\tilde{d}_r^k = \tilde{d}_x^k + \tilde{o}^k, \quad (3.31)$$

$$\tilde{d}_y^k = Z_k^{-1}(\tilde{q}^k - Y_k \tilde{d}_z^k), \quad (3.32)$$

$$\tilde{d}_s^k = R_k^{-1}(\tilde{v}^k - S_k \tilde{d}_r^k), \quad (3.33)$$

where

$$\tilde{p}^k = R_k^{-1}(S_k \tilde{o}^k - \tilde{v}^k) + Z_k^{-1}(\tilde{q}^k - Y_k \tilde{f}^k), \quad (3.34)$$

and  $\theta$  is defined in (3.21).

### 3.4. Step Size

After calculating the search directions for the predictor and corrector steps, it is possible to move to a new point  $(x^{k+1}, w^{k+1}, z^{k+1}, r^{k+1}, y^{k+1}, s^{k+1})$ , while ensuring that  $s^{k+1} > 0$ ,  $z^{k+1} > 0$ ,  $y^{k+1} > 0$  and  $r^{k+1} > 0$ . In order to ensure nonnegativity constraints over the slack variables, the step to be taken in each direction in both the predictor and corrector steps must be controlled. The basics of this procedure are described in [31] and are discussed below.

#### 3.4.1. Predictor Step Size

Considering the variables defined in the predictor step, the step size for primal and dual variables is calculated as described below:

$$x^{k+1} = x^k + \alpha_{P_k} d_x^k, \quad (3.35)$$

$$r^{k+1} = r^k + \alpha_{P_k} d_r^k, \quad (3.36)$$

$$z^{k+1} = z^k + \alpha_{P_k} d_z^k, \quad (3.37)$$

$$w^{k+1} = w^k + \alpha_{D_k} d_w^k, \quad (3.38)$$

$$y^{k+1} = y^k + \alpha_{D_k} d_y^k, \quad (3.39)$$

$$s^{k+1} = s^k + \alpha_{D_k} d_s^k. \quad (3.40)$$

The step size  $\alpha_{P_k}$  for the primal variables is calculated through

$$\alpha_{P_k} = \text{Min}\{\alpha_P, \alpha_Q, \alpha_{LS}\}, \quad (3.41)$$

where, for  $0 < \alpha < 1$ , the step sizes  $\alpha_P, \alpha_Q, \alpha_{LS}$  are determined as follows.

- (i) The step size for primal variables  $\alpha_P$  is obtained without violating the nonnegativity requirements of the primal variables:

$$\alpha_P = \text{Min}\left\{-\frac{\alpha z_i^k}{d_{z_i}^k}, -\frac{\alpha r_i^k}{d_{r_i}^k} \text{ such that } d_{z_i}^k, d_{r_i}^k < 0\right\}. \quad (3.42)$$

- (ii) The step size  $\alpha_Q$  is the maximum step size possible without increasing the objective value:

$$\alpha_Q = -\frac{\left[(d_x^k)^t (Qx^k + c)\right]}{\left[(d_x^k)^t Q d_x^k\right]}. \quad (3.43)$$

- (iii) The step size  $\alpha_{LS}$  is determined as in (3.44) from the Fibonacci line search strategy, which is briefly summarized as follows:

$$\alpha_{LS} = \text{Min}_{\alpha_k} \left\{ F_\mu \left( \hat{x}^{k+1}(\alpha_k) \right) \right\} = \text{Min}_{\alpha_k} \left\{ F_\mu \left( \hat{x}^k + \alpha_k d_{\hat{x}}^k \right) \right\}, \quad (3.44)$$

where  $F_\mu$  is defined in (3.3).

For notation simplicity, in the summary of the Fibonacci search, the following identities are defined:  $\hat{x}^{k+1}(\alpha_{LS}) = (x^{k+1}(\alpha_{LS}), r^{k+1}(\alpha_{LS}), z^{k+1}(\alpha_{LS}))^T = (x^k + \alpha_{LS} d_x^k, r^k + \alpha_{LS} d_r^k, z^k + \alpha_{LS} d_z^k)^T$ , and  $d_{\hat{x}}^k = (d_x^k, d_r^k, d_z^k)$ ; so that  $d_x^k$  is defined in (3.23),  $d_r^k$  is defined in (3.18), and  $d_z^k$  is defined in (3.17). Only primal variables are considered in the Fibonacci search algorithm used by the IPLS method. Starting at a point  $\hat{x}^k = (x^k, r^k, z^k)^T$ , the algorithm searches a new point  $\hat{x}^{k+1}$  in direction  $d_{\hat{x}}^k$  using the function  $F_\mu$  defined in (3.3). The Fibonacci method calculates  $\alpha_{LS}$  so that the minimization of  $F_\mu$  is ensured.  $\alpha_{LS}$  is determined considering the Fibonacci sequence. The initial value of  $\alpha_{LS}$  is set taking into account the interval of uncertainty  $[0, 1]$ .

The step size  $\alpha_{D_k}$  for the dual variables (3.38)–(3.40) is calculated through (3.45) for  $0 < \alpha < 1$ :

$$\alpha_{D_k} = \text{Min}\left\{1, -\frac{\alpha s_i^k}{d_{s_i}^k}, -\frac{\alpha y_i^k}{d_{y_i}^k} \text{ such that } d_{s_i}^k, d_{y_i}^k < 0\right\}. \quad (3.45)$$

### 3.4.2. Corrector Step Size

In the corrector step, the calculation of step size is defined analogously, considering the variables from the following

$$x^{k+1} = x^k + \beta_{P_k} \tilde{d}_x^k, \quad (3.46)$$

$$r^{k+1} = r^k + \beta_{P_k} \tilde{d}_r^k, \quad (3.47)$$

$$z^{k+1} = z^k + \beta_{P_k} \tilde{d}_z^k, \quad (3.48)$$

$$w^{k+1} = w^k + \beta_{D_k} \tilde{d}_w^k, \quad (3.49)$$

$$y^{k+1} = y^k + \beta_{D_k} \tilde{d}_y^k, \quad (3.50)$$

$$s^{k+1} = s^k + \beta_{D_k} \tilde{d}_y^k, \quad (3.51)$$

where

$$\beta_{P_k} = \text{Min}\{\beta_P, \beta_Q, \beta_{LS}\}, \quad (3.52)$$

while the step sizes  $\beta_P, \beta_Q, \beta_{LS}$  are determined as follows.

- (i) The step size for primal variables  $\beta_P$  is obtained without violating the nonnegativity requirements of the primal variables:

$$\beta_P = \text{Min} \left\{ 1, -\frac{\alpha z_i^k}{\tilde{d}_{z_i}^k}, -\frac{\alpha r_i^k}{\tilde{d}_{r_i}^k} \text{ such that } \tilde{d}_{z_i}^k, \tilde{d}_{r_i}^k < 0 \right\}. \quad (3.53)$$

- (ii) The step size  $\beta_Q$  is the maximum step size possible without increasing the objective value:

$$\beta_Q = -\frac{\left[ (\tilde{d}_x^k)^t (Qx^k + c) \right]}{\left[ (\tilde{d}_x^k)^t Q d_x^k \right]}. \quad (3.54)$$

- (iii) The step size  $\beta_{LS}$  in (3.55) is determined from the Armijo line search:

$$\beta_{LS} = \beta_k, \quad (3.55)$$

so that

$$F_\mu(\hat{x}^{k+1}(\beta_k)) \leq F_\mu(\hat{x}^k) + \beta_k \nabla F_\mu(\hat{x}^k)^T \tilde{d}_{\hat{x}}^k, \quad (3.56)$$

where  $\tilde{x}^{k+1}(\beta_k) = (x^{k+1}(\beta_k), r^{k+1}(\beta_k), z^{k+1}(\beta_k))^T = (x^k + \beta_k \tilde{d}_x^k, r^k + \beta_k \tilde{d}_r^k, z^k + \beta_k \tilde{d}_z^k)^T$ ;  $\tilde{d}_{\tilde{x}}^k = (\tilde{d}_x^k, \tilde{d}_r^k, \tilde{d}_z^k)$ ; so that  $\tilde{d}_x^k, \tilde{d}_r^k, \tilde{d}_z^k$  are defined in (3.29)–(3.31), respectively. In the proposed IPLS method, the Armijo search is also performed in the direction defined by primal variables. Starting at a point  $\tilde{x}^k = (x^k, r^k, z^k)$ , the method searches for a new point  $\tilde{x}^{k+1}$ , in direction  $\tilde{d}_{\tilde{x}}^k$ , so that the function  $F_\mu$  defined in (3.3) decreases. This search calculates  $\beta_{LS}$ , thereby ensuring the reduction of  $F_\mu$ . To prevent oscillations in the iterative process, the initial choice for  $\beta_k$  should not be too high, and to prevent the process from stopping prematurely, it should not be too low. Therefore, this value is generally adjusted to  $\beta_0 = 1$ . If (3.44) is not satisfied,  $\beta_k$  is updated using the following sequence:

$$\beta_{k+1} = \frac{\beta_k}{\delta}, \quad k = 0, 1, \dots, n; \text{ with } \delta > 1, \text{ usually set to } \delta = 2. \quad (3.57)$$

The following equation is used for the analysis of (3.56):

$$\nabla F_\mu(\tilde{x}^k) = (\nabla F_\mu(x^k), \nabla F_\mu(r^k), \nabla F_\mu(z^k))^T = (Qx^k + c, -\mu_k R_k^{-1}e, -\mu_k Z_k^{-1}e)^T \quad (3.58)$$

with  $\mu_k$  determined as in Section 3.6.

The step size for dual variables  $\beta_{D_k}$  is calculated by (3.59), without violating the nonnegativity requirements of the dual variables:

$$\beta_{D_k} = \text{Min} \left\{ 1, -\frac{\alpha s_i^k}{\tilde{d}_{s_i}^k}, -\frac{\alpha y_i^k}{\tilde{d}_{y_i}^k} \text{ such that } \tilde{d}_{s_i}^k, \tilde{d}_{y_i}^k < 0 \right\}. \quad (3.59)$$

The general principle used here to calculate  $\alpha_{P_k}$  and  $\beta_{P_k}$  is to choose a step size that reduces the quadratic objective by a maximum amount without violating the nonnegativity requirements of the primal variables.

The basic idea for defining the line searches in both the predictor and corrector steps is to use a more accurate search for the predictor step (which uses first order approximation to calculate the residuals) and a simpler search, albeit more robust, for the corrector step (which uses second-order approximation to calculate the residuals). Therefore, the Fibonacci search is used in the predictor step, since it is more accurate, and provides the minimum value for the objective function in the predefined direction; the Armijo search is used in the corrector step because it is simpler and more robust.

### 3.5. Stopping Rules

Interior point algorithms do not find exact solutions for linear or quadratic programming problems. Therefore, stopping rules are needed to decide when the solution obtained in a current iteration is sufficiently close to the optimal solution. In this study, the stopping rules are based on [32].

Many algorithms consider that a good approximate solution is the one that presents sufficiently small values for primal and dual residuals  $t^k, u^k$ , and also for the dual metric  $\mu_k$ . Nevertheless, it is possible to use relative values for the metrics  $t^k, u^k$ , and  $\mu_k$  in order to

reduce scaling problems, as described in [32]. Typical stopping rules are shown in (3.60)–(3.65), as follows:

(i) primal feasibility:

$$\frac{\|t^k\|}{\|b\| + 1} \leq \varepsilon_1 \quad (\text{predictor step}), \quad (3.60)$$

$$\frac{\|\tilde{t}^k\|}{\|b\| + 1} \leq \varepsilon_1 \quad (\text{corrector step}); \quad (3.61)$$

(ii) dual feasibility:

$$\frac{\|u^k\|}{\|Qx^k + c\| + 1} \leq \varepsilon_2 \quad (\text{predictor step}), \quad (3.62)$$

$$\frac{\|\tilde{u}^k\|}{\|Qx^k + c\| + 1} \leq \varepsilon_2 \quad (\text{corrector step}); \quad (3.63)$$

(iii) complementary slackness:

$$\|v^k\| \leq \varepsilon_3, \quad \|q^k\| \leq \varepsilon_3 \quad (\text{predictor step}) \quad (3.64)$$

$$\|\tilde{v}^k\| \leq \varepsilon_3, \quad \|\tilde{q}^k\| \leq \varepsilon_3 \quad (\text{corrector step}), \quad (3.65)$$

where  $\varepsilon_1$ ,  $\varepsilon_2$ , and  $\varepsilon_3$  are sufficiently small positive numbers. Eventually, other criteria can be adopted according to the specific characteristics of each problem, as can be seen in [24, 32].

### 3.6. Update of the Barrier Parameter

According to [32], the barrier parameter is updated using an inner product that involves the primal variables  $r^k$  and  $z^k$ , and the dual variables  $s^k$  and  $y^k$ , respectively, as shown in the following equation

$$\mu_k^1 = \frac{(r^k)^T s^k}{n}; \quad \mu_k^2 = \frac{(z^k)^T y^k}{n}, \quad (3.66)$$

so that  $\mu_k$  is calculated using the following equation

$$\mu_k = \text{Min}\left\{\sigma\mu_k^1, \sigma\mu_k^2\right\} \text{ for a constant } 0 < \sigma < 1, \quad (3.67)$$

where the parameter  $\sigma$  is used to accelerate the convergence of the iterative process. This procedure for updating the barrier parameter proposed by Wright in [32] helps in the theoretical convergence proof and also in the complexity analysis of primal-dual methods.

### 3.7. Algorithm for the Proposed IPLS Method

*Step 1* (initialisation). Adjust  $k = 0$ . Choose an arbitrary point:  $(x^0; w^0; z^0, y^0, r^0, y^0, s^0) \in \Omega^0$  and choose  $\varepsilon_1, \varepsilon_2$ , and  $\varepsilon_3$  as sufficiently small positive numbers.

*Step 2* (intermediate calculations—predictor). Calculate:  $t^k; g^k; f^k; o^k; q^k; v^k$  using (3.10),  $\mu_k$  using (3.67) and matrix  $\theta$  using (3.21).

*Step 3* (finding directions of translation—predictor). Calculate search directions  $d_x^k, d_w^k, d_z^k, d_r^k, d_y^k$ , and  $d_s^k$  for the predictor step using (3.17)–(3.24).

*Step 4* (computing step size—predictor). Calculate Fibonacci step sizes  $\alpha_{P_k}$  and  $\alpha_{D_k}$  using (3.41)–(3.44).

*Step 5* (moving to a new solution—predictor). Update  $x^{k+1}; w^{k+1}; s^{k+1}; y^{k+1}; z^{k+1}; r^{k+1}$  obtained from the predictor step, according to (3.35)–(3.44).

*Step 6* (checking for optimality—predictor). If the criteria defined in (3.60), (3.62), and (3.64) are satisfied, stop. The solution is optimal. Otherwise, go on to the following step.

*Step 7* (intermediate calculations—corrector). Calculate:  $\tilde{t}^k; \tilde{g}^k; \tilde{f}^k; \tilde{o}^k; \tilde{q}^k; \tilde{v}^k$  using (3.27),  $\mu_k$  using (3.67) and matrix  $\theta$  using (3.21).

*Step 8* (finding directions of translation—corrector). Calculate  $\tilde{d}_x^k, \tilde{d}_w^k, \tilde{d}_z^k, \tilde{d}_r^k, \tilde{d}_y^k$ , and  $\tilde{d}_s^k$ , for the corrector step using (3.28)–(3.33).

*Step 9* (computing step size—corrector). Calculate the Armijo step size  $\beta_{P_k}$  and  $\beta_{D_k}$  using (3.52)–(3.58).

*Step 10* (moving to a new solution—corrector). Update  $x^{k+1}; w^{k+1}; s^{k+1}; y^{k+1}; z^{k+1}; r^{k+1}$  using (3.46)–(3.48).

*Step 11* (checking for optimality—corrector). If the criteria defined in (3.61), (3.63), and (3.65) are satisfied, stop. The solution is optimal. Otherwise, go on to the next step.

*Step 12.* Adjust  $k = k + 1$  and return to Step 2.

The predictor Steps 2 through Step 6 are held in odd iterations, while the corrector Steps 7 through Step 11 are performed in even iterations. The next section describes numerical simulations involving the application of the proposed method to ED problems.

## 4. Application of the Proposed Algorithm to Solve ED Problems

In this section, the proposed method is applied to solve three power systems with 3, 6, and 13 generators, respectively. Tables 1, 3, and 5 show the data related to the generating units of the power systems. These data were extracted from [26, 27]. The tables list the coefficients of the cost function for each generating unit, as described in (2.1), and also minimum and maximum power output limits.

Tables 2 and 3 present the results of the application of the proposed method to solve the power system with 3 generators, while Tables 5 and 6 list the results for the system with

**Table 1:** Characteristics of the system with 3 generators.

Unit	$P_i^{\min}$ (MW)	$P_i^{\max}$ (MW)	$a_i$ (\$/MW <sup>2</sup> )	$b_i$ (\$/MW)	$c_i$ (\$)
1	100	600	0,001562	7,92	561
2	50	200	0,004820	7,97	78
3	100	400	0,001940	7,85	310

6 generators, and Tables 8 and 9 depict the results for the system with 13 generators. These tables compare the solutions obtained by the proposed IPLS method against those calculated by the following methods: the predictor-corrector primal-dual (PCPD) method described in [6], the hybrid genetic algorithm (HGA), the coevolutionary genetic algorithm (COEGA), the hybrid atavistic genetic algorithm (HAGA) and the cultural algorithm (CA). The solutions determined by the HGA and COEGA methods are available in [27], while the solutions obtained with the CA method are described in [26], and those obtained with HAGA method are given in [25], but only for the system comprising 13 generators.

The comparison between the IPLS method and the evolutionary approaches cited above were introduced in this section because these are traditional methods used in power system to solve ED problems, especially when a more general nondifferentiable objective function (as shown in (2.2)) is used. However, it is important to highlight that these evolutionary approaches are heuristic procedures, which provide only approximate solutions to ED problems. When ED is formulated as a quadratic problem, it can be solved by means of exact methods, such as the IPLS and PCPD method.

Therefore, to better evaluate the performance of proposed IPLS method, this method has been compared to the PCPD method in Table 10. The results in Table 10 have the main purpose to show the reduction in the computational effort when the IPLS method is compared to the PCPD method. Both methods were implemented using Borland Pascal 7.0 programming language.

#### 4.1. Power System with 3 Generators

The main characteristics of the system containing 3 generators are described in Table 1. Parameters  $a_i, b_i, c_i$  stand for the coefficients of the cost functions for the generators, while  $P_i^{\min}, P_i^{\max}$  represent minimum and maximum power output capabilities, respectively, for generating unit  $i$ .

The following values are adopted to initialise the method:

$$x^0 = (450, 100, 300); \quad w^0 = (0, 0, 0); \quad y^0 = (0, 0, 0). \quad (4.1)$$

The parameters related to the system's total demand are set at  $D = 850$  MW and the active power losses are neglected. The values adopted for  $\varepsilon_1, \varepsilon_2$ , and  $\varepsilon_3$  are  $10^{-8}$ .

Table 2 shows the active power output calculated by the methods. The results for the HAGA algorithm are not presented in this case study. Table 3 compares optimal values for the objective functions obtained by each method. From the results presented in this table, it is clear that the dispatches calculated by all the types of genetic algorithms cannot reach the global optimum dispatch attained by the interior point methods PCPD and IPLS. As Table 3 indicates, the cost calculated by the IPLS and PCPD methods is lower than that obtained by

**Table 2:** Comparison of the power generation outputs obtained for the system with 3 generators.

Power output	HGA	COEGA	PCPD	IPLS
$P_1$ (MW)	470,8421	344,7295	393,1698	393,1698
$P_2$ (MW)	109,4012	193,9445	122,2264	122,2264
$P_3$ (MW)	269,7567	311,3260	334,6038	334,6038
$\Sigma P_i$ (MW)	850,0000	850,0000	850,000	850,000

**Table 3:** Characteristics of the system with 3 generators.

Results	Generation costs (\$)
HGA	8.212,73
COEGA	8.223,86
PCPD	8.194,36
IPLS	8.194,36

**Table 4:** Characteristics of the system with 6 generators.

Unit	$P_i^{\text{Min}}$ (MW)	$P_i^{\text{Max}}$ (MW)	$a_i$ (\$/MW <sup>2</sup> )	$b_i$ (\$/MW)	$c_i$ (\$)
1	10	125	0,15247	38,53973	756,79886
2	10	150	0,10587	46,15916	451,3251
3	35	225	0,02803	40,39655	1049,9977
4	35	210	0,03546	38,30533	1243,5311
5	130	325	0,02111	36,32782	1658,5696
6	125	315	0,01799	38,24041	1356,6592

the HGA and COEGA methods. As already discussed, this is an expected result, since the evolutionary approaches provide only approximate solution to the problem.

#### 4.2. Power System with 6 Generators

Table 4 describes the main characteristics of the system containing 6 generators.

As in the previous case study, the parameters  $a_i, b_i, c_i$  stand for the coefficients of the cost functions for the generating unit  $i$ , while  $P_i^{\text{min}}, P_i^{\text{max}}$  represent minimum and maximum power output capabilities, respectively, for the generating unit  $i$ .

The following values are adopted to initialise the method:

$$x^0 = (20, 30, 75, 75, 145, 155); \quad w^0 = (0, 0, 0, 0, 0, 0); \quad y^0 = (1, 1, 1, 1, 1, 1). \quad (4.2)$$

The parameters related to the system's total demand are set at  $D = 500$  MW and the active power losses are neglected. The values adopted for  $\varepsilon_1, \varepsilon_2$ , and  $\varepsilon_3$  are  $10^{-8}$ .

Table 5 shows the active power dispatch calculated by the methods. The solutions for COEGA and HAGA algorithms were not presented by their authors. Table 6 compares optimal values for the objective function obtained by each method. Again, as expected, the costs calculated by the IPLS and PCPD methods are lower than those obtained by the HGA method.

**Table 5:** Comparison of the power generation outputs obtained for the system with 6 generators.

Generated power	HGA	PCPD	IPLS
$P_1$ (MW)	20,1367	17,36596	17,36596
$P_2$ (MW)	14,8645	10,00000	10,00000
$P_3$ (MW)	72,4008	61,340667	61,340667
$P_4$ (MW)	72,4008	77,97487	77,97487
$P_5$ (MW)	180,0617	177,81828	177,81828
$P_6$ (MW)	139,5865	155,500216	155,500216
$\sum P_i$ (MW)	500,0000	500,0000	500,0000

**Table 6:** Values of the objective function for the system with 6 generators.

Results	Generation cost (\$)
HGA	27.037,29
PCPD	27.003,50
IPLS	27.003,50

**Table 7:** Characteristics of the system with 13 generators.

Unit	$P_i^{\text{Min}}$ (MW)	$P_i^{\text{Max}}$ (MW)	$a_i$ (\$/MW <sup>2</sup> )	$b_i$ (\$/MW)	$c_i$ (\$)
1	0	680	0,00028	8,1	550
2	0	360	0,00056	8,1	309
3	0	360	0,00056	8,1	307
4	60	180	0,00324	7,74	240
5	60	180	0,00324	7,74	240
6	60	180	0,00324	7,74	240
7	60	180	0,00324	7,74	240
8	60	180	0,00324	7,74	240
9	60	180	0,00324	7,74	240
10	40	120	0,00284	8,6	126
11	40	120	0,00284	8,6	126
12	55	120	0,00284	8,6	126
13	55	120	0,00284	8,6	126

#### 4.3. Power System with 13 Generators

Table 7 describes the main characteristics of the system containing 13 generators.

The parameters in this table are analogous to those described in the preceding case studies. The following values are adopted to initialise the method:

$$\begin{aligned} x^0 &= (660, 320, 330, 150, 140, 155, 165, 150, 140, 70, 80, 75, 85); \\ w^0 &= (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); \\ y^0 &= (30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30). \end{aligned} \quad (4.3)$$

The parameters related to the system's total demand are set at  $D = 2520$  MW and the active power losses are neglected. The values adopted for  $\varepsilon_1$ ,  $\varepsilon_2$ , and  $\varepsilon_3$  are  $10^{-8}$ .

**Table 8:** Comparison of the power generation outputs obtained for the system with 13 generators.

Power output	HGA	COEGA	CA	PCPD	IPLS
$P_1$	651,145	735,626	679,2551	680,0000	680,0000
$P_2$	319,982	337,495	359,8672	360,0000	360,0000
$P_3$	320,463	292,625	357,2368	360,0000	360,0000
$P_4$	137,776	146,713	154,8137	155,0000	155,0000
$P_5$	156,688	177,346	158,0946	155,0000	155,0000
$P_6$	147,007	131,552	155,8520	155,0000	155,0000
$P_7$	159,165	154,197	146,1697	155,0000	155,0000
$P_8$	145,378	159,550	146,8364	155,0000	155,0000
$P_9$	151,551	167,339	168,7979	155,0000	155,0000
$P_{10}$	82,259	60,677	40,0181	40,0000	40,0000
$P_{11}$	86,320	74,681	40,0000	40,0000	40,0000
$P_{12}$	82,893	56,537	55,0175	55,0000	55,0000
$P_{13}$	79,368	25,655	55,0488	55,0000	55,0000
$\sum P_i$ (MW)	2,520,00	2,520,00	2,520,00	2,520,00	2,520,00

**Table 9:** Values of the objective function for the system with 13 generators.

Results	Generation cost (\$)
HGA	24.111,69
COEGA	24.072,03
HAGA	24.052,34
CA	24.052,10
PCPD	24.050,08
IPLS	24.050,08

Table 8 lists the power generation outputs obtained by the methods. The values of the dispatch calculated by the HAGA are not given by [25], who provided only the optimal value for the objective function. Once more, as Table 9 indicates, the total cost calculated by the PCPD and IPLS methods is lower than that calculated by all the others, although CA and HAGA approaches get close to the optimal solution point.

#### 4.4. Performance of the Interior Point Methods Tested

This section evaluates the computational effort of the interior point methods tested here (PCPD and IPLS), measuring it in terms of the number of iterations required to obtain the optimal solution. To this end, Table 10 shows the number of iterations obtained by the solution algorithms of each method for the previously studied systems. It is important to highlight that the same parameter settings were used for the two methods.

Computational tests were carried out in an Intel Corel Quad Q9550, with 3.5 GB of RAM memory, in order to calculate and compare the CPU times between the proposed IPLS method and the PCPD method. For such a purpose, we utilized the specific unit GET TIME from Borland Pascal 7.0. The precision of this unit is up to milliseconds. The computational times calculated by this unit for the power systems tested (which include the systems with 3, 6, and 13 generators) were all null for both, the proposed method, and the PCPD method. This occurred due to the efficiency of the machine processor and also due to the efficiency of

**Table 10:** Number of iterations for the PCPD and IPLS methods.

Simulated system	PCPD	IPLS
3 generating units	26	12
6 generating units	22	15
13 generating units	23	17

both methods tested. Therefore, the efficiency of the methods is compared only in terms of number of iterations, as described in Table 10.

The basic difference between the PCPD and IPLS algorithms is that line searches are incorporated in the corrector and predictor steps of the IPLS method. Therefore, these results demonstrate that the introduction of line searches in the IPLS method greatly reduces the number of iterations required to solve ED problems. Since line searches represent only a minor additional computational effort in the overall process of finding a solution, the computational effort for solving ED problems is significantly improved by the proposed IPLS method.

#### 4.5. Postoptimization Analysis: Incremental Costs

This section conducts a postoptimization analysis to evaluate the optimality conditions of the solutions obtained in the previous section for the systems in question. The purpose of this analysis is to evaluate the complete picture concerning optimal solutions for ED problems. The classical ED problem (2.1) can be rewritten as follows:

$$\begin{aligned} & \text{Minimize} && C_T \\ & \text{subject to} && \sum_{i=1}^n P_i = D \\ & && P_i \in S, \end{aligned} \tag{4.4}$$

where  $S = \{P_i \in \mathbb{R} \mid P_i - P_i^{\min} \geq 0 \text{ and } P_i^{\max} - P_i \geq 0, i = 1 \dots, n\}$ , and  $C_T = \sum_{i=1}^n (C_i(P_i) = a_i P_i^2 + b_i P_i + c_i)$ .

Problem (4.4) is subsequently used for the postoptimization analysis of the solutions obtained by the systems studied in the previous section.

#### 4.6. The Lagrangian Function and KKT Conditions

Considering ED as redefined in (4.4), one finds the following associated Lagrangian function:

$$L(P_i, w, s_i, y_i) = C_T(P_i) + w \left( D - \sum_{i=1}^n P_i \right) + \sum_{i=1}^n s_i (P_i^{\min} - P_i) + \sum_{i=1}^n y_i (P_i - P_i^{\max}), \tag{4.5}$$

where  $w \in \Re$ ,  $s_i \in \Re$ , and  $y_i \in \Re$ ,  $i = 1, \dots, n$ , are the associated Lagrange multipliers. The KKT conditions associated with the Lagrangian function (4.5) are expressed through

$$D - \sum_{i=1}^n P_i = 0, \quad (4.6)$$

$$P_i - P_i^{\min} \geq 0; \quad s_i(P_i - P_i^{\min}) = 0; \quad i = 1, \dots, n, \quad (4.7)$$

$$P_i^{\max} - P_i \geq 0; \quad y_i(P_i^{\max} - P_i) = 0; \quad i = 1, \dots, n, \quad (4.8)$$

$$\lambda_i(P_i) - w - s_i + y_i = 0; \quad i = 1, \dots, n, \quad (4.9)$$

where

$$\lambda_i(P_i) = \frac{\partial C_i(P_i)}{\partial P_i} = 2a_i P_i + b_i; \quad i = 1, \dots, n. \quad (4.10)$$

In the literature on power systems, the Lagrange multiplier  $w$  is called energy price (i.e., the price of 1 MWh), while  $\lambda_i(P_i)$  is called incremental or marginal cost of generating unit  $i$ . Using expression (4.9), the energy price can be calculated by

$$w = \lambda_i(P_i) - s_i + y_i; \quad i = 1, \dots, n. \quad (4.11)$$

#### 4.7. Energy Prices and Incremental Costs Using KKT Conditions

If the optimal solution provided by the IPLS method satisfies the KKT conditions, then there are four possible combinations for analysing prices and incremental costs, which are examined below.

##### 4.7.1. No Active Constraint

In this case there is no active constraint in the optimal solution of problem (4.4), so that  $s_i^* = 0$ ,  $y_i^* = 0$ ;  $i = 1, \dots, n$ . Thus, from (4.11), one reaches

$$w = \lambda_i^*(P_i^*); \quad i = 1, \dots, n, \quad (4.12)$$

that is, the energy price is equal to the incremental (marginal) costs for all the generating units. This situation corresponds to the rule commonly used by power system operators, which states that all marginal costs are equal. It is important to emphasize that this rule is valid only for this case and should, therefore, be used cautiously.

##### 4.7.2. Active Constraints for Maximum Power Output Limit

In this case, some generating units are dispatched in their maximum power output capabilities, so that the constraints associated with maximum power output capabilities are

active in the optimal solution of (4.4). To analyse this case, let the set  $\Omega_{\max}$  be defined with the indices of the generating units that have been optimally dispatched in their maximum power output. In this situation,  $s_i^* = 0$  for  $i = 1, \dots, n$ ,  $y_j^* > 0$  for  $j \in \Omega_{\max}$ , and  $y_j^* = 0$  for  $j \notin \Omega_{\max}$ . Starting from (4.11), one reaches

$$\begin{aligned} w &= \lambda_j^*(P_j^*) + y_j^*; \quad j \in \Omega_{\max}, \\ w &= \lambda_j^*(P_j^*); \quad j \notin \Omega_{\max}. \end{aligned} \tag{4.13}$$

Based on (4.13), it is possible to show that the marginal costs for the group of generating units that belong to  $\Omega_{\max}$  are lower than the marginal costs for the group that does not belong to  $\Omega_{\max}$ , since  $y_j^* > 0$ ;  $j \in \Omega_{\max}$ .

#### 4.7.3. Active Constraints for Minimum Power Output Limit

In this case, some generating units are dispatched in their minimum power output capabilities, so that the constraints associated with minimum power output capabilities are active in the optimal solution of (4.4). To analyse this case, let the set  $\Omega_{\min}$  be defined with the indices of the generating units that have been optimally dispatched in their minimum power output. In this situation,  $y_i^* = 0$  for  $i = 1, \dots, n$ ,  $s_j^* > 0$  for  $j \in \Omega_{\min}$ , and  $s_j^* = 0$  for  $j \notin \Omega_{\min}$ . Thus, using (4.11), one reaches

$$\begin{aligned} w &= \lambda_j^*(P_j^*) - s_j^*; \quad j \in \Omega_{\min}, \\ w &= \lambda_j^*(P_j^*); \quad j \notin \Omega_{\min}. \end{aligned} \tag{4.14}$$

Based on (4.14), it can be shown that the marginal costs for the group of generating units that belong to  $\Omega_{\min}$  are higher than the marginal costs for the group that does not belong to  $\Omega_{\min}$ , since  $s_j^* > 0$ ;  $j \in \Omega_{\min}$ .

#### 4.7.4. Active Constraints for Both Minimum and Maximum Power Output Capabilities

In this case, some generating units are dispatched in their upper limits, while others are dispatched in their lower limits, or at some other operational point between their upper and lower limits. Obviously, the same unit cannot simultaneously assume lower and upper limits. The comments made in the preceding sections concerning marginal costs for the generating units that are dispatched in their lower or upper limits also apply to this case.

### 4.8. Results of Incremental Analyses for the Systems under Study

#### 4.8.1. Case 1: Power System with 3 Generating Units

Table 2 describes the optimal dispatch,  $P_i^*$ ,  $i = 1, \dots, n$ , calculated by the IPLS method for the system with 3 generating units. As this table indicates, no generating unit has reached its

**Table 11:** Incremental analysis for the 3-generator system.

Unit	$\lambda_i^*$	$s_i^*$	$y_i^*$	$w^*$
1	9,14826	0	0	9,14826
2	9,14826	0	0	
3	9,14826	0	0	

**Table 12:** Incremental analysis of the 6-generator system.

Unit	$\lambda_i^*$	$s_i^*$	$y_i^*$	$w$
1	43,8449	0	0	43,8449
2	48,2765	4,43164	0	
3	43,8449	0	0	
4	43,8449	0	0	
5	43,8449	0	0	
6	43,8449	0	0	

lower or upper limit, so there is no active constraint in the optimal solution. This situation coincides with the one described in Section 4.7.1. The energy price, marginal costs, and Lagrange multipliers are shown in Table 11 for this case. As described in Section 4.7.1, the values obtained by the IPLS method for  $s_i^*$  and  $y_i^*$ ;  $i = 1, \dots, n$  are all zero. Also, the energy price is equivalent to the marginal costs, which are all equal, as expected. These results are in accordance with the theory described in Section 4.7.1.

#### 4.8.2. Case 2: Power System with 6 Generating Units

Table 5 shows the optimal dispatch,  $P_i^*$ ,  $i = 1, \dots, n$ , calculated by the IPLS method for the system with 6 generating units. As can be seen in this table, the generating unit 2 has reached its lower limit, so the optimal solution has one active constraint. This situation coincides with the one described in Section 4.7.3. Table 12 lists the energy price, marginal costs, and Lagrange multipliers calculated by the IPLS method for this case. As described in Section 4.7.3, the values obtained by the IPLS method for  $s_i^*$  and  $y_i^*$ ;  $i = 1, \dots, n$  are all zero except for the value of  $s_2^*$ , which is associated with the constraint for lower output capabilities at unit 2. Except for unit 2, all the marginal costs are equal to the energy price. These results are in accordance with the theory described in Section 4.7.3 and also with (4.14).

#### 4.8.3. Case 3: Power System with 13 Generating Units

Table 8 shows the optimal dispatch,  $P_i^*$ ,  $i = 1, \dots, n$ , calculated by the IPLS method for the system with 13 generating units. Table 13 lists the energy price, marginal costs, and Lagrange multipliers calculated by the IPLS method. As the latter table indicates, generated units 1, 2, and 3 have reached their upper limits, and generated units 10, 11, 12, and 13 have reached their lower limits, indicating that there are 6 active constraints in the optimal solution calculated by the IPLS method. This situation coincides with the one described in Section 4.7.4. As described previously, the values obtained by the IPLS method for  $s_i^*$  and  $y_i^*$ ;  $i = 1, \dots, n$  are zero except for the values of  $y_1^*$ ,  $y_2^*$ ,  $y_3^*$ , which are associated with upper output capabilities, and also for  $s_{10}^*$ ,  $s_{11}^*$ ,  $s_{12}^*$ ,  $s_{13}^*$ , which are associated with lower output

**Table 13:** Incremental analysis for the 13-generator system.

Unit	$\lambda_i^*$	$s_i^*$	$y_i^*$	$w$
1	8,4807	0	0,2635	8,7443
2	8,5031	0	0,2411	
3	8,5031	0	0,2411	
4	8,7443	0	0	
5	8,7443	0	0	
6	8,7443	0	0	
7	8,7443	0	0	
8	8,7443	0	0	
9	8,7443	0	0	
10	8,8272	0,0828	0	
11	8,8272	0,0828	0	
12	8,9124	0,1680	0	
13	8,9124	0,1680	0	

capabilities. The marginal costs for all the remaining units (4, 5, 6, 7, 8, and 9) are equal to the energy price. These results are in accordance with the theory described in Sections 4.7.2 and 4.7.3 and also with (4.13) and (4.14).

As expected, in all the cases analysed in Tables 11, 12, and 13, the value of the generation cost  $w = \lambda_i^* - s_i^* + y_i^*$  was determined univocally, which proves the equal incremental cost criterion [29].

The analysed results indicate that the dispatches  $P_i^*$ ,  $i = 1, \dots, n$  calculated, respectively, in Tables 2, 5, and 8 and the results for  $\lambda_i^*$ ,  $w$ ,  $s_i^*$ , and  $y_i^*$  determined, respectively, in Tables 11, 12, and 13, satisfy the complementary slackness conditions set forth in (4.7) and (4.8).

## 5. Conclusions

This paper proposes a predictor-corrector primal-dual interior point method which introduces line search procedures (IPLS) in both the predictor and corrector steps. The Fibonacci search is used in the predictor step, while an Armijo search is used in the corrector step. The method is developed for application to the economic dispatch (ED) problem, which is an example of a quadratic programming problem studied in the field of power systems analysis. ED problems have already been solved through primal-dual interior point methods, although the strategy adopted here to solve the problem has not yet been tested.

The proposed algorithm is applied to solve ED problems in case studies involving systems with 3, 6, and 13 generating units. The results provided by the proposed IPLS method are compared to those provided by several other methods described in the literature, such as the predictor-corrector primal-dual (PCPD) interior point method, hybrid genetic algorithm, coevolutionary genetic algorithm, hybrid atavistic genetic algorithm, and cultural algorithm. The dispatches calculated by all the types of genetic algorithms could not reach the global optimal dispatch attained by the interior point methods PCPD and IPLS.

The computational effort of the interior point methods tested (PCPD and IPLS) was evaluated and measured in terms of the number of iterations required to find the optimal solution. The results indicate that the introduction of line searches in the IPLS method

considerably reduces the number of iterations required for solving ED problems. Line searches pose represent only a minor additional computational effort in the overall process of finding a solution; hence, the computational effort for solving ED problems is also greatly improved by the proposed IPLS method.

A theory for performing a postoptimization analysis was also analysed. This theory highlights the relation among energy prices, incremental generation costs, and other Lagrange multipliers. This mathematical relation is used to validate optimal solutions for ED problems calculated by the IPLS method.

Further research could involve the representation of the so-called valve point loading in the objective function of ED problems. In that case, the nondifferentiability of the objective function should be treated appropriately.

## Acknowledgment

The authors are grateful for the financial support of FAPESP and CNPq (Brazil).

## References

- [1] N. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica*, vol. 4, no. 4, pp. 373–395, 1984.
- [2] I. I. Dikin, "Iterative solution of problems of linear and quadratic programming," *Doklady Akademii Nauk SSSR*, vol. 174, pp. 747–748, 1967 (Russian).
- [3] E. R. Barnes, "A variation on Karmarkar's algorithm for solving linear programming problems," *Mathematical Programming*, vol. 36, no. 2, pp. 174–182, 1986.
- [4] R. J. Vanderbei, M. S. Meketon, and B. A. Freedman, "A modification of Karmarkar's linear programming algorithm," *Algorithmica*, vol. 1, no. 4, pp. 395–407, 1986.
- [5] I. Adler, M. G. C. Resende, G. Veiga, and N. Karmarkar, "An implementation of Karmarkar's algorithm for linear programming," *Mathematical Programming*, vol. 44, no. 1–3, pp. 297–335, 1989.
- [6] A. R. Balbo, M. A. S. Souza, and E. C. Baptista, "Métodos primal-dual de pontos interiores aplicados à resolução de problemas de despacho econômico: sobre a influência da solução inicial," in *Proceedings of the Simpósio Brasileiro de Pesquisa Operacional, João Pessoa—Pb. Anais do XL SBPO (XL SBPO '08)*, pp. 2074–2085, ILTC, Rio de Janeiro, RJ, Brazil, 2008.
- [7] N. Megiddo and M. Shub, "Boundary behavior of interior point algorithms in linear programming," *Mathematics of Operations Research*, vol. 14, no. 1, pp. 97–146, 1989.
- [8] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley & Sons, New York, NY, USA, 1968.
- [9] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming*, vol. 4 of *Classics in Applied Mathematics*, SIAM, Philadelphia, Pa, USA, 2nd edition, 1990.
- [10] K. R. Frisch, *The Logarithmic Potential Method of Convex Programming*, University Institute of Economics, Oslo, Norway, 1955.
- [11] P. E. Gill, W. Murray, M. A. Saunders, J. A. Tomlin, and M. H. Wright, "On projected Newton barrier methods for linear programming and an equivalence to Karmarkar's projective method," *Mathematical Programming*, vol. 36, no. 2, pp. 183–209, 1986.
- [12] Y. Ye, "An  $O(n^3L)$  potential reduction algorithm for linear programming," in *Contemporary Mathematics*, vol. 114, pp. 91–107, American Mathematical Society, Providence, RI, USA, 1990.
- [13] J. Renegar, "A polynomial-time algorithm, based on Newton's method, for linear programming," *Mathematical Programming*, vol. 40, no. 1, pp. 59–93, 1988.
- [14] P. M. Vaidya, "An algorithm for linear programming which requires  $O(((m + n)n^2 + (m + n)^{1.5}n)L)$  arithmetic operations," *Mathematical Programming*, vol. 47, no. 2, pp. 175–201, 1990.
- [15] N. Megiddo, "On the complexity of linear programming," in *Advances in Economic Theory*, T. Bewley, Ed., pp. 225–268, Cambridge University Press, Cambridge, UK, 1989.

- [16] C. C. Gonzaga, "An algorithm for solving linear programming problems in  $O(n^3L)$  operations," in *Progress in Mathematical Programming: Interior-Point and Related Methods*, N. Megiddo, Ed., pp. 1–28, Springer, New York, NY, USA, 1989.
- [17] C. C. Gonzaga, "Polynomial affine algorithms for linear programming," *Mathematical Programming*, vol. 49, no. 1, pp. 7–21, 1990.
- [18] R. D. C. Monteiro and I. Adler, "Interior path following primal-dual algorithms. I. Linear programming," *Mathematical Programming*, vol. 44, no. 1, pp. 27–41, 1989.
- [19] R. D. C. Monteiro, I. Adler, and M. G. C. Resende, "A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extension," *Mathematics of Operations Research*, vol. 15, no. 2, pp. 191–214, 1990.
- [20] M. Kojima, S. Mizuno, and A. Yoshise, "A primal-dual interior point algorithm for linear programming," in *Progress in mathematical programming: Interior-Point and Related Methods*, N. Megiddo, Ed., pp. 29–47, Springer, New York, NY, USA, 1989.
- [21] S. Mehrotra and J. Sun, "An algorithm for convex quadratic programming that requires  $O(n^{3.5}L)$  arithmetic operations," *Mathematics of Operations Research*, vol. 15, no. 2, pp. 342–363, 1990.
- [22] I. J. Lustig, R. E. Marsten, and D. F. Shanno, "On implementing Mehrotra's predictor-corrector interior-point method for linear programming," *SIAM Journal on Optimization*, vol. 2, no. 3, pp. 435–449, 1992.
- [23] Y. C. Wu, A. S. Debs, and R. E. Marsten, "Direct nonlinear predictor-corrector primal-dual interior point algorithm for optimal power flows," *IEEE Transactions on Power Systems*, vol. 9, no. 2, pp. 876–883, 1994.
- [24] S. C. Fang and S. Puthenpura, *Linear Optimization and Extensions: Theory and Algorithms*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.
- [25] J. O. Kim, D. J. Shin, J. N. Park, and C. Singh, "Atavistic genetic algorithm for economic dispatch with valve point effect," *Electric Power Systems Research*, vol. 62, no. 3, pp. 201–207, 2002.
- [26] N. M. Rodrigues, *Um algoritmo cultural para problemas de despacho de energia elétrica*, Dissertação de Mestrado, Universidade Estadual de Maringá, 2007.
- [27] M. M. A. Samed, *Um algoritmo genético Hibrido co-evolutivo para resolver problemas de despacho*, Tese de Doutorado, UEM, Departamento de Engenharia Química, 2004.
- [28] H. H. Happ, "Optimal power dispatch—a comprehensive survey," *IEEE Transactions on Power Apparatus and Systems*, vol. 96, no. 3, pp. 841–854, 1977.
- [29] M. J. C. Steinberg and T. H. Smith, *Economic Loading of Power Plants and Electric Systems*, MacGraw-Hill, 1943.
- [30] A. R. L. Oliveira, S. Soares, and L. Nepomuceno, "Optimal active power dispatch combining network flow and interior point approaches," *IEEE Transactions on Power Systems*, vol. 18, no. 4, pp. 1235–1240, 2003.
- [31] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Non Linear Programming-Theory and Algorithm*, John Wiley & Sons, New York, NY, USA, 2nd edition, 1993.
- [32] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, Pa, USA, 1997.

*Research Article*

## **Applying Hierarchical Bayesian Neural Network in Failure Time Prediction**

**Ling-Jing Kao<sup>1</sup> and Hsin-Fen Chen<sup>2</sup>**

<sup>1</sup> Department of Business Management, National Taipei University of Technology, 10608, Taiwan

<sup>2</sup> Graduate Institute of Industrial and Business Management, National Taipei University of Technology, 10608, Taiwan

Correspondence should be addressed to Ling-Jing Kao, lingjingkao@ntut.edu.tw

Received 31 December 2011; Accepted 21 February 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 L.-J. Kao and H.-F. Chen. 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.

With the rapid technology development and improvement, the product failure time prediction becomes an even harder task because only few failures in the product life tests are recorded. The classical statistical model relies on the asymptotic theory and cannot guarantee that the estimator has the finite sample property. To solve this problem, we apply the hierarchical Bayesian neural network (HBNN) approach to predict the failure time and utilize the Gibbs sampler of Markov chain Monte Carlo (MCMC) to estimate model parameters. In this proposed method, the hierarchical structure is specified to study the heterogeneity among products. Engineers can use the heterogeneity estimates to identify the causes of the quality differences and further enhance the product quality. In order to demonstrate the effectiveness of the proposed hierarchical Bayesian neural network model, the prediction performance of the proposed model is evaluated using multiple performance measurement criteria. Sensitivity analysis of the proposed model is also conducted using different number of hidden nodes and training sample sizes. The result shows that HBNN can provide not only the predictive distribution but also the heterogeneous parameter estimates for each path.

### **1. Introduction**

In this high technology era, the society operations highly depend on various machinery and equipments. Once the machinery or equipment is broken down, enormous trouble and economics cost will be brought to the entire society. To enhance the product reliability, the methodologies to assess product reliability have received much discussion in both academics and industries. Among several mature techniques, degradation testing provides an efficient way for reliability assessment when product quality is associated with a time-varying degradation process. Typically, degradation measures can provide more reliability information, particularly in modeling the failure-causing mechanism, than time-to-failure data in few or no failure situation.

Predicting the remaining lifetime of a product is also an important issue in quality control. For example, knowing the remaining equipment lifetime can help in optimizing the machine maintenance schedule. The equipment lifetime is traditionally studied by fitting a statistical probability distribution, and most of these statistical models are constructed to study various degradation processes of a product. Examples include Lu and Meeker [1], Lu et al. [2], and Meeker and Hamada [3]. The method of stochastic processes is the alternative used to study the degradation data. Examples can be found in Doksum and Hoyland [4].

Most of above methods emphasize on parameter estimations or the process of hypothesis testing. Under the assumption that data follows a certain probability distribution, the statistical inference is made based on the asymptotic theory. The statistical inferences based on the asymptotic theory are valid only if the sample size is large or close to infinite. When the sample information is small or when the discrete data are provided, the finite sample property of the estimation based on the asymptotic theory is not held. Therefore, nonparametric or semiparametric statistics have been proposed to perform the reliability prediction. However, these statistical methods are far from perfect because the overfitting problem usually leads to inaccurate parameter estimates.

Due to the data limitations and the drawbacks of classical statistics approaches, Bayesian approach provides the solution from a different perspective. Unlike these frequentist's approaches which consider the data random and the test statistics or estimators are investigated over imaginary samples  $f(y | \theta)$ , Bayesian approach regards the sampling distribution irrelevant to the statistical inferences because it considers events that have not occurred yet. Bayesian inference is conducted using Bayes theorem in which posterior distribution is defined by the likelihood function which contains sample information times the prior distribution of parameter of the interest. Since Bayesian inference follows the formal rules of probability theory, Bayes estimators are consistent, asymptotically efficient, and admissible under mild conditions. The detail discussion of Bayesian approach can be found in Bernardo and Smith [5], Gelman et al. [6], Robert and Casella [7], and Liu [8].

Lately Bayesian has been applied in the fatigue crack growth prediction in the literature. For example, Zheng and Ellingwood [9] generalize a stochastic fatigue crack growth model by incorporating a time-dependent noise term described by arbitrary marginal distributions and autocorrelations to model the uncertainty in the crack growth under constant amplitude loading. Zhang and Mahadevan [10] propose a Bayesian procedure to quantify the uncertainty in mechanical and statistical model selection and the uncertainty in distribution parameters. The procedure is applied to a fatigue reliability problem, with the combination of two competing crack growth models and considering the uncertainty in the statistical distribution parameters for each model. Akama [11] performs the Bayesian analysis to estimate an appropriate value of the uncertain propagation rate of cracks that can be initiated at the wheel seat of a Shinkansen vehicle axle.

Neural network (NN) is the other popular prediction method. Neural network is a computer-intensive, algorithmic procedure for transforming inputs into desired outputs using highly inter-connected networks of relatively simple processing elements (often termed neurons, units, or nodes; we will use nodes thereafter). Neural networks are modeled following the neural activity in the human brain. The essential features of a neural network are the nodes, the network architecture describing the connections between the nodes, and the training algorithm used to find values of the network parameters (weights) for a particular network. The nodes are connected to one another in the sense that the output from one node can be served as the inputs to other nodes. Each node transforms an input to an output using some specified function that is typically monotone, but otherwise arbitrary. This function

depends on parameters whose values must be determined with a training set of inputs and outputs. Network architecture is the organization of nodes and the types of connections permitted. The nodes are arranged in a series of layers with connections between nodes in different layers, but not between nodes in the same layer.

Several researchers also integrate neural network algorithm with Bayesian theory, which has been known as Bayesian neural network, in prediction. For examples, Neal [12] applied Hybrid Markov chain Monte Carlo (MCMC) numerical integration techniques for the implementation of Bayesian procedures. Müller and Rios Insua [13] proposed a MCMC scheme based on a static or dynamic number of hidden nodes. In their subsequent paper, they have extended their research results by releasing the constraint of number of hidden nodes [13]. Also, Holmes and Mallick [14] used Bayesian neural network modeling in the regression context.

In this paper, we conduct a hierarchical Bayesian neural network analysis with MCMC estimation procedure in the failure time prediction problem. Here, hierarchy means that the coefficients in our constructed HBNN model are specified by random effect distributions. We attempt to use this hierarchical structure to determine if the heterogeneity exists among paths. The advantage of proposed HBNN model cannot only provide a better failure time prediction by incorporating the heterogeneity of components and autocorrelated structure of error term but also provide a predictive distribution for the target value. Different from previous research, the proposed HBNN model can successfully offer the full information of parameter estimation and covariance structure. Engineers can use the heterogeneity estimates to identify the causes of the quality differences and further enhance the product quality.

The data of the fatigue crack growth from Bogdanoff and Kozin [15] is used to illustrate the proposed model. To demonstrate the effectiveness of the proposed model, the prediction performance of the proposed model is evaluated using multiple performance measurement criteria. Sensitivity evaluation of the proposed model is also conducted using different number of hidden nodes and training sample sizes. The result shows that HBNN can provide not only the predictive distribution but also the heterogeneous parameter estimates for each path.

The rest of this paper is organized as follows: Section 2 introduces the proposed HBNN model for failure time prediction. In Section 3, the fatigue crack growth data from Bogdanoff and Kozin [15] is illustrated, and the model estimation procedure is provided. Failure time prediction and sensitivity analysis are demonstrated in Section 4. Concluding remarks are offered in Section 5.

## 2. HBNN Model for Failure Time Prediction

To model failure time, we adapted the growth-curve equation used by Liski and Nummi [16] as follows:

$$t_{i,j} = \beta_{i,m=0} + \sum_{m=1}^M \beta_{i,m} \Gamma_{i,m,j} + \varepsilon_{i,j}, \quad (2.1)$$

$$\Gamma_{i,m,j} = \psi(\varpi_{i,k=0,m} + \varpi_{i,k=1,m} y_{i,j} + \varpi_{i,k=2,m} \ln(y_{i,j})),$$

where  $y_{i,j}$  is the  $j$ th crack length of the  $i$ th path and  $t_{i,j}$  is the observed cycle time of the  $i$ th path, where  $i = 1, 2, \dots, N$  and  $j = 1, 2, \dots, n_i$ . In addition,  $\beta_{i,m}$  are the weights of the  $i$ th path

attached to the hidden nodes  $m$  ( $m = 1, 2, \dots, M$ ),  $M$  is the number of hidden nodes,  $\Gamma_{i,m,j}$  is the output of the  $m$ th hidden node when the  $j$ th crack length of the  $i$ th path is presented,  $\varpi_{i,k=1,m}$  are the weights from the first input,  $y_{i,j}$ , to the hidden node  $m$ , and  $\psi$  is the activation function. Typically, the choice of  $M$  depends upon the problem under consideration. The testing results of neural network with combinations of different numbers of hidden nodes have been investigated. In the present case, we have set the number of hidden nodes  $M$  equal to 3 because it gives the best predicting result.

According to the above equation, we know that there are totally  $N$  paths from a given population, and  $n_i$  observations are available for path  $i$  at fixed crack lengths  $y_{i,1}, y_{i,2}, \dots, y_{i,n_i}$  (i.e., The observations at length  $y_{i,1}, y_{i,2}, \dots, y_{i,n_i}$  are  $t_{i,1}, t_{i,2}, \dots, t_{i,n_i}$ , resp.). Herein, we assume that the conditional distribution of  $t_{i,j}$  given  $y_{i,j}$  is normally distributed as  $f(t_{i,j} | y_{i,j}) \sim N(\beta_{i,0} + \sum_{m=1}^M \beta_{i,m} \Gamma_{i,m,j}, \sigma^2)$ . It means that each value of  $y_{i,j}$  produces a random value of  $t_{i,j}$  from a normal distribution with a mean of  $\beta_{i,0} + \sum_{m=1}^M \beta_{i,m} \Gamma_{i,m,j}$  and a variance  $\sigma^2$ . Moreover, from literature [17, 18], we understand that degradation signals are usually autocorrelated in nature. We also noticed that the values of the first-order autocorrelation  $r_1$  of the residuals in Lu and Meeker [1] are not exactly equal to 2.0. Therefore, we suspected that the error term might be characterized as a first-order autoregressive process. Based on this finding, we proposed a new parametric crack growth model with autocorrelated errors as the following equations:

$$t_{i,j} = \beta_{i,0} + \sum_{m=1}^M \beta_{i,m} \Gamma_{i,m,j} + \varepsilon_{i,j}, \quad (2.2)$$

$$\Gamma_{i,m,j} = \psi(\varpi_{i,0,m} + \varpi_{i,1,m} y_{i,j} + \varpi_{i,2,m} \ln(y_{i,j})), \quad (2.3)$$

$$\varepsilon_{i,j} = \rho_i \varepsilon_{i,j-1} + Z_{i,j}, \quad (2.4)$$

where  $\rho_i$  is the autocorrelation coefficient and  $Z_{i,j}$  is a normal distributed error with  $N(0, \sigma^2)$  form. Note that the elements  $t_{i,1}, t_{i,2}, \dots, t_{i,n_i}$  in (2.2) are independent given  $\beta_{i,m}, \varpi_{i,k,m}, \sigma^2, \rho_i$  and  $y_{i,j}$ , where  $k$  is the number of inputs. The function  $\psi(\cdot)$  is referred to as an activation function in a neural network. Typically, the activation function is nonlinear. Some of the most common choices of  $\psi(\cdot)$  are the logistic and the hyperbolic tangent functions. In order to describe the heterogeneity varying from path to path, we characterized  $\beta_i$  by a 4-variate normal distribution with mean vector  $\bar{\beta}$  and covariance matrix  $V_\beta$ ,  $\beta_i | \bar{\beta}, V_\beta \sim N_4(\bar{\beta}, V_\beta)$ ,  $i = 1, 2, \dots, N$ , and  $\varpi_{i,m}$  is characterized by a 3-variate normal distribution with mean vector  $\bar{\varpi}_m$ , and covariance matrix  $V_{\varpi_m}$  for  $m = 1, 2$ , and 3. Equations (2.2)–(2.4) specify a general model for studying when observed cycle time sensitivity to crack length may increase. The heterogeneity among paths is captured by parameters  $\beta_i, \varpi_{i,k,m}$ , and the specification of covariates  $V_\beta$  and  $V_{\varpi_m}$ .

According to the above setting, the likelihood function for the data can be written as

$$l(\beta_i, \varpi_{i,m}, \sigma^2, \rho_i | \{t_{i,j}\}) \propto \prod_{i=1}^N \prod_{j=1}^{n_i} [t_{i,j} | Y_{i,j}, \beta_i, \varpi_{i,m}, \sigma^2, \rho_i]. \quad (2.5)$$

To reduce the computational burden of posterior calculation and exploration, we define  $\bar{\beta} \sim N_4(\alpha, V_\alpha)$ ,  $V_\beta \sim \text{Inv.Wishart}(f_0, G_0)$ ,  $\bar{\varpi}_m \sim N_3(p, V_p)$ ,  $V_{\varpi_m} \sim \text{Inv.Wishart}(f_1, G_1)$ ,

$\sigma^2 \sim \text{Inv.Gamma}(r_0/2, S_0/2)$ ,  $\rho_i \sim \text{Uniform}(-1, 1)$  as the conjugate priors on the parameters  $\beta_i, \bar{\beta}, V_\beta, \varpi_{i,m}, \bar{\varpi}_m, V_{\varpi_m}, \sigma^2$ , and  $\rho_i$ , respectively. Typically, the selection of priors is problem-specific. Some have even criticized Bayesian approach as relying on “subjective” prior information. However, we should also notice that the basis of prior information could be “objective” or data-based. The power prior developed by Ibrahim et al. [19] is an example of it. However, in most empirical cases, the utilization of diffuse prior for parameters is a reasonable default choice.

By using the Bayes theorem with the sample information and prior distribution of each parameter, the posterior distribution of each parameter can be derived. The posterior distributions and the details of the estimation procedure can be referred to Carlin and Louis [20]. The posterior distributions of estimated parameters can be summarized as the full conditional probability formulas shown in the Appendix.

In addition to the posterior distribution for the estimated parameter, the predictive distribution of the unobserved cycle time,  $t_{\text{pred}}$ , given the observed cycle time,  $t$ , is one of our main objectives. The predictive distribution is analytically intractable because of the requirement of highly dimensional numerical integration. However, the Markov chain Monte Carlo (MCMC) method provides an alternative, whereby we sample from the posterior directly and obtain sample estimates of the quantities of interest, thereby performing the integration implicitly [21]. In other words, Bayesian analysis of hierarchical models has been made feasible by the development of MCMC methods for generating samples from the full conditionals of the parameters given the remaining parameters and the data.

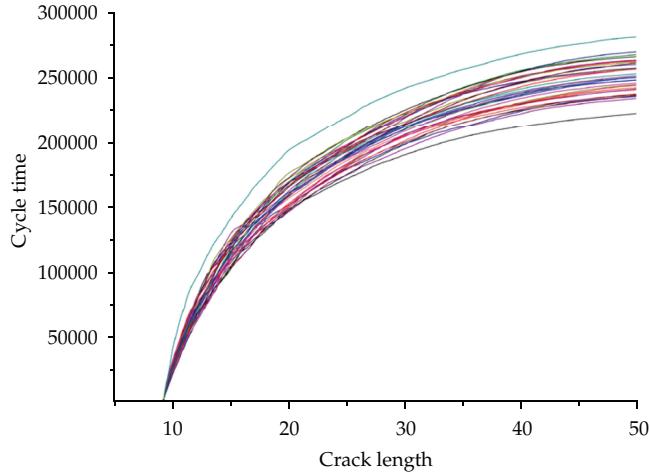
Among these MCMC methods, Gibbs sampling algorithm is one of the best known estimation procedures that uses simulation as its basis [22] and will be used herein to estimate our parameters. It has been shown that, under the mild condition, the Markov chain will converge to a stationary distribution [23]. Beginning with the conditional probability distributions in (5), the Gibbs and Metropolis-Hastings sampling procedure uses recursive simulation to generate random draws. Details of the conditional distributions for the full information model are available upon request. The values of these random draws are then used as the conditional values in each conditional probability distribution, and according to the procedure, generated random draws are carried out again in the next iteration. After numerous reiterated simulations are performed in this way, the convergent results yield random draws that are the best estimates of the parameters.

### 3. Illustrative Example and Model Estimation

#### 3.1. Illustrative Example

We use the fatigue crack growth data from Bogdanoff and Kozin [15] as an illustrative example to demonstrate the modeling procedure and effectiveness of the proposed Hierarchical Bayesian neural network approach. Figure 1 is a plot of the total 30 sample degradation paths. It is obvious that variability amongst paths does exist. There are several possible factors, such as different operating conditions and different material properties, which could cause the variability. Therefore, it is a big challenge to construct a model to capture the statistical properties of degradation paths and to predict failure time.

In this data set, there are 30 sample paths in total and each sample path has 164 paired observations, cycle time, and crack length. The cycle time is observed at some fixed crack lengths. We predefined the path as “failure” as soon as its crack length reaches a particular



**Figure 1:** Thirty paths of fatigue crack growth data from Bogdanoff and Kozin [15].

critical level of degradation (i.e.,  $D_f = 49$  mm) and assumed the experiment was terminated at 40 mm. In other words, based on the measurements of degradation from 9 mm to 40 mm, we would like to model the degradation process and use the proposed model to predict the failure time  $t_{i,j} = D_f$  to the assumed critical level for the degradation path (i.e., crack length = 49 mm). As mentioned, because the fatigue experiment was conducted on paths with fixed crack length, we are interested in the predicted failure time for the path when a specific crack length (i.e., 49 mm) is reached.

### 3.2. Model Estimation

Because the coefficients  $\beta_{i,m}$  and  $\varpi_{i,k,m}$  used to depict the degradation process are high dimensional, it is difficult to integrate out these parameters to obtain the distribution of failure time, especially when complex interactions among random parameters are present. To solve this problem, estimation was carried out using the Markov chain Monte Carlo methods using *R* language. The chain ran for 20,000 iterations, and the last 10,000 iterations are used to obtain parameter estimates. Convergence was assessed by starting the chain from multiple points and inspecting time-series plots of model parameters. Posterior draws from the full conditional are used to compute means and standard deviations of the parameter estimates.

Table 1 reports the posterior mean and standard deviation of the parameters for the proposed model. It shows that the values of  $\Gamma_{m=1,j}$ ,  $\Gamma_{m=2,j}$ , and  $\Gamma_{m=3,j}$  become steady when  $\ln(y_{i,j})$  becomes a large number. The covariance matrix of the heterogeneity distribution is reported in Table 2. It shows that the posterior mean of the diagonal elements of matrix  $V_\beta$  are ranged from 0.0002 to 0.0004. Compared to the outputs of hidden nodes ( $\Gamma_{m,j}$  ranged from 0 to 1), all these diagonal elements are not really small. It represents that the heterogeneity across paths does exist. According to above findings, we can conclude that the proposed HBNN model can successfully determine the heterogeneity across various paths even though, in this particular data set, we were unable to provide explanation to the cause of heterogeneities because of the limited information in data.

**Table 1:** Estimated mean and STD for posterior parameters.

	Estimated parameter					
	$\bar{\beta}$	$\bar{w}_{m=1}$	$\bar{w}_{m=2}$	$\bar{w}_{m=3}$	$\sigma^2$	$\bar{\rho}_i$
Posterior Mean	$\begin{bmatrix} -0.0013 \\ 0.3218 \\ 0.2925 \\ 0.2749 \end{bmatrix}$	$\begin{bmatrix} -8.0899 \\ 0.1952 \\ 0.8385 \end{bmatrix}$	$\begin{bmatrix} -38.3625 \\ -0.6606 \\ 19.3739 \end{bmatrix}$	$\begin{bmatrix} -26.610 \\ -0.1878 \\ 9.8855 \end{bmatrix}$	0.001521	0.7806321
Posterior STD	$\begin{bmatrix} 0.0153 \\ 0.0119 \\ 0.0178 \\ 0.0152 \end{bmatrix}$	$\begin{bmatrix} 0.0075 \\ 0.0003 \\ 0.0012 \end{bmatrix}$	$\begin{bmatrix} 0.0093 \\ 0.0008 \\ 0.0093 \end{bmatrix}$	$\begin{bmatrix} 0.0019 \\ 0.0070 \\ 0.0084 \end{bmatrix}$	0.000313	0.073125

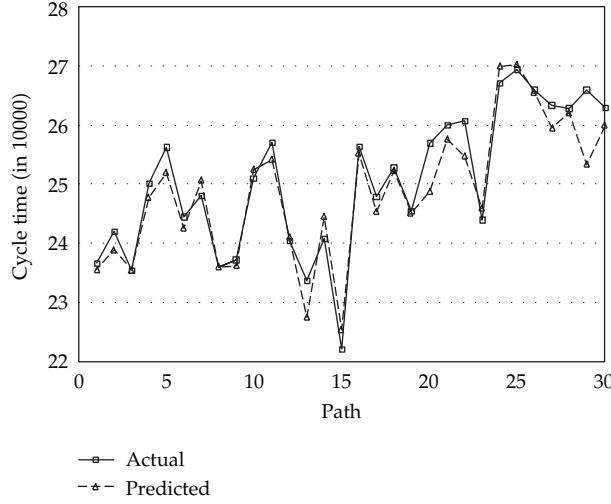
**Table 2:** Covariance matrix of the distribution of heterogeneity.

Estimated parameter	Posterior Mean	Posterior STD
$V_{\beta}$	$\begin{bmatrix} 0.043 & 0.016 & 0.023 & 0.021 \\ 0.023 & 0.001 & 0.009 & 0.044 \\ 0.029 & 0.050 & 0.024 & 0.020 \\ 0.023 & 0.018 & 0.006 & 0.041 \end{bmatrix}$	$\begin{bmatrix} 0.0043 & 0.0024 & 0.0044 & 0.0006 \\ 0.0051 & 0.0022 & 0.0044 & 0.0051 \\ 0.0030 & 0.0009 & 0.0040 & 0.0030 \\ 0.0039 & 0.0003 & 0.0001 & 0.0033 \end{bmatrix}$
$V_{\bar{w}_{m=1}}$	$\begin{bmatrix} 0.062 & 0.022 & 0.044 \\ 0.074 & 0.053 & 0.016 \\ 0.033 & 0.046 & 0.063 \end{bmatrix}$	$\begin{bmatrix} 0.009 & 0.013 & 0.038 \\ 0.055 & 0.039 & 0.068 \\ 0.051 & 0.006 & 0.018 \end{bmatrix}$
$V_{\bar{w}_{m=2}}$	$\begin{bmatrix} 0.144 & 0.051 & 0.020 \\ 0.026 & 0.003 & 0.024 \\ 0.081 & 0.043 & 0.099 \end{bmatrix}$	$\begin{bmatrix} 0.033 & 0.021 & 0.028 \\ 0.006 & 0.011 & 0.057 \\ 0.066 & 0.012 & 0.069 \end{bmatrix}$
$V_{\bar{w}_{m=3}}$	$\begin{bmatrix} 0.075 & 0.005 & 0.033 \\ 0.036 & 0.076 & 0.001 \\ 0.020 & 0.045 & 0.073 \end{bmatrix}$	$\begin{bmatrix} 0.071 & 0.016 & 0.034 \\ 0.029 & 0.001 & 0.057 \\ 0.062 & 0.002 & 0.078 \end{bmatrix}$

## 4. Failure Time Prediction and Sensitivity Analysis

### 4.1. Failure Time Prediction

The model estimation shown in Section 3 allows us to predict failure time  $t_{i,j}$  to the assumed critical level of degradation (i.e.,  $D_f = 49$  mm) based on the measurements of degradation from 9 mm to 40 mm. In order to demonstrate the effectiveness of the proposed hierarchical Bayesian neural network model, the prediction performance is evaluated using the following performance measures: the root mean square error (RMSE), mean absolute difference (MAD), mean absolute percentage error (MAPE), and root mean square percentage error (RMSPE). The definitions of these criteria were summarized in Table 3. RMSE, MAD, MAPE, and RMSPE are measures of the deviation between actual and predicted failure times. The smaller the deviation, the better the accuracy. The failure time prediction results using the proposed hierarchical Bayesian neural network model are computed and summarized in Figure 2 and Table 4. Table 4 shows that RMSE, MAD, MAPE, and RMSPE of the HBNN model are 0.37340, 0.27121, 1.058%, and 1.440%, respectively. It can be observed that these values are very small. It indicates that there is a smaller deviation between the actual and predicted failure times obtained by the proposed model. Moreover, the proposed HBNN can provide not only posterior estimates of the spatial covariance but also a natural way to incorporate the model uncertainty in statistical inference.



**Figure 2:** Prediction of failure time at 49 mm (when data collection is stopped at 40 mm).

**Table 3:** Performance measures and their definitions.

Metrics	Calculation
RMSE	$\text{RMSE} = \sqrt{\sum_{i=1}^{30} (T_i - P_i)^2 / 30}$
MAD	$\text{MAD} = \sum_{i=1}^{30}  T_i - P_i  / 30$
MAPE	$\text{MAPE} = \sum_{i=1}^{30}  (T_i - P_i) / T_i  / 30$
RMSPE	$\text{RMSPE} = \sqrt{\sum_{i=1}^{30} ((T_i - P_i) / T_i)^2 / 30}$

\*Note that  $T$  and  $P$  represent the actual and predicted failure time, respectively.

**Table 4:** Summary of failure time prediction results by HBNN model.

Models	RMSE	MAD	MAPE	RMSPE
HBNN	0.37340	0.27121	1.058%	1.440%

## 4.2. Sensitivity Analysis

To evaluate the sensitivity of the proposed method, the performance of the HBNN model was tested using different number of hidden nodes and training sample sizes. In this section, we set the number of hidden nodes as 3, 4, 5, and 6. And three different sizes of the training dataset (observations collected from 9 (mm) to 30 (mm), 9 (mm) to 35 (mm), and 9 (mm) to 40 (mm) resp.) were considered. The prediction results made by the HBNN model are summarized in Table 5 in terms of RMSE, MAD, MAPE, and RMSPE.

According to the table, the HBNN model has a lower RMSE, MAD, MAPE, and RMSPE with observations collected from 9 (mm) to 40 (mm) than with observations collected from 9 (mm) to 30 (mm). This is because the sample size of the 9 (mm) to 30 (mm) dataset is smaller than the sample size of the 9 (mm) to 40 (mm) dataset. However, the RMSE, MAD, MAPE, and RMSPE are almost the same for the cases of hidden nodes = 3, 4, 5, or 6. This result suggests that there is no difference for the predictions when the number of hidden nodes varies.

**Table 5:** Sensitivity analysis.

# of hidden nodes	Training dataset	RMSE	MAD	MAPE	RMSPE
3	from 9 (mm) to 30 (mm)	0.40576	0.29048	1.108%	1.543%
	from 9 (mm) to 35 (mm)	0.38022	0.27297	1.081%	1.486%
	from 9 (mm) to 40 (mm)	0.37340	0.27121	1.058%	1.440%
4	from 9 (mm) to 30 (mm)	0.40568	0.29060	1.089%	1.558%
	from 9 (mm) to 35 (mm)	0.38070	0.27326	1.057%	1.487%
	from 9 (mm) to 40 (mm)	0.37366	0.27088	1.089%	1.467%
5	from 9 (mm) to 30 (mm)	0.40592	0.29036	1.082%	1.520%
	from 9 (mm) to 35 (mm)	0.38076	0.27350	1.056%	1.456%
	from 9 (mm) to 40 (mm)	0.37391	0.27065	1.062%	1.435%
6	from 9 (mm) to 30 (mm)	0.40606	0.29047	1.104%	1.536%
	from 9 (mm) to 35 (mm)	0.38051	0.27393	1.098%	1.489%
	from 9 (mm) to 40 (mm)	0.37357	0.27068	1.076%	1.461%

## 5. Conclusion

In this paper, we applied the HBNN approach to model the degradation process and to make the failure time prediction. In the process of developing the HBNN model, the MCMC was utilized to estimate the parameters. Since the prediction of failure time made by HBNN model can sufficiently represent the actual data, the time-to-failure distribution can also be obtained successfully. In order to demonstrate the effectiveness of the proposed hierarchical Bayesian neural network model, the prediction performance of the proposed model is evaluated using multiple performance measurement criteria. Sensitivity evaluation of the proposed model is also conducted using different number of hidden nodes and training sample sizes. As the results reveal, using HBNN can provide not only the predictive distribution but also accurate parameter estimate. By specifying the random effects on the coefficients  $\beta_i$  and  $\varpi_{i,m}$  in the HBNN model, the heterogeneity varying across individual products can be studied. Based on these heterogeneities, the engineers will be able to conduct a further investigation in the manufacturing process and then to find out the causes of differences.

For the future research, statistical inferences of failure time based on degradation measurement, such as failure rate and tolerance limits, can be further evaluated given the predicted failure time. In addition, for some highly reliable products, it is not easy to obtain the failure data even under the elevated stresses. In such case, accelerated degradation testing (ADT) can be an alternative that provides an efficient channel for failure time prediction. The proposed HBNN approach can also be applied to depict the stress-related degradation process by including those stress factors as covariates in the model.

## Appendix

### The Full Conditional Probability of Estimated Parameters

$$\begin{aligned} [\beta_i \mid \text{rest}] &\propto [t_i \mid Y_i, \beta_i, \varpi_{i,m}, \sigma^2, \rho_i] \cdot [\beta_i \mid \bar{\beta}, V_\beta], \\ [\bar{\beta} \mid \text{rest}] &\propto \prod_{i=1}^N [\beta_i \mid \bar{\beta}, V_\beta] \cdot [\bar{\beta} \mid \mu_{\bar{\beta}}, V_{\bar{\beta}}], \end{aligned}$$

$$\begin{aligned}
[V_\beta \mid \text{rest}] &\propto \prod_{i=1}^N [\beta_i \mid \bar{\beta}, V_\beta] \cdot [V_\beta \mid f_0, G_0], \\
[\varpi_{i,m} \mid \text{rest}] &\propto \prod_{j=1}^{n_i} [t_i \mid \Gamma_i, \beta_i, \varpi_{i,m}, \sigma^2, \rho_i] \cdot [\varpi_{i,m} \mid \bar{\varpi}_m, V_{\varpi_m}], \\
[\bar{\varpi}_m \mid \text{rest}] &\propto \prod_{i=1}^N [\varpi_{i,m} \mid \bar{\varpi}_m, V_{\varpi_m}] \cdot [\bar{\varpi}_m \mid p, V_p], \\
[V_{\varpi_m} \mid \text{rest}] &\propto \prod_{i=1}^N [\varpi_{i,m} \mid \bar{\varpi}_m, V_{\varpi_m}] \cdot [V_{\varpi_m} \mid f_1, G_1], \\
[\sigma^2 \mid \text{rest}] &\propto \prod_{i=1}^N [t_i \mid Y_i, \beta_i, \varpi_{i,m}, \sigma^2, \rho_i] \cdot [\sigma^2 \mid r_0, s_0], \\
[\rho_i \mid \text{rest}] &\propto [t_i \mid Y_i, \beta_i, \varpi_{i,m}, \sigma^2, \rho_i] \cdot [\rho_i].
\end{aligned} \tag{A.1}$$

## References

- [1] C. J. Lu and W. Q. Meeker, "Using degradation measures to estimate a time-to-failure distribution," *Technometrics*, vol. 35, no. 2, pp. 161–174, 1993.
- [2] C. J. Lu, W. Q. Meeker, and L. A. Escobar, "A comparison of degradation and failure-time analysis methods for estimating a time-to-failure distribution," *Statistica Sinica*, vol. 6, no. 3, pp. 531–546, 1996.
- [3] W. Q. Meeker and M. Hamada, "Statistical tools for the rapid development & evaluation of high-reliability products," *IEEE Transactions on Reliability*, vol. 44, no. 2, pp. 187–198, 1995.
- [4] K. A. Doksum and A. Hoyland, "Models for variable-stress accelerated life testing experiments based on Wiener processes and the inverse Gaussian distribution," *Technometrics*, vol. 34, no. 1, pp. 74–82, 1992.
- [5] J. M. Bernardo and A. F. M. Smith, *Bayesian Theory*, John Wiley & Sons, 1994.
- [6] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian Data Analysis*, Chapman & Hall, London, UK, 1995.
- [7] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer, New York, NY, USA, 1999.
- [8] S. I. Liu, "Bayesian model determination for binary-time-series data with applications," *Computational Statistics & Data Analysis*, vol. 36, no. 4, pp. 461–473, 2001.
- [9] R. Zheng and B. R. Ellingwood, "Stochastic fatigue crack growth in steel structures subjected to random loading," *Structural Safety*, vol. 20, no. 4, pp. 305–324, 1998.
- [10] R. Zhang and S. Mahadevan, "Model uncertainty and Bayesian updating in reliability-based inspection," *Structural Safety*, vol. 22, no. 2, pp. 145–160, 2000.
- [11] M. Akama, "Bayesian analysis for the results of fatigue test using full-scale models to obtain the accurate failure probabilities of the Shinkansen vehicle axle," *Reliability Engineering and System Safety*, vol. 75, no. 3, pp. 321–332, 2002.
- [12] R. M. Neal, *Bayesian Learning for Neural Networks*, Springer, New York, NY, USA, 1996.
- [13] P. Müller and D. Rios Insua, "Issues in bayesian analysis of neural network models," *Neural Computation*, vol. 10, pp. 571–592, 1998.
- [14] C. C. Holmes and B. K. Mallick, "Bayesian wavelet networks for nonparametric regression," *IEEE Transactions on Neural Networks*, vol. 11, no. 1, pp. 27–35, 2000.
- [15] J. L. Bogdanoff and F. Kozin, *Probabilistic Models of Cumulative Damage*, John Wiley, New York, NY, USA, 1984.

- [16] E. P. Liski and T. Nummi, "Prediction in repeated-measures models with engineering applications," *Technometrics*, vol. 38, no. 1, pp. 25–36, 1996.
- [17] R. B. Chinnam, "On-line reliability estimation for individual components using statistical degradation signal models," *Quality and Reliability Engineering International*, vol. 18, no. 1, pp. 53–73, 2002.
- [18] R. B. Chinnam and P. Baruah, "A neuro-fuzzy approach for on-line reliability estimation and condition based-maintenance using degradation signals," *International Journal of Materials and Product Technology*, vol. 20, no. 1–3, pp. 166–179, 2004.
- [19] J. G. Ibrahim, L. M. Ryan, and M.-H. Chen, "Using historical controls to adjust for covariates in trend tests for binary data," *Journal of the American Statistical Association*, vol. 93, no. 444, pp. 1282–1293, 1998.
- [20] B. P. Carlin and T. A. Louis, *Bayes and Empirical Bayes Methods for Data Analysis*, CRC Press, London, UK, 2000.
- [21] S. P. Brooks, "Markov chain Monte Carlo method and its application," *Journal of the Royal Statistical Society D*, vol. 47, part 1, pp. 69–100, 1998.
- [22] A. E. Gelfand and A. F. Smith, "Sampling-based approaches to calculating marginal densities," *Journal of the American Statistical Association*, vol. 85, no. 410, pp. 398–409, 1990.
- [23] H. J. Kushner and P. Dupuis, *Numerical Methods for Stochastic Control Problems in Continuous Time*, vol. 24, Springer, New York, NY, USA, 2001.

*Research Article*

# A Fuzzy Dropper for Proportional Loss Rate Differentiation under Wireless Network with a Multi-State Channel

**Yu-Chin Szu**

*Department of Information Management, St. John's University, 499, Section 4, Tam King Road, Tamsui District, New Taipei City 25135, Taiwan*

Correspondence should be addressed to Yu-Chin Szu, belinda@mail.sju.edu.tw

Received 18 October 2011; Revised 25 January 2012; Accepted 3 February 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Yu-Chin Szu. 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 proportional loss rate differentiation (PLD) model was proposed to provide controllable and predictable loss rate for different classes of wired network connections. However, these algorithms cannot be directly applied to wireless networks, because of the location-dependent and time-varying wireless channel capacity. This paper proposes a novel packet dropper for fuzzy controlling of the proportional loss rate differentiation in a wireless network with multistate channel. The proposed dropper, fuzzy proportional loss rate dropper (FPLR), prefers to drop the small packets destined to a poor condition channel to improve the network performance. The loss rate debts of the poor channel will be compensated later to keep PLD. From simulation results, FPLR does achieve accurate loss rate proportion, lower queuing delay and loss rate, and higher throughput, compared with other methods in the wireless environment.

## 1. Introduction

The Internet becomes an important infrastructure of global communication. However, many multimedia applications require some guarantee of the quality of services (QoS). Thus the best-effort service is not suitable to such applications because it cannot promise any guarantee about packet loss, delay, or jitter. The Internet Engineering Task Force (IETF) first proposed the Integrated Service (IntServ) as a QoS architecture for IP networks [1]. Because IntServ suffers from the scalability problem, the Differentiated Service (DiffServ) [2] was then proposed. Diffserv has proceeded in two directions—absolute differentiated service and relative differentiated service. The absolute differentiated service ensures that an admitted user can enjoy certain and steady performance, while the relative differentiated service ensures that users with a higher service class experience better performance than the

users with a lower one. The proportional differentiation model, a branch of relative service differentiation, has received a lot of attention because it can perform the controllable and predictable differentiation [3, 4]. That is, the proportional differentiation model offers the network manager a means of varying quality spacing between service classes according to the given pricing or policy criteria and ensures that the differentiation between classes is consistent in any measured timescale.

The proportional services can be differentiated according to different performance metrics, such as throughput, delay, loss rate, or jitter. When adopting packet loss rate as the performance metric, the proportional differentiation model is referred to as the proportional loss rate differentiation (PLD) model. To provide PLD, some methods, such as Proportional Loss Rate (PLR) [5], Average Drop Distance (ADD) [6], Debt Aware [7], Weighted Random Early Detection (WRED) [8], Weight Simple Adaptive Proportion (WSAP) [9], Hop-count Probabilistic Packet Dropper (HPPD) [10], and RED with maximum drop probability adjustment, were proposed.

As wireless technologies rapidly advance and lightweight portable computing devices become popular, wireless networks have become pervasive. Accordingly, the PLD model is also urgently required for wireless environments, just as it was for wired networks. However, the above approaches designed in a wired network are not applicable in a wireless environment, which has some distinct characteristics, such as high error rate, location-dependent and time-varying capacity, and scarce bandwidth [11].

In a wireless network with a multirate channel, when many packets encountering a good channel are dropped and many packets encountering a poor channel are kept in the buffer, the system performance becomes poor because most packets are transmitted in a poor-capacity channel. Actually, dropping the packet having the poorest channel will cause the optimal performance, but this behavior completely ignores to maintain the PLD model. Therefore, how to keep the PLD model and raise the overall performance in a wireless network with a multirate channel is a challenge and will be investigated in this paper.

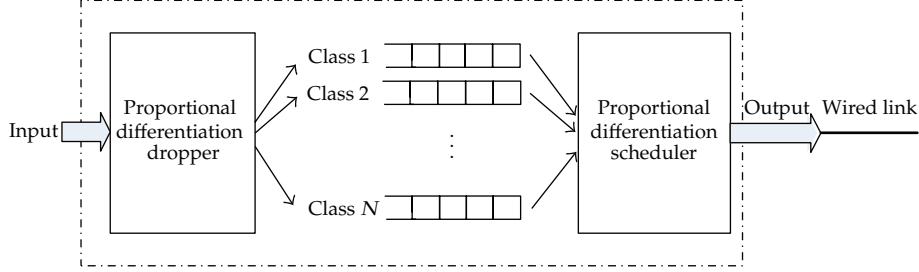
This paper proposes a novel algorithm, named fuzzy proportional loss rate dropper (FPLR). In the FPLR dropper, we utilize the concept of fuzzy control to make an optimal decision for dropping the small packets destined to a poor channel and reserving the large packets destined to a good channel, causing a lot of large packets to be transmitted via a good channel. Therefore, FPLR can provide high performance and keep PLD in a wireless with a multistate channel.

Organization of the remainder of the paper is as follows. Section 2 describes the background, including the proportional differentiation model, some previous related works, structure of fuzzy controller systems, and what problems occur when these works are directly applied into a wireless network. Section 3 describes in more detail the FPLR and the design of the fuzzy controller. In Section 4, some simulation results and their implications are exhibited. The conclusion of this work is given in Section 5.

## 2. Background

### 2.1. Proportional Differentiation Model

The proportional differentiation model was proposed first by Dovrolis and Ramanathan [5] and Bobin et al. [6], and its structure is shown as Figure 1. The arrival traffic is classified into  $N$  service classes where each class has a dedicated queue. Let  $q_i$  denote the measured



**Figure 1:** The proportional differentiation model.

performance of class  $i$ . For proportional differentiation model, the following equation should be satisfied for all pairs of classes,  $i$  and  $j$ ,

$$\frac{q_i}{q_j} = \frac{c_i}{c_j} \quad (1 \leq i, j \leq N), \quad (2.1)$$

where  $c_1, c_2, \dots, c_N$  are the quality differentiation parameters (QDP). A network operator could manipulate the service quality spacing between classes by adjusting QDPs.

Let  $\bar{L}_i$  be the average loss of the class- $i$  packets with  $\sigma_i$  as the loss differentiation parameters (LDPs). For all pairs of service classes,  $i$  and  $j$ , the proportional loss differentiation model is specified by

$$\frac{\bar{L}_i}{\bar{L}_j} = \frac{\sigma_i}{\sigma_j} \quad (1 \leq i, j \leq N). \quad (2.2)$$

Let  $\bar{D}_i$  be the average queuing delay of the class- $i$  packets with  $\delta_i$  as the delay differentiation parameters (DDPs). The proportional delay differentiation model has the following constraint for any pair of classes:

$$\frac{\bar{D}_i}{\bar{D}_j} = \frac{\delta_i}{\delta_j} \quad (1 \leq i, j \leq N). \quad (2.3)$$

## 2.2. Proportional Loss Rate (PLR)

Dovrolis et al. proposed the proportional loss rate dropper (PLR) to offer the PLD model [5]. In order to determine which packet should be dropped, PLR uses a loss history table (LHT) to record the loss rate of each class at present. Let  $\bar{L}_i(t)$  be the average loss rate and  $\tilde{L}_i(t)$  be the normalized average loss rate of class  $i$  at time  $t$ . Also  $L_i(t)$  and  $A_i(t)$  denote the numbers of dropped packets and arrived packets, respectively, of class  $i$  until time  $t$ .  $B(t)$  denotes the set of backlogged classes. PLR chooses class  $J$  to drop its tail packet as follows:

$$\bar{L}_i(t) = \frac{L_i(t)}{A_i(t)} \quad (1 \leq i \leq N), \quad (2.4)$$

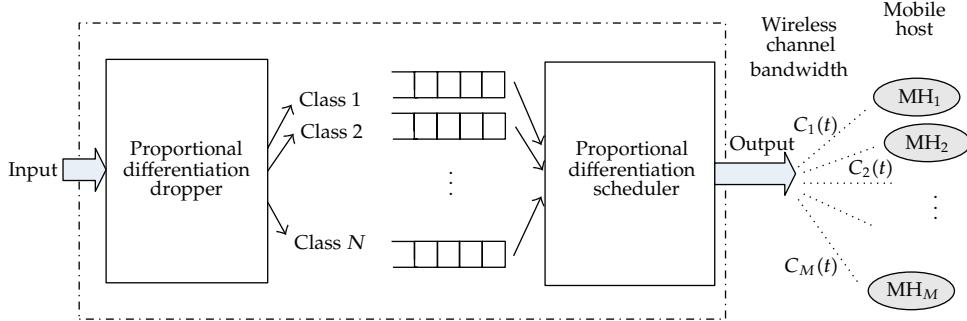


Figure 2: The proportional differentiation model in a wireless network.

$$\tilde{L}_i(t) = \frac{\bar{L}_i(t)}{\sigma_i} \quad (1 \leq i \leq N), \quad (2.5)$$

$$J = \arg \min_{i \in B(t)} \tilde{L}_i(t) \quad (1 \leq i \leq N). \quad (2.6)$$

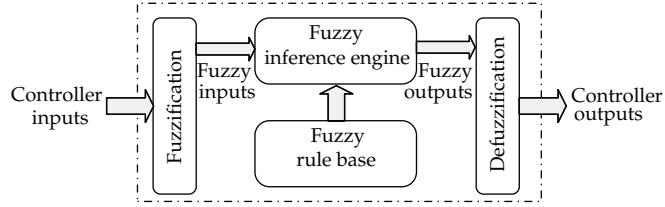
PLR aims to maintain a unanimous normalized average loss rate among all classes. According to the number of packets that PLR estimates, the calculated average loss rate is different; so there are two kinds of algorithms, namely, PLR with infinite memory,  $\text{PLR}(\infty)$ , and PLR with memory  $M$ ,  $\text{PLR}(M)$ . When calculating the average loss rate,  $\text{PLR}(\infty)$  counts packets from initial to present, while  $\text{PLR}(M)$  observes the last  $M$  packets of every class at present. From the long-term observation, the result of  $\text{PLR}(\infty)$  is closer to targeted proportion than that of  $\text{PLR}(M)$ . However, when the class load significantly oscillates, adopting  $\text{PLR}(M)$  is preferred because of its adaptation, but determining an optimal  $M$  is difficult.

### 2.3. The Problem of Achieving PLD in a Wireless Network

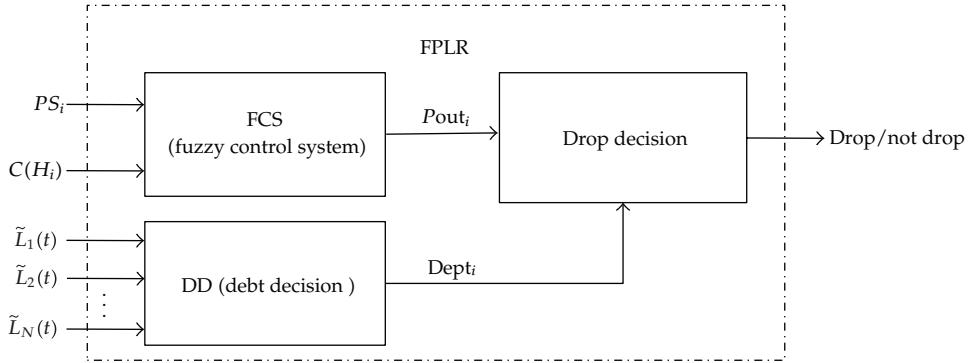
Previous studies on achieving the PLD model focused on increasing the accuracy of the achieved loss proportion between classes in wired networks. These proposed algorithms did not consider any characteristics in a wireless network.

Figure 2 depicts the PLD model applied in a wireless network. In this environment, all mobile hosts share one wireless link. Since each host could be located at different places, different capacities exist when the scheduler transmits data to different mobile hosts via this wireless link. Also as the mobile host moves, the capacity to this destined host varies. Thus a wireless link has a location-dependent and time-varying capacity, and it is called as a multistate link herein. For simplicity, the term channel  $j$  means the wireless link at transmitting the packet to the mobile host  $j$ . Let  $C_j(t)$  denote the encountered capacity when the scheduler transmits packets via channel  $j$  at time  $t$ .

In a wireless network with a multirate link, when many packets encountering a good channel are dropped and many packets encountering a poor channel are kept in the buffer, the system performance becomes bad because most packets will be transmitted via a poor-capacity channel. Actually, dropping the packet having the poorest channel will cause the optimal performance, but this behavior completely ignores keeping the PLD model.



**Figure 3:** The architecture of the fuzzy controller.



**Figure 4:** The structure of FPLR.

Therefore, designing a dropper to simultaneously keep the PLD model and raise the overall performance in a wireless network with a multirate link is a challenge.

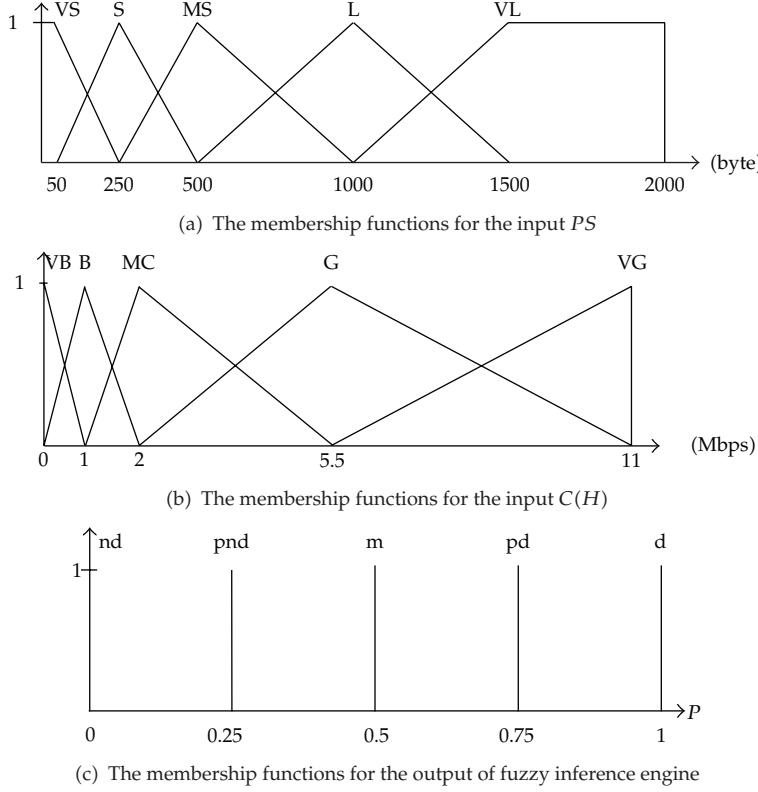
#### 2.4. The Fuzzy Controller

The fuzzy controller that is a nonlinear mapping system developed by Mamdani [12, 13] consists of four main components: fuzzification, fuzzy inference engine, rule base, and defuzzification, as shown in Figure 3.

The fuzzification (or fuzzifier) calculates suitable sets of degree of membership, called “fuzzy sets,” for crisp inputs. The fuzzy inference engine performs inference procedure for given inputs according to the fuzzy rules to derive proper actions. The rule base consists of a set of linguistic rules, as well as a set of membership functions for linguistic values. The defuzzification (or defuzzifier) converts the fuzzy output to crisp values.

### 3. Fuzzy Proportional Loss Rate Dropper (FPLR)

FPLR uses three processes to decide how to drop packets, as shown in Figure 4. A fuzzy controller system on the first process takes advantage of fuzzy theory to PLD. The debt decision module in the second process is to compensate for the loss rate debts. The third process is to determine an appropriate packet to drop.



**Figure 5:** The membership functions.

### 3.1. Fuzzification

In this paper, most of the membership functions for the fuzzy sets are chosen to be triangular, because triangular membership functions are the most economic. All membership functions are defined on common interval  $[0, 1]$ . With fuzzy logic, we assign grade values to our two variables, which are packet sizes and channel states. Thus the fuzzy set A consists of two fuzzy variables,  $PS$ ,  $C(H)$ , where  $PS$  is the fuzzy variable term for the packet size and  $C(H)$  is the channel capacity encountered when the scheduler transmits the HOL packet. The membership functions of  $PS$  and  $C(H)$  are shown in Figures 5(a) and 5(b), respectively, and they are the input of the FCS. The fuzzification component translates the inputs into corresponding fuzzy values.

In Figure 5(a), VS, S, MS, L, and VL denote very small, small, medium small, large, and very large, respectively. In Figure 5(b), VB, B, MC, G, and VG indicate very bad channel, bad channel, medium channel, good channel, and very good channel, respectively.

Let the fuzzified output of the fuzzy inference engine be  $P$ , which uses singleton fuzzy sets representing the drop probability of the packet, as shown in Figure 5(c). In Figure 5(c), nd, pnd, m, pd, and d denote not drop, probably not drop, maybe drop, probably drop, and drop, respectively.

**Table 1:** The rule base.

		Packet sizes				
		VS	S	MS	L	VL
Channel capacity	VB	d	d	d	d	d
	B	pd	pd	pd	m	m
	MC	pd	pd	m	pnd	pnd
	G	m	m	pnd	pnd	pnd
	VG	nd	nd	nd	nd	nd

### 3.2. Rule Establishment

Fuzzy rules can effectively treat the nonlinearity, and the whole fuzzy rules are shown in Table 1. Each rule performs the mapping process from the fuzzy input to fuzzy out. These rules are intuitive. For example, when channel state is very bad and no matter a packet is big or small, an incoming packet must be dropped.

### 3.3. Defuzzification

The center of gravity (CoG) technique, which computes the weighted average of the center of gravity of each membership function, is used to compute the defuzzified output,  $P_{\text{out}}$ , of the FCS. That is  $P_{\text{out}i} = \sum_{f=1}^n \mu_{i,f}(P_{i,f}) P_{i,f} / \sum_{f=1}^n \mu_{i,f}(P_{i,f})$ , where  $P_{i,f}$  is the center of the membership function recommended by the consequence of rule  $f$  of class  $i$ , and the height is cut by the minimum value after MAX-MIN inference engine [14].  $\mu_{i,f}(P_{i,f})$  is the degree of membership of input variables, packet size and channel states, of class  $i$ . The  $n$  is number of rules.

### 3.4. Debt Decision

The debt decision is adopted because, in this work, maintaining the loss rate differentiation is considered to be more important than achieving high throughput. A debt value debt is used to record the “drop debt” carried by class  $i$ . debt $_i$  having the positive value represents that the number of loss happening is less than the expectation in class  $i$ . That is, some other classes instead of class  $i$  drop the packets. Similarly, the value of debt $_i$  being negative and zero means that class  $i$  has more and equal number of losses than its expectation, respectively. debt is specified by

$$\begin{aligned} \text{If } \tilde{L}_J(t) > \left( \frac{\sum_{i=1}^N \tilde{L}_i(t)}{N} \right) \text{ then } \text{debt}_J = -0.25, \\ \text{If } \tilde{L}_J(t) < \left( \frac{\sum_{i=1}^N \tilde{L}_i(t)}{N} \right) \text{ then } \text{debt}_J = 0.25, \end{aligned} \quad (3.1)$$

where  $\tilde{L}_J(t)$  is the normalized average loss rate of class  $J$  at time  $t$  and  $N$  is the number of service classes.

We would like to use the decision function of FPLR to decide that the packet should be dropped and need not be dropped. We use a threshold  $\theta$  to compare with  $P_{\text{out}}$ . Thus the condition of dropping the packet of the candidate class  $J$  is as follows:

$$(P_{\text{out}_J} + \text{debt}_J) > \theta. \quad (3.2)$$

### 3.5. Algorithm

Algorithm 1 presents the pseudocode of the FPLR. When the buffer has empty space, the dropper simply inserts the packet into a proper queue. When buffer overflow occurs, the dropper selects an appropriate packet to drop. Let  $L_i(t)$  and  $A_i(t)$  be the number of dropped packets and the number of arrived packets of class  $i$  at time  $t$ , respectively.

When an arriving packet encounters a full buffer, the dropper selects a proper packet, which may be this arriving packet or a packet in the buffer to be dropped for keeping the proportional loss rate among classes. At determining which packet to be dropped, the dropper first calculates the normalized average loss rate  $\tilde{L}_i(t) = L_i(t) / A_i(t)\sigma_i$  for each class  $i$ . Then the class with the smallest normalized average loss rate, that is,  $J = \arg \min_{i \in B(t)} \tilde{L}_i(t)$ , is regarded as the candidate class.

The dropper considers three important factors, including channel state, packet size, and debt degree, to determine whether the packet of the class  $J$  should be dropped.

Note the dropper drops the HOL packet, rather than the tail packet or arriving packet, because using this method can reduce the queuing delay of queued packets.

If the candidate class  $J$  does not satisfy the above condition, the dropper will choose the candidate class  $K$ , which has the next smallest normalized average loss rate. Judging whether the HOL packet of candidate class  $K$  will be dropped is similar to (3.2), that is,  $(P_{\text{out}_K} + \text{debt}_K) > \theta$ .

Some noticeable points about the FPLR are explained as follows.

- (1) FPLR drops the HOL packet, rather than the tail packet, of the selected class, because using this method can reduce the queuing delay of queued packets. Also the HOL packet will actually experience the current channel capacity, but the tail packet may not be transmitted at this capacity because of the time-varying bandwidth.
- (2) For FPLR, because the packets destined to a low-capacity channel are easily dropped and the packets destined to a high-capacity channel are easily kept in the buffer, most packets are transmitted in a good channel, generating high performance.
- (3) The fuzzy logic is an effective tool for efficient buffer management. It can easily handle several nonlinear factors and does not need detailed mathematic descriptions for the system.

## 4. Simulation and Discussion

The simulations compare the performance of FPLR and WPLR in terms of *loss rate ratio*, *loss improvement*, *throughput improvement*, and *delay improvement*. The simulations are conducted to investigate the effects of packet arrival rate, number of mobile hosts, and state transition rate

$L_i$ : the number of lost packets of class  $i$   
 $A_i$ : the number of arrived packets of class  $i$   
 $\sigma_i$ : the loss differentiation parameter of class  $i$   
 $C(H_i)$ : the channel capacity encountered when the scheduler transmits the HOL packet of class  $i$   
 $B(t)$ : the set of backlogged classes at present  $t$   
 $PS_i$ : the size of the HOL packet in class  $i$   
 $P_{i,f}$ : the center of the membership function recommended by the consequence of rule  $f$  of class  $i$   
 $\mu_{i,f}(P_{i,f})$ : the degree of membership of input variables, packet size and  
    channel states, of class  $i$   
 $P_{\text{out},i}$ : the defuzzified output of FCS of class  $i$   
 $\theta$ : drop threshold  
 $\text{debt}_i$ : lost debt of class  $i$   
**FPLR Dropper**  
 {  
        a class- $i$  packet arrives at time  $t$ ,  $A_i(t)++$ ;  
        if (buffer overflow){  
            calculate  $\tilde{L}_i(t) = L_i(t)/A_i(t)\sigma_i, i = 1, 2, \dots, N$ ;  
            calculate  $P_{\text{out},i} = \sum_{f=1}^n \mu_{i,f}(P_{i,f})P_{i,f} / \sum_{f=1}^n \mu_{i,f}(P_{i,f}), i = 1, 2, \dots, N; f = 1, 2, 3, 4, 5$ ;  
             $J = \arg \min_{i \in B(t)} \tilde{L}_i(t);$   
             $\text{debt}_i = 0;$   
            if ( $\tilde{L}_J(t) > (\sum_{i=1}^N \tilde{L}_i(t)/N)$ ) then  $\text{debt}_J = -0.25, i = 1, 2, \dots, N$ ;  
            else if ( $\tilde{L}_J(t) < (\sum_{i=1}^N \tilde{L}_i(t)/N)$ ) then  $\text{debt}_J = 0.25, i = 1, 2, \dots, N$ ;  
            if ( $(P_{\text{out},J} + \text{debt}_J) > \theta$ ){  
                drop the HOL packet from class  $J$ ;  
                 $L_J(t)++$ ;  
            }  
            else{  
                do {  
                    find  $K = \arg \min_{i \in B(t)} \tilde{L}_i$ ;  
                    if ( $(P_{\text{out},K} + \text{debt}_K) > \theta$ ){  
                        drop the HOL packet from class  $K$ ;  
                         $L_K(t)++$ ;  
                    }  
                } while(one packet is dropped or all classes have been visited)  
                if (no packet is dropped){  
                    drop the HOL packet from class  $J$ ;  
                     $L_J(t)++$ ;  
                }  
            }  
            accept the incoming packet;  
        }
 }

**Algorithm 1:** The FPLR algorithm.

upon WPLR and FPLR. Also, the behaviors of these two droppers under different timescales are explored. First, the average loss rate of each class  $i$ ,  $\bar{L}_i$ , is obtained by using (2.4), and the loss rate ratio of class  $i$  over class  $j$  is defined as  $R_{ij} = \bar{L}_i / \bar{L}_j$ . The loss improvement of class  $i$  is defined as  $(\bar{L}_i^P - \bar{L}_i^F) / \bar{L}_i^P$  where  $\bar{L}_i^P$  and  $\bar{L}_i^F$  are the loss rates of class  $i$  made by WPLR and FPLR, respectively. Similarly, the throughput improvement and delay improvement of class  $i$  are defined as  $(T_i^F - T_i^P) / T_i^P$  and  $(\bar{D}_i^P - \bar{D}_i^F) / \bar{D}_i^P$ , respectively, where  $T_i$  denotes the throughput of class  $i$  and  $\bar{D}_i$  denotes the average queuing delay of class  $i$ .

#### 4.1. Simulation Models

The model we simulated is depicted in Figure 2, and the scheduler is using the First-Come First-Served (FCFS) scheduler. In all simulations, three service classes ( $N = 3$ ) are assumed, and the corresponding loss rate differentiation parameters are set as  $\sigma_1 = 1$ ,  $\sigma_2 = 2$ , and  $\sigma_3 = 4$ . Packet arrival follows a Poisson process, and its mean overall arrival rate is  $\lambda = 120$  packets/sec. The distribution of packet sizes for all classes is such that 40% of packets are 40 bytes, 50% packets are 550 bytes, and 10% packets are 1500 bytes. Thus, the mean packet size is 441 bytes. The threshold  $\theta$  is set as 0.5. The total buffer size is 10,000 bytes, and the number of hosts is five, that is,  $M = 5$ . The wireless channel is simulated by a multistate Markov process, which has five states with the values of capacities varying among 0 (purely bad), 1 Mbps, 2 Mbps, 5.5 Mbps, and 11 Mbps (purely good). The transition rate matrix of channel capacity is set as

$$\begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{bmatrix} -a & ap_1 & ap_1^2 & ap_1^3 & ap_1^4 \\ ap_2 & -a & ap_2 & ap_2^2 & ap_2^3 \\ ap_3^2 & ap_3 & -a & ap_3 & ap_3^2 \\ ap_4^3 & ap_4^2 & ap_4 & -a & ap_4 \\ ap_5^4 & ap_5^3 & ap_5^2 & ap_5 & -a \end{bmatrix} \end{matrix} \quad (4.1)$$

where  $a$  is the state transition rate to other states and  $p_i$  is the probability of state  $i$  being translated to its neighbor states when the transition occurs. The default value of  $a$  is 30, and the values of  $p_1, p_2, p_3, p_4$ , and  $p_5$  can be calculated by letting the sum of each row equal to 0. In each simulation, at least 500,000 packets for each class are generated for the sake of stability.

#### 4.2. Packet Arrival Rate

To observe the influence of different packet arrival rates on achieved loss rate ratios, loss improvement, delay improvements and throughput improvement, the arrival rate ( $\lambda$ ) is varied from 100 to 150 packets/sec. The target loss rate ratios are 4 ( $\sigma_3/\sigma_1 = 4/1$ ) for class 3 over class 1 and 2 ( $\sigma_2/\sigma_1 = 2/1$ ) for class 2 over class 1. Figure 6(a) shows that WPLR and FPLR can achieve the accurate loss rate ratios. For the FPLR dropper, because the class with a larger debt has a higher probability to be dropped, the target loss rate proportions can be still maintained.

Observed from Figure 6(b), the loss improvements attained by FPLR increase as arrival rate increases, and three classes have the same loss improvement. At a low packet arrival rate, the packet losses seldom happen, causing that the fuzzy mechanism has minor influence. As the arrival rate increases, since the opportunity of using the fuzzy mechanism becomes large, more small packets transmitted via low-capacity channels are dropped, and more large packets transmitted via high-capacity channels are kept. Thus the loss improvements made by FPLR increase.

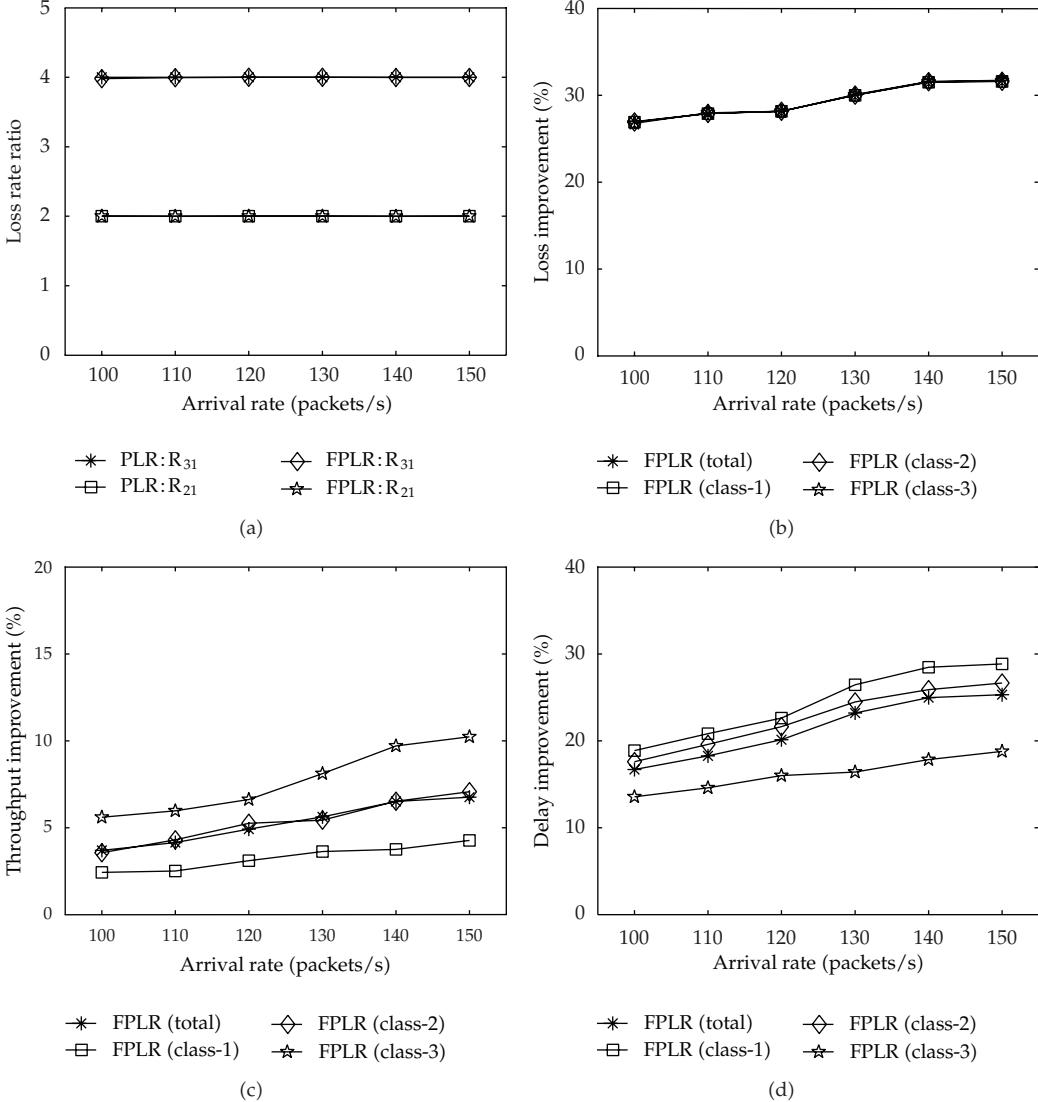


Figure 6: The effect of packet arrival rate.

From Figure 6(c), two phenomena are observed. First, the throughputs improved by FPLR increase as arrival rate increases. Second, the throughput improvements of all classes are in the order class 3 > class 2 > class 1. The first phenomena is because the more loss improvement, the more throughput improvement. The second phenomenon is not trivial and explained in the following.

Let  $\lambda_i$  be the packet arrival rate of class  $i$ . Thus for WPLR, the throughput of class  $i$ ,  $T_i^P$ , can be expressed as  $T_i^P = \lambda_i(1 - \bar{L}_i^P)$ . Also let the loss improvement of class  $i$ , made by the dropper FPLR, be denoted as  $f_i^F$ , which is equal to  $(\bar{L}_i^P - \bar{L}_i^F)/\bar{L}_i^P$ . Therefore, the throughput

of class  $i$  can be easily obtained as  $T_i^F = \lambda_i(1 - \bar{L}_i^F) = \lambda_i(1 - (1 - f_i^F)\bar{L}_i^P)$ . Hence, the throughput improvement of class  $i$  is

$$\frac{T_i^F}{T_i^P} - 1 = \frac{1 - (1 - f_i^F)\bar{L}_i^P}{1 - \bar{L}_i^P} - 1. \quad (4.2)$$

From this equation, we can prove the more loss improvement, the more throughput improvement.

For different classes, observed from Figure 6(b), the loss improvements,  $f_i^F$ , of all classes are the same. Also Figure 6(a) shows that  $\bar{L}_3^P > \bar{L}_2^P > \bar{L}_1^P$ . Under these conditions, from (4.2), the throughput improvements of all classes are in the order class 3 > class 2 > class 1. For example, all  $\lambda_i = 120$  packets/sec, loss rates of classes 1, 2, and 3 for WPLR are 20%, 40%, and 80%, respectively, that is, their throughputs are 100, 80, and 40. When the loss improvement is 50% for all classes, the loss rates of classes 1, 2, and 3 for our proposed dropper reduce to 10%, 20%, and 40%, respectively, that is, their throughputs are 110, 100, and 80. In this example, the throughput improvements for classes 1, 2, and 3 are 10%, 25%, and 100%, respectively.

Figure 6(d) shows the delay improvements achieved by FPLR. The delay improvements are caused by two reasons. First, because the packet destined to a poor channel is easier to be dropped than that destined to a good channel, nondropped packets usually encounter a good channel, causing that they have short transmission time and thus short queuing delay. Second, FPLR dropping the HOL packet, rather than dropping the tail packet, reduces the queuing delay of the queued packets. Also the delay improvements enhance as the arrival rate increases because the opportunity of using the fuzzy mechanism becomes large. Finally, the delay improvements of all classes are in the order class 1 > class 2 > class 3 since the more throughput improvement, the less delay improvement.

### 4.3. Timescale

In this simulation, the loss rate ratios between successive classes are measured over five time intervals—100, 500, 1000, 5000, and 10000 p-units, where a p-unit is the average packet transmission time. During each time interval, the loss rate ratios of class 2 over class 1 and class 3 over class 2 are measured and averaged.

Figure 7 shows five percentiles, 5%, 25%, 50% (median), 75%, and 95%, of the average loss rate ratio. FPLR has the broader ranges on short timescales than WPLR and similar ranges on long timescales. The reason is that although using the fuzzy mechanism will achieve the loss rate ratio temporarily away from target loss rate proportion in the short term, they will let the opportunity of using the debt decision mechanism keep PLD on the long term.

### 4.4. Number of Mobile Hosts

The number of mobile hosts is varied from 5 to 25 to observe the influence on achieved loss rate ratios, loss improvement, delay improvement, and throughput improvement. Packets of each class are evenly distributed to each mobile host. Since below all simulation results for different classes have the similar trends as Figure 6. Therefore we will only show the total

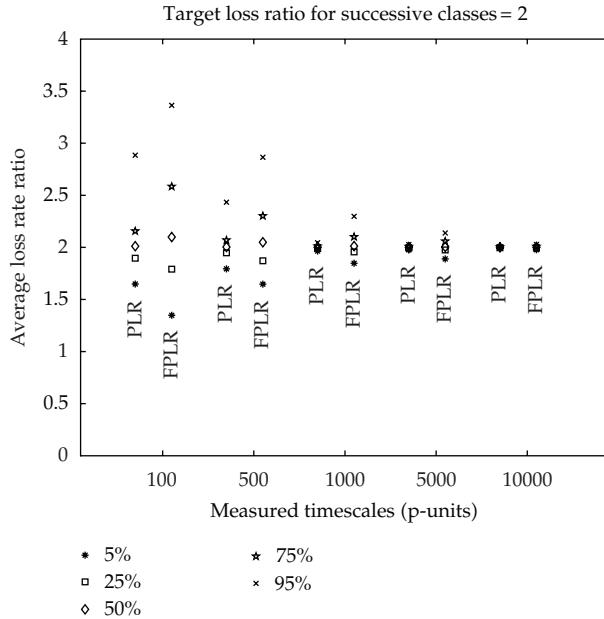


Figure 7: Five percentiles of average loss rate ratio on various measured timescales.

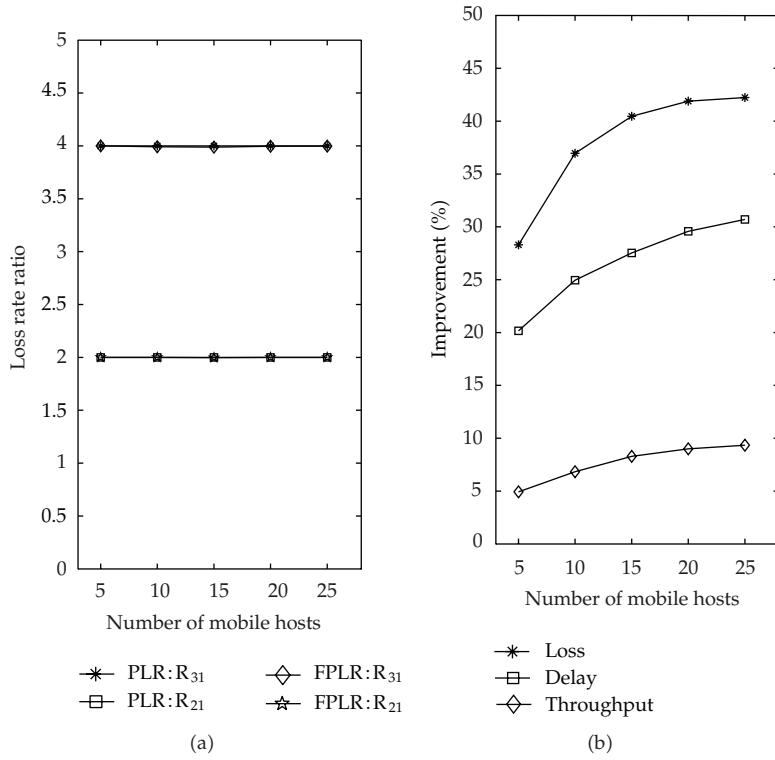


Figure 8: The effect of the number of mobile hosts.

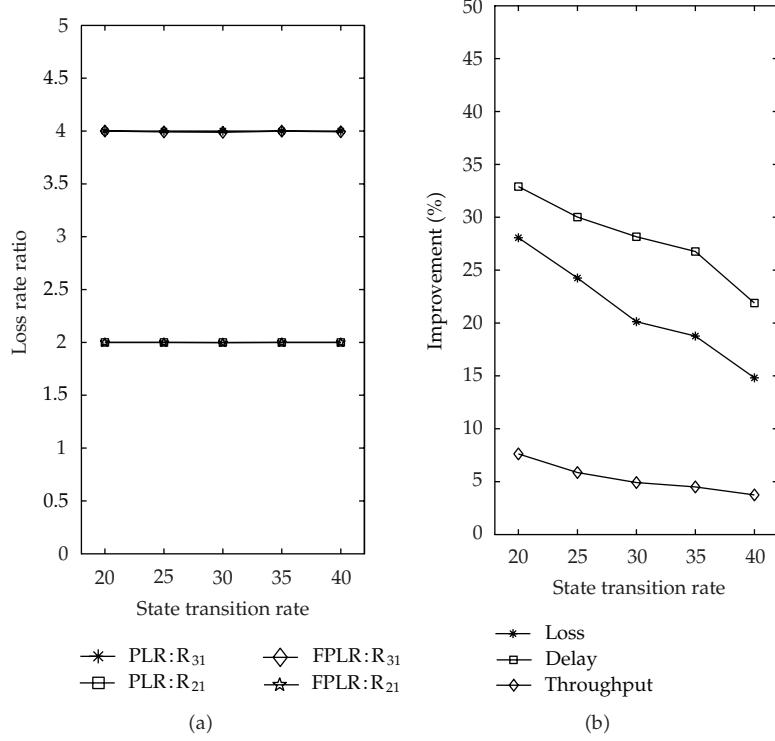
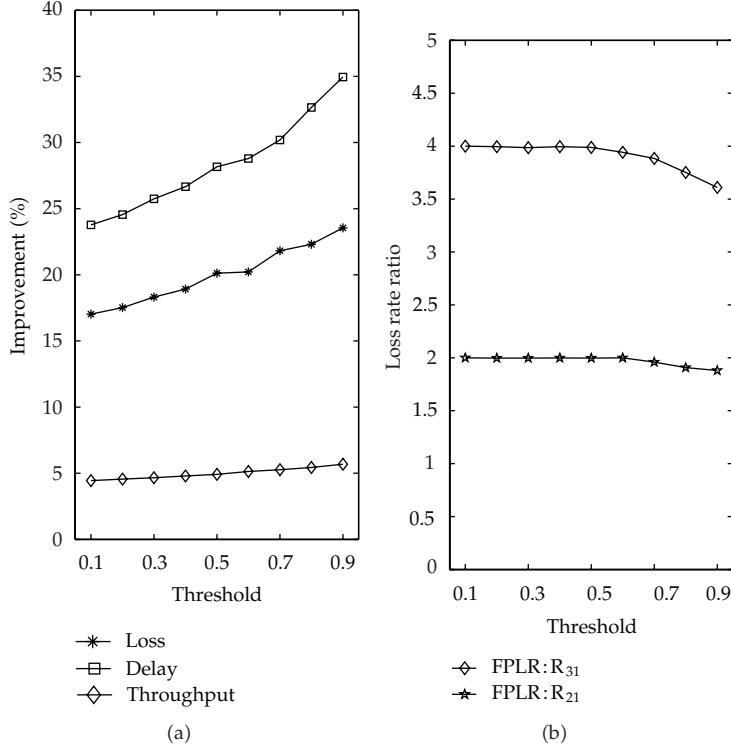


Figure 9: The effect of the state transition rate.

loss improvement, total throughput improvement, and total delay improvement. Figure 8(a) shows that the number of mobile hosts does not affect the loss rate ratios achieved by two droppers. Figure 8(b) reveals that the loss improvement, throughput improvement, and delay improvement made by FPLR increase as the number of mobile hosts increases. When there are only a few mobile hosts, for FPLR, few HOL packets destined to poor channels can be selected as a substitution, so that the improvements are not high. As the number of mobile hosts increases, since more small packets transmitted via poor channels can be chosen, the improvements made by FPLR raise.

#### 4.5. State Transition Rate

Given that a channel varies among the five states, 0, 1 Mbps, 2 Mbps, 5.5 Mbps, and 11 Mbps, the transition rate,  $a$ , is varied from 20 to 40 to observe its effect upon loss rate ratios and all improvements. Figure 9(a) shows that the state transition rate does not affect the loss rate ratios. From Figure 9(b), as state transition rate changes faster, all improvements of FPLR become smaller. When the state transition rate is small, the bad channel will last long, causing that the HOL blocking also last long. Many packets will be accumulated in the buffer, and packet losses usually occur, leading FPLR to have high opportunities to adopt the fuzzy mechanism. When the state transition rate becomes larger, the packets caused by the HOL blocking can be absorbed in the buffer, implying a smaller loss rate. Thus, the improvements made by FPLR also become smaller.



**Figure 10:** The effect of threshold  $\theta$ .

#### 4.6. Threshold $\theta$

Herein we investigate the effects of the parameter  $\theta$  for FPLR. Figure 10(a) reveals that loss improvement, throughput improvement, and delay improvement made by FPLR increase as threshold  $\theta$  increases. As the threshold  $\theta$  increases, since the opportunity of considering packet size and channel states becomes large, smaller packets transmitted via low-capacity channels are dropped, and more large packets transmitted via high-capacity channels are kept. Thus the improvements made by FPLR increase. However, Figure 10(b) shows when threshold  $\theta$  exceeds 0.5, the target loss rate ratios slide down as threshold  $\theta$  increases. The reason is that in this case the loss rate proportions have minor influence, so the target loss rate proportions cannot be maintained.

#### 5. Conclusions

This paper proposed a high-performance algorithm, FPLR, which can be used to provide proportional loss differentiation in a wireless network with multi-state channel. FPLR uses three processes to decide how to drop packets. A fuzzy controller system on the first process takes advantage of fuzzy theory to PLD. The debt decision module in the second process is to compensate for the loss rate debts. The third process is to determine an appropriate packet to drop. FPLR not only considers the normalized average loss rate, but also considers the channel state and packet sizes, in order to improve the performance of dropping.

From the simulation results, FPLR is examined to deal with location-dependent channel capacity well and does provide more accurate proportional loss rate differentiation than WPLR. Compared with WPLR, FPLR indeed provides better throughput and lower queuing delay and loss in the wireless network with a multistate channel.

Our future works include developing a wireless proportional scheduler to provide both proportional loss differentiation and delay differentiation.

## References

- [1] R. Braden, D. Clark, and S. Shenker, "Integrated services in the internet architecture: an overview," IETF RFC 1633, June 1994.
- [2] S. Blake, D. Black, M. Carlson, Z. Wang, and W. Weiss, "An architecture for differentiated services," IETF RFC 2475, December 1998.
- [3] C. Dovrolis, D. Stiliadis, and P. Ramanathan, "Proportional differentiated services: delay differentiation and packet scheduling," *ACM SIGCOMM Computer Communication Review*, vol. 29, no. 4, pp. 109–120, 1999.
- [4] C. Dovrolis, D. Stiliadis, and P. Ramanathan, "Proportional differentiated services: delay differentiation and packet scheduling," *IEEE/ACM Transactions on Networking*, vol. 10, no. 1, pp. 12–26, 2002.
- [5] C. Dovrolis and P. Ramanathan, "Proportional differentiated services, part II: loss rate differentiation and packet dropping," in *Proceedings of the IEEE/IFIP International Workshop on Quality of Service (IWQoS '00)*, pp. 52–61, June 2000.
- [6] U. Bobin, A. Jonsson, and O. Schelen, "On creating proportional loss-rate differentiation: predictability and performance," *Lecture Notes in Computer Science*, vol. 2092, pp. 372–379, 2001.
- [7] J. Zeng and N. Ansari, "An enhanced dropping scheme for proportional differentiated services," in *Proceedings of the International Conference on Communications (ICC '03)*, pp. 11–15, 2003.
- [8] J. Awuya, M. Ouellette, and D. Y. Montuno, "Weighted proportional loss rate differentiation of TCP traffic," *International Journal of Network Management*, vol. 14, no. 4, pp. 257–272, 2004.
- [9] M. Zhang, J. Wu, C. Lin, and K. Xu, "WSAP: provide loss rate differentiation with active queue management," in *Proceedings of the International Conference on Communication Technology (ICCT '03)*, pp. 385–391, July 2003.
- [10] X. Zhou, D. Ippoliti, and T. Boult, "HPPD: a hop-count probabilistic packet dropper," in *Proceedings of the IEEE International Conference on Communications (ICC '06)*, pp. 116–121, July 2006.
- [11] Y. Cao and V. O. K. Li, "Scheduling algorithms in broad-band wireless networks," *Proceedings of the IEEE*, vol. 89, no. 1, pp. 76–87, 2001.
- [12] E. H. Mamdani, "Application of fuzzy algorithms for simple dynamic plant," *Proceedings of the Institution of Electrical Engineers*, vol. 121, no. 12, pp. 1585–1588, 1974.
- [13] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, vol. 7, no. 1, pp. 1–13, 1975.
- [14] L. Zadeh, "Soft computing and fuzzy logic," *IEEE Software*, vol. 11, no. 6, pp. 48–56, 1994.

*Research Article*

## **A Cost-Effective Planning Graph Approach for Large-Scale Web Service Composition**

**Szu-Yin Lin,<sup>1</sup> Guan-Ting Lin,<sup>1</sup> Kuo-Ming Chao,<sup>2</sup> and Chi-Chun Lo<sup>1</sup>**

<sup>1</sup> Institute of Information Management, National Chiao Tung University, Hsin-Chu 300, Taiwan

<sup>2</sup> Faculty of Computing and Engineering, Coventry University, Coventry CV1 5FB, UK

Correspondence should be addressed to Szu-Yin Lin, szuyinlin@gmail.com

Received 16 November 2011; Revised 25 January 2012; Accepted 28 January 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Szu-Yin Lin 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.

Web Service Composition (WSC) problems can be considered as a service matching problem, which means that the output parameters of a Web service can be used as inputs of another one. However, when a very large number of Web services are deployed in the environment, the service composition has become sophisticated and complicated process. In this study, we proposed a novel cost-effective Web service composition mechanism. It utilizes planning graph based on backward search algorithm to find multiple feasible solutions and recommends a best composition solution according to the lowest service cost. In other words, the proposed approach is a goal-driven mechanism, which can recommend the approximate solutions, but it consumes fewer amounts of Web services and less nested levels of composite service. Finally, we implement a simulation platform to validate the proposed cost-effective planning graph mechanism in large-scale Web services environment. The simulation results show that our proposed algorithm based on the backward planning graph has reduced by 94% service cost in three different environments of service composition that is compared with other existing service composition approaches which are based on a forward planning graph.

### **1. Introduction**

Research on Web Service Composition (WSC) has become increasingly important in recent years due to the growing number of web services over the Internet and the challenge of automating the process. Particularly with the development of cloud computing, there will be more and more diverse Web services deployed and published on cloud environments. Web services are Internet-based software components which have the capabilities of delivering service cross-platforms and languages. The W3C organization has defined “Web Services” that “a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically

Web Services Description Language WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards [1]". The W3C has also pointed out that "We can identify two major classes of Web services, REST-compliant Web services, in which the primary purpose of the service is to manipulate XML representations of Web resources using a uniform set of stateless operations; and arbitrary Web services, in which the service may expose an arbitrary set of operations [2]".

Since the growth of Web services to a large number is happening and possible interactions among them are huge, searching, analyzing, and processing them to find the required services to achieve user goals is very difficult via a manual process. This also means service composition problem has become increasingly sophisticated and complicated in the real world [3]. Therefore, the issue of finding solutions efficiently via composing services to form a complex composed service is one of the important studies.

The process of combining and linking existing Web services to create new service is known as WSC. In other words, WSC problems can be considered as a service matching problem, which means that the output parameters of a Web service can be used as inputs of another Web service. The aim of WSC is to provide a means for composing diverse Web services to accomplish user request which cannot be satisfied by a single Web service. The Web services composition approaches can be broadly classified as static or dynamic based on the process and the way of composing services. A static Web service composition is constructed to solve the particular problem through manually identifying Web services by their capabilities. They are composed by a series of known Web services and a set of known data in order to obtain the expected results. Dynamic Web service composition is to automatically select Web services and compose those at the execution/run time. The aim is to build and utilize an automated service discovery and its associated execution mechanism to produce the required composite services. There have been numerous methods proposed for solving the problem of service composition, such as workflow [4] and AI planning [5]. The Web service composition is commonly described by using the Web Services Business Process Execution Language (BPEL) [6] which is an XML-based language that provides particular functionalities for processes, such as define variables, create conditionals, design loops, and handle exception. It utilizes Web services as the model for the decomposition and the composition of the process. However, BPEL promotes the development of workflow and the integration of business processes.

Nowadays, numerous researches focus on finding and developing new approaches to fit in with WSC. The task of WSC usually assumed that the composition process generates a composition workflow, which starts from the known variables from the requirements or the related constraints to the expected goal. Therefore, many algorithms based on AI planning techniques that can facilitate to automat Web service composition have been proposed [3, 5, 7–9], but it is still a great challenge for solving large-scale WSC problem to obtain multiple flexible service composition solutions with acceptable service cost and execution time. It can assume that Web services as actions, and the process of composing them to produce the desired result as planning, so planning graph is one of the most suitable techniques could be used for WSC problem. However, there are very few studies using planning graph approach to achieve WSC problem, especially with considering both sides of cost and effectiveness in large-scale Web services composition.

Therefore, this paper proposes a new cost-effective planning graph approach based on backward strategy for large-scale Web service composition on cloud environment, which can find multiple solutions and recommend a list of best composite services composition to

users. In addition, we can recommend the approximate match services which may not totally meet to user requests, but the user may accept the services and it uses fewer amounts of Web services and less nested levels of composite Web services. The main research objectives in this paper are (1) to present a novel framework and composition processes for WSC on cloud environment, (2) to design a cost-effective WSC algorithm which can obtain multiple service composition solution using fewer number of Web services with low cost and in acceptable execution time, (3) the proposed approach must process large-scale Web services which amount over 10000, and (4) to provide an approximate solution when there is no composite solution which exactly corresponds to the request.

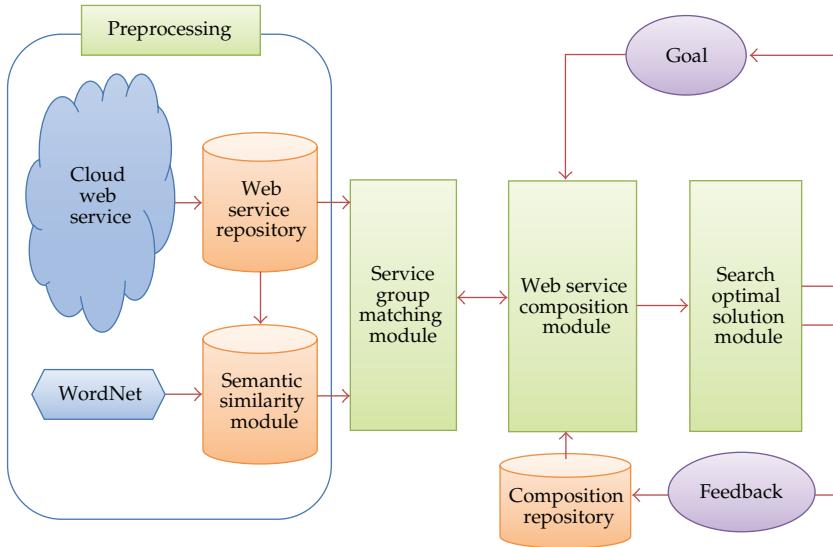
The rest of this paper is structured as follows. Section 2 describes the related works. Section 3 proposes the planning graph service composition algorithm based on the backward strategy. Section 3 presents the details of experiment and its results, and we give a summary discussion about the result. Finally, Section 4 concludes this study and proposes the future work.

## 2. Related Works

The planning graph, which is a representation technique by AI planning, provides a very powerful search technique in a space [10] to improve the efficiency of AI algorithms. A WSC problem can be modeled as a planning graph. The input parameters of the composition request are mapped to the initial state, and the output parameters of the composition request are mapped to goal propositions. If a planning graph reaches a proposition level which contains all required parameters, then it searches backward from the last level of the graph for a solution. However, the disadvantage of this approach is the difficulty of designing a strategy to trade off two key criteria that are cost and effectiveness. Therefore, there are always two problems of time consuming and redundant actions existed in the solution.

The planning graph is a layered graph whose edges are only allowed to connect two nodes from one layer to next layer. And the planning graph's layers are with an alternating sequence of action layer and proposition layer. The proposition layer contains a finite set of states, and the action layer contains a finite set of actions (the action has preconditions, negative effects, and positive effects). For example, the first layer of planning graph, P0, is a proposition layer which contains the initial states of the planning problem. The next layer, A1, is an action layer which contains a set of actions which preconditions can be satisfied by P0, and P1 is the union of the states of P0 and the effects of all A1's actions. Those preconditions of actions in A1 are connected to the state nodes in P0 by incoming arcs, and those positive or negative effects in P1 are connected to the state nodes in P1 by outgoing arcs. The process continues until it reaches the goal states or the fixed-point level of the graph.

The study conducted by [10] showed a Planning Graph planner technique called GRAPHPLAN. The GRAPHPLAN algorithm is operated in two main steps which alternate within a loop: graph expansion and solution extraction. The solution extraction can be formulated as a constraint solving problem [11] or as a search problem [12]. In Peer's survey [9], GRAPHPLAN's advantages include good performance, soundness, completeness, generation of shortest plans, and termination on unsolvable problems. However, the original GRAPHPLAN algorithm has some limitations: (1) its representation language is restricted to pure STRIPS operators, so no conditional or universally quantified effects are allowed; (2) the performance can decrease drastically if too much irrelevant information is contained in the specification of a planning task [13].



**Figure 1:** The flow diagram of the service composition mechanism.

Zheng and Yan [7] also transformed the problem of service composition into the problem of simplified planning graph based on forward search, which could be constructed in polynomial time. In classical AI planning technique, generating final solutions by a backward search is a popular approach, but it is the most time consuming technique. Researchers have been working on it to improve it. However, forward search could improve efficiency, but the redundant Web services during the construction of the planning graph lead to the increase of service cost. Zheng and Yan [7] put efforts into using forward search in planning graph algorithm to solve WSC problem, and it shows a good result which can find a solution in polynomial time but encounters some drawbacks: (1) there are many redundant Web services existed in the solution of service composition, and (2) it is lack of flexible search mechanism which can recommend multiple solutions for service composition when few input unknown parameters occur. In other words, the composition algorithm based on the forward strategy aims to minimize the search time, but there are many possible redundant and unnecessary Web services included in the final solution.

### **3. Backward Planning Graph Approach for Web Services Composition**

In this section, we will introduce the proposed service composition mechanism which includes four steps, such as preprocessing, service group matching, service composition, and search optimal solution. Those modules will be described in following subsections.

The overview and its flow diagram of the proposed service composition mechanism are shown in Figure 1, which contains the following four main processes and modules

- (I) Preprocessing. There are two components involved in the first step. One is the Web service repository, and the other is semantic similarity module. The Web Service Repository will search Web services from distributed UDDIs on cloud and store those services in a repository database and entries in the repository will be updated regularly. Therefore, the input of this mechanism is Cloud Web Services. Semantic

Similarity Module precalculates the semantic similarity values between any two concepts and stores the similarity values in a semantic similarity database for retrieval.

- (II) Service group matching module. It utilizes Web Service Repository and Semantic Similarity Module to select the Web services that can satisfy the query, and group them based on the degree of their similarity. It will provide a set of service groups for service composition.
- (III) Web Service Composition Module. It will query Service Group Matching Module to get services which are required by composition algorithm. Web Service Composition Module will generate multiple service composition solutions according to the goal which is described in final output parameters of each solution. With the expansion of levels in backward planning graph composition algorithm, the goal will be refined at each iteration.
- (IV) Search Optimal Solution Module: It will calculate the score of each solution and choose the most suitable solution of WSC from these identified solutions according to the given goal.

### **3.1. Preprocessing**

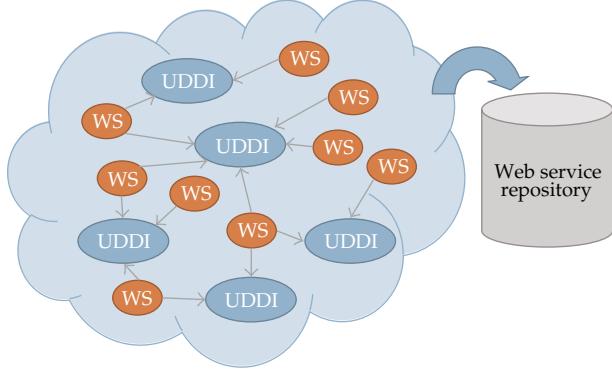
In large-scale WSC, the number of querying services could be large. Querying Web service entries registered in distributed UDDIs at runtime in process of service composition, the efficiency is likely lower than those entries stored in one centralized database. So, all Web service entries to be used in this approach will be stored to a centralized structured Web Service Repository. In addition, the calculation of semantic similarity between concepts is a time consuming task which is not efficient to meet dynamic service composition, so it will be preprocessed by Semantic Similarity Module. Service Group Matching Model according to the repository and the relationships of concept similarity to respond the query. The preprocessing task requires two components.

#### *(I) Web Service Repository*

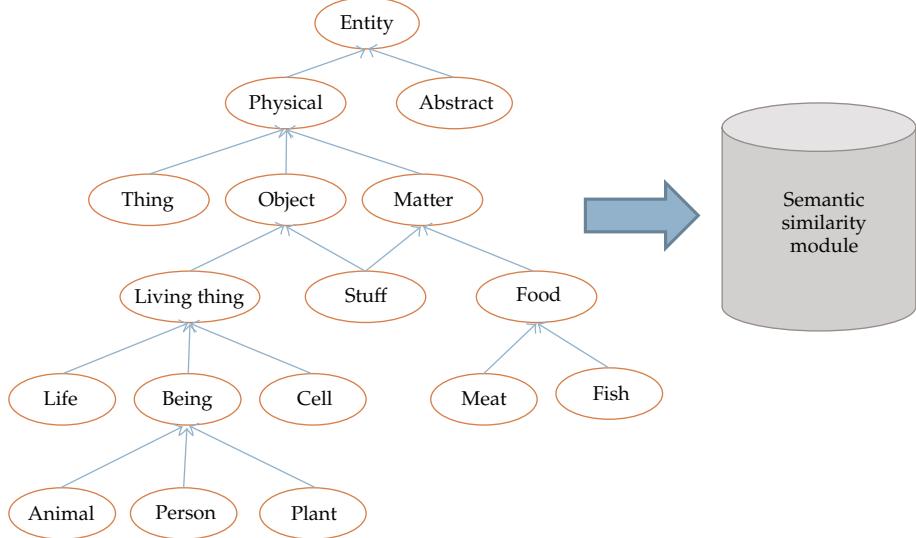
The aim of service repository is to virtualize service discovery. We query Web services registered in distributed UDDIs on cloud and parse the WSDL of Web services to store them in a repository database, as shown in Figure 2. It will search regularly Web services from distributed UDDIs and analyze the structure to update database. The structure is to facilitate the process of service composition by including a set of input parameters, output parameters, and their associated service name.

#### *(II) Semantic Similarity Module*

The semantic module is for discovering the relationship between Web services. According to the definition of lexicon and classification on WordNet, it transforms the description of services into the concept and relationship of Ontology, as shown in Figure 3. The similarities between concepts can be calculated based on their semantic similarity. Through these functions, obtaining semantic similarity values between semantic concepts become possible. The values of similarity will fall between 0 and 1, and the higher value represents higher similarity. Those similarity values are precalculated and stored in a database. However, this



**Figure 2:** Capture web services into repository from cloud.

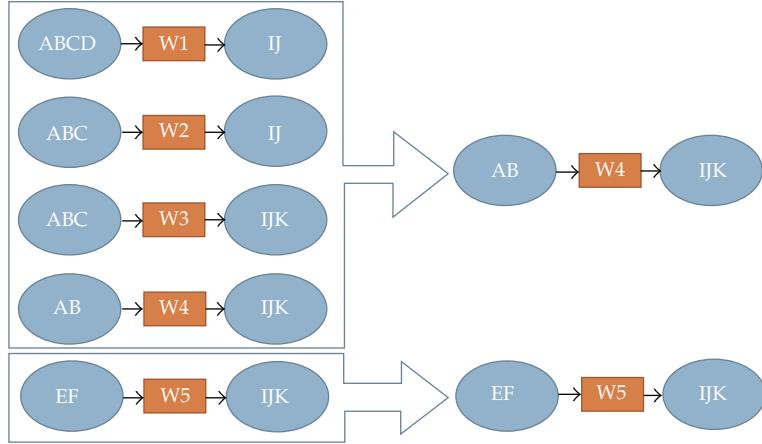


**Figure 3:** Import semantic concepts from WordNet.

module is useful but optional in the proposed service composition approach, so that it is not introduced to experiments in Section 4.

### 3.2. Service Group Matching Module

There will be full of many similar Web services on cloud, due to rapid expansion of solution space, so we could group similar Web services together based on their semantics as a service group. “Service group” is a concept that we proposed in this algorithm, which means that a group of Web services have certain degree of similarity in their input parameters and output parameters. This module will provide appropriate extracted service groups according to system requests or user queries for service composition module. It will be the key to reduce the amount of system queries and calculation. Service group matching algorithm includes three steps.



**Figure 4:** The example of service group extraction.

#### (I) Semantic Parameter Expansion

Semantic expansion is based on Sematic Similarity Module (SSM), which records relationships between semantic concepts. Querying the SSM according to the request will get a set of concepts which can meet the request. Then, the set of parameters can be used to select the Web services which satisfy the query and group them based on the degree of their similarity.

#### (II) Query Web Service Repository

From the previous step, we have a set of parameterized queries. Using them to query the database by matching services in the repository which output parameters can provide one of query parameter sets. It is assumed that there exists at least one service that can produce the expected output.

#### (III) Extract Web Service Group

Those Web services that are collected from the previous step will be classified into different groups and each group can be represented by one service. The extraction rule of service group is “ $\text{effect}(w) \subseteq \text{effect}(wg) \wedge \text{precond}(w) \supseteq \text{precond}(wg)$ .” The effect ( $w$ ) and precond ( $w$ ) mean the output and input results of Web services, respectively. For example, in Figure 4, there are five services, W3 has input parameters {A, B, C} and output parameters {I, J, K}. W4 has input parameters {A, B} and output parameters {I, J, K}. W4 uses fewer inputs and gets the same outputs, and then it concludes that W4 contains W3, and so on. It helps to reduce the search space of solutions.

### 3.3. Web Service Composition Module

In the proposed algorithm, a planning graph approach based on the backward strategy is adopted to solve the problem of large search space. The aim of a backward search is to find the initial states from end or intermediate states, so we propose an algorithm for solution

```

 $i \leftarrow 0, P_0 \leftarrow s_0, G \leftarrow P_0$ 
repeat
     $G \leftarrow \text{Expand Based On Backward } (G)$ 
     $S \leftarrow \text{Extract Solutions } (G, S)$ 
     $S \leftarrow \text{Reduce Solutions } (S)$ 
     $i = i + 1$ 
     $\text{Validate Solution } (G, S)$ 
     $\text{Compute Score } (G, S)$ 
     $\text{Output } (\text{Search Solution } (S))$ 

```

**Algorithm 1**

extraction from a planning graph, which help to find the initial state. The main composition algorithm is shown as follows.

### 3.3.1. Algorithm Compose $(G, S)$

$G = \langle P_0, A_1, P_1, \dots, A_i, P_i \rangle$  is a simplified planning graph.

$S = \{s_1, \dots, s_n\}$  is a set of solution candidates (Algorithm 1).

Web Service Composition Module includes the following four steps.

#### (I) Expand the Planning Graph

In this step, we will expand the planning graph with more action level for backward search. From the last proposition level in the planning graph, a list of expected parameters can be obtained, and then Service Group Matching Module can be interrogated to get service groups. Those service groups can be added to a new action level and arrange a new proposition level.

### 3.3.2. Algorithm Expand Based on Backward $(G)$

$G = \langle P_0, A_1, P_1, \dots, A_i, P_i \rangle$  is a simplified planning graph.

$W = \{w_1, \dots, w_n\}$  is a set of web services.

$WG = \{wg_1, \dots, wg_n\}$  is a set of web service groups (Algorithm 2).

#### (II) Extract Solutions from the Planning Graph

After the previous step, we can get a planning graph  $G(P_0, A_1, P_1, \dots, A_i, P_i)$  which contains action and proposition levels. In the solution extraction process, we have to trace to obtain possible solutions from the planning graph layer by layer, so keep those lists of service composition first and then expand them to the next new layer in the planning graph. In other words, we find out the service combinations to extend the service composition solutions from the action level. This step is for finding feasible composition solutions which correspond to the initial state of the request according to the planning graph established in previous steps.

```

for  $w \in W$  do
    for  $wg \in WG$  do
        if effect ( $w$ )  $\subseteq$  effect ( $wg$ )  $\wedge$  precond ( $w$ )  $\supseteq$  precond ( $wg$ ) then
             $wg \leftarrow wg \cup w$ 
        if effect ( $w$ )  $\supseteq$  effect ( $wg$ )  $\wedge$  precond ( $w$ )  $\subseteq$  precond ( $wg$ ) then
             $wg \leftarrow wg \cup w$ 
            effect ( $wg$ )  $\leftarrow$  effect ( $w$ )
            precond ( $wg$ )  $\leftarrow$  precond ( $w$ )
        if not be filled with service group then
             $WG \leftarrow WG \cup$  new  $wg$ 
     $A_{i+1} \leftarrow WG$ 
return  $G$ 

```

**Algorithm 2**

```

for  $s \in S$  do
    for  $a \in A_i$  do
        available ( $a$ )  $\leftarrow$  true
         $S \leftarrow S - s$ 
    do
        required  $\leftarrow$  inputs ( $s$ )
        ns  $\leftarrow$  new solution
        parent (ns)  $\leftarrow s$ 
        for  $a \in A_i$  do
            if available ( $a$ )  $\wedge$  required  $\cap$  effect ( $a$ )  $\neq$  NULL then
                required  $\leftarrow$  required  $-$  (required  $\cap$  effect ( $a$ ))
                available ( $a$ )  $\leftarrow$  false
                ns  $\leftarrow$  ns  $\cup$   $a$ 
            if required = NULL then
                 $S \leftarrow S \cup ns$ 
                break
            while (required = NULL)
        return  $S$ 

```

**Algorithm 3**

### 3.3.3. Algorithm Extract Solutions ( $G, S$ )

$S = \{s_1, \dots, s_n\}$  is a set of solution candidates.

inputs ( $s$ ): a set of input parameters of solution  $s$ .

parent ( $s$ ): the parent node of solution  $s$ .

available ( $a$ ): it records action  $a$  whether available or not (Algorithm 3).

### (III) Reduce Solutions

From the above step, it extracts possible solutions to form a set of service compositions. Two strategies in this research can be used to select the most appropriate solution. One of the strategies is that removing the solutions which utilize the number of services more than the threshold given in any new expansive level. “Service Threshold” means the max number of

```

for  $s \in S$  do
    if count ( $s$ ) > Service Threshold then
         $S \leftarrow S - s$ 
for  $s \in S$  do
    if initial ( $s$ ) ⊃ {initial ( $s_2$ ) |  $s_2 \in S$ } then
         $S \leftarrow S - s$ 
return  $S$ 

```

**Algorithm 4**

services used in any composition solution. It is the key to reduce redundant solutions and facilitate the efficiency of composition algorithm. It can be determined by users according to their server performance. The Service Threshold value is set to 3000 by default value which is very huge in the experiments. The other is to remove the solutions which have too many services and similar to other short solutions in each action level. The process helps to filter large number of unwanted solutions and identified the most appropriate ones.

### 3.3.4. Algorithm Reduce Solutions ( $S$ )

initial ( $s$ ): a set of initial input parameters of solution  $s$ .

count ( $s$ ): a number of web services of solution  $s$ .

Service Threshold: a number of max services used in one solution (Algorithm 4).

### (IV) Validate and Score Solutions

In order to filter out less appropriate solutions, a threshold value is given in this step to remove solutions with lower precision. “Threshold” means the minimum tolerable precision of output and input parameters matching results of Web services. It can also be described as the threshold of solution precision. Users can set the value to 1 when they want to obtain the completely matching solutions, otherwise, decrease this threshold value to allow unmatched but possible solutions. In addition, we repeat the above steps to expand the planning graph for finding all solutions and then calculate the score for each solution to find the best one. If the process stops in this step by threshold or finishes complete planning graph but still without solution, it means that there is no final solution found in this composition problem. The validate algorithm is shown as below, and the score approach for search optimal solution is in the Section 3.4.

### 3.3.5. Algorithm Validate Solution ( $G, S$ )

Initial ( $s$ ): a set of initial input parameters of solution  $s$ .

precise ( $s$ ): the precise of solution  $s$  that correspond to user request.

intersection: the amount of all same concepts.

union: the amount of all different concepts.

Precise Threshold: The threshold of solution precision (Algorithm 5).

Here is an example to explain the proposed composition mechanism. Assume a user’s request which includes a set of input parameter  $r_{in} = \{A, B, C, D\}$  and a set of output

```

for  $s \in S$  do
    intersection  $\leftarrow$  Count (initial ( $s$ )  $\cap$  initial ( $g$ ))
    union  $\leftarrow$  Count (initial ( $s$ )  $\cup$  initial ( $g$ ))
    precise ( $s$ )  $\leftarrow$  intersection/union
if Precise Threshold  $\leq$  precise ( $s$ ) then
    Search Solution ( $S$ )  $\leftarrow$  Search Solution ( $S$ )  $\cup$   $s$ 

```

**Algorithm 5****Table 1:** The example of Web services.

Web service	Input parameters	Output parameter
W1	A, B, C	E, F
W2	A, B	H
W3	D, E	I
W4	E, F	J
W5		K, L
W6	G	L
W7	H	M
W8	I, J, K	M, N
W9	L	N

parameters  $r_{\text{out}} = \{M, N\}$ , and there are nine Web services in our Web Service Repository. Table 1 shows the details of the example of Web Service Repository.

Figure 5 shows the expanded planning graph result of the above given example.  $\{A, B, C, D\}$  and  $\{M, N\}$  are the input and output parameters of the composition request. At first, we search Web services which can output  $\{M, N\}$ , then we get  $\{w_7, w_8, w_9\}$  which can support the proposition 4 (P4), our goal. Those three Web services will be involved in action 3 (A3). We collect input parameters of Web service in action 3 (A3), and we will get proposition 3 (P3). The rest of proposition and action are like this, and so forth. From the previous step, a planning graph is generated to extract solutions. We utilize our proposed algorithm to extract solutions which store them in a tree structure to form a solution tree for tracing. Every leaf node in solution tree means that there is a solution from leaf node to root. The result is shown in Figure 6. Many possible paths that can reach initial state of the tree have been discovered. This is one of advantages of adopting the backward strategy, so that multiple solutions for user request can be found.

$\{M, N\}$  are the output parameters of user request. It is located in proposition 4 (P4), so we need to find the combinations of Web services in action 3 (A3), which corresponds to  $\{M, N\}$ . And we will get two combinations, which are  $\{w_7, w_9\}$  and  $\{w_8\}$ . Those combinations will be added to the root as its child. Now, there are two nodes at second level. The solution node composed by  $\{w_7, w_9\}$  requires a set of input parameters  $\{H, L\}$ , so we need to find the combinations of Web services in action 2 (A2), which can correspond to  $\{H, L\}$ . So we get  $\{w_2, w_5\}$  and  $\{w_2, w_6\}$ , and so forth.

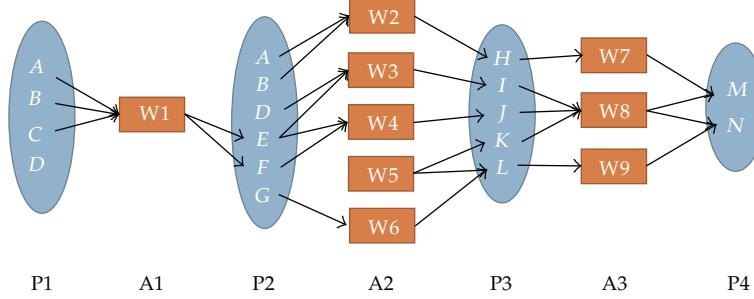


Figure 5: The simplified planning graph for the above example.

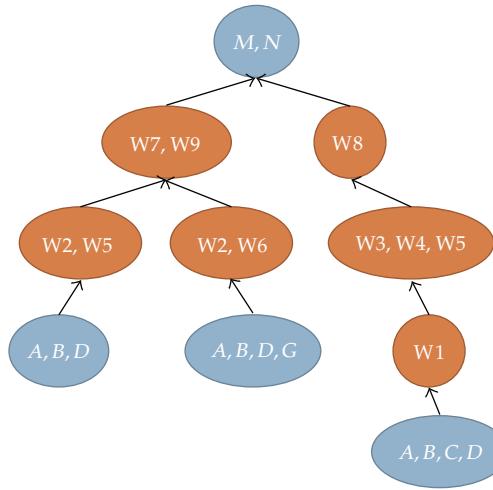


Figure 6: The solution tree for the planning graph for the above example.

### 3.4. Search Optimal Solution Module

After establishing the solution tree, it found a number of service composition solutions which possibly satisfy user request. Then, the Search Optimal Solution Module needs to give a score on each solution and select the highest score one. At first, we calculate the precision of the matched solutions with user request. We utilize the ratio of the number of intersections to the number of unions, which are between the solution's initial states and the user request's inputs. The precision equation is in the following.

$$\text{Precise } (s_1, s_2) = \frac{\cap s_1 s_2}{\cup s_1 s_2}. \quad (3.1)$$

Equation (3.1) shows the precision. In this equation,  $\cap s_1 s_2$  represents the amount of the same concepts and  $\cup s_1 s_2$  represents the amount of all different concepts, where  $s_1$  and  $s_2$  both are lists of concepts. This equation evaluates the similarity between the solution's initial states and the user request's inputs. It also means the difference between two lists of concepts. After the previous step, we have the likelihood of the solutions to achieve the goal.

For calculation of the score for each solution, we need to calculate matching degree between the levels of the solutions. The matching equation is illustrated in the following

$$\begin{aligned} \text{Mat}(s_1, s_2) &= (\text{KM} - \alpha(\cup s_1 s_2 - \cap s_1 s_2)) \\ \text{where } \text{KM} &= \text{KM} \langle \text{Sim}(c_1, c_2) \rangle \quad (c_1 \in s_1, c_2 \in s_2). \end{aligned} \quad (3.2)$$

Equation (3.2) shows the matching score. In the above equation, KM represents classical Kuhn-Munkres algorithm which solved the assignment problem, and Sim represents the similarity between any two concepts in  $s_1$  and  $s_2$ . This equation is to evaluate the matching score of two solutions. With the previous two formulas, we can calculate the solution score. It sums the matching scores between levels of the solutions and divides by the number of levels to gain the average. Then, we get the average of matching scores, and multiple by the matching precision of the solution and the request. The score equation is shown as below.

$$\text{Score}(\text{slu}) = \text{Precise}(\text{slu} \cdot \text{in}, r \cdot \text{in}) \frac{\sum_{s_1, s_2}^{\text{slu}} \text{Mat}(s_1, s_2)}{n}. \quad (3.3)$$

Equation (3.3) shows the solution score. Solution slu is a list of nodes from node leaf to the root, which represents each level of service composition,  $\text{slu} \cdot \text{in}$  represents the input parameters of the solution slu,  $r \cdot \text{in}$  represents the input parameters of the request  $r$ , and the number of levels is represented by  $n$ . The Precision calculates the matching precision between the input parameters of the request  $r$  and the input parameters of the solution slu.

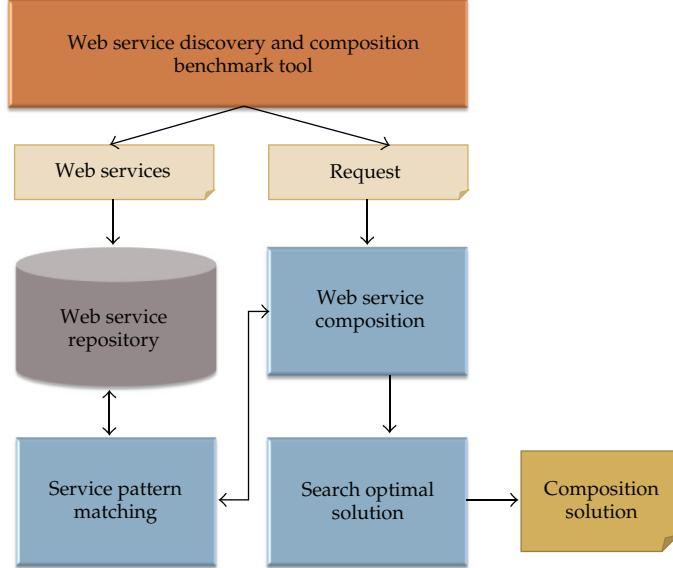
Briefly, our proposed mechanism is particularly suitable for large-scale service composition due to the problems that service composition is an error-prone and complicated process. In this study, we designed a cost-effective planning graph mechanism for finding multiple service composition solutions with lower service cost to overcome the above problems. Furthermore, in some cases, the exact solution does not exist, and our proposed mechanism still can recommend approximate solutions. On the other hand, in order to provide multiple solutions, it must need to trace each possible solution. Our proposed mechanism can filter out and reduce the less appropriate solutions in order to avoid exponential growth in complexity.

## 4. Experiments

In this section, the simulation design, assumption, performance metrics, and simulation results are described. Moreover, there is a brief discussion in the later section.

### 4.1. Simulation Design

We established a simulation platform which is based on the proposed mechanism for validating our algorithm. In this platform, WSBen [14], which is widely used to evaluate the efficiency and effectiveness in Web service discovery and composition, is utilized to generate different test sets to validate the proposed algorithm. The simulation platform will carry out the algorithm according to the test dataset and the requests derived from WSBen.



**Figure 7:** The architecture of simulation platform.

The architecture of simulation platform is shown in Figure 7. WSBen was set up with the test data including a set of Web services and a set of feasible requests for testing WSC algorithm. In the simulation, a program extracts the information of Web service generated by WSBen and then stores it in the Web Service Repository which includes service name, input parameters, and output parameters. The Service Group Matching Module is in charge of Web services selection according to the service query which has been processed by the Web Service Composition Module. In addition, it also groups the selected Web services as Web service groups for the composition algorithm. The Web Service Composition Module including a composition algorithm composes a composite service according to the composition request, which interacts with the service matching module in the discovery and selection process until a possible solution had been found, if there is any. Therefore, it will generate a list of candidate composite Web services which can fulfill the required services. The Search Optimal Solution Module calculates each candidate solution to obtain a score in order to generate an ordered recommendation list for selection.

#### 4.2. Simulation Assumption

WSBen is a Web services generation and benchmark tool for Web services composition and discovery, which provides a set of functions to simplify the generation of Web service test datasets and builds test environments including the testing requests [14]. A complex graph network with nontrivial topological features does not occur in a simple form such as lattices or random graphs but often occur in real applications. Many systems in nature can be described by models of complex networks, which are structures consisting of nodes or vertices connected by links or edges [15]. Such as a Web service can be assumed to be a transformation between two difference domain nodes, this could be regarded as clusters of parameters. The development of WSBen is based on the above assumption. The complex

network is most commonly used in three concepts—"random," "small-world," and "scale-free" network [15]. Therefore, the tool provides these three different types of network models to generate Web services test datasets. It also can generate diverse sizes of datasets based on the complex network type specified.

The network topology will be constructed as a directed graph based on graph theory. Each node is represented as a parameter cluster, and each edge is represented as a connecting between two different clusters, that will be regarded as generation template of Web services. The development of WSBen is based on the above assumption. This test environment also includes five feasible and correct requests ( $r_1, r_2, r_3, r_4$ , and  $r_5$ ) generated by WSBen. We put them into all experimental cases as the test requests in three network models for evaluation. There is an input framework that users can specify the generated Web service and the characteristics of network topology in WSBen. The input framework  $\text{xTS} = \langle |J|, G_r, \eta, M_p, |W| \rangle$  is described as below.

- (I)  $|J|$  is the total number of parameter cluster.
- (II)  $G_r$  donates a graph model to specify the topology of parameter cluster network. The three types of network, which are "random," "small-world," and "scale-free" complex networks, can be simulated by the three network model, as follows.
  - (i) *Erdo-Renyi* ( $|J|, p$ ). The model has such a simple generation approach that it creates  $|J|$  nodes in graph and assign each edge in the graph with probability  $p$ .
  - (ii) *Newman-Watts-Strogatz* ( $|J|, k, p$ ). The initialization is a ring graph with  $k$  nodes. Each node adds to graph and constructs edge by connecting to others with probability  $p$ . The process will iterate until there are  $|J|$  nodes in the graph.
  - (iii) *Barabasi-Albert* ( $|J|, m$ ). There are  $m$  nodes with no edge in the initial graph. Each node adds with  $m$  edges, which are preferentially attached to existing nodes with high degrees.
- (III)  $\eta$  donates the parameter condense rate. Users can control the density of partial matching cases in generated Web services.
- (IV)  $M_p$  donates the minimum number of parameters in a cluster. In other words, each cluster has at least  $M_p$  parameters.
- (V)  $|W|$  donates the total number of Web services in a test dataset.

The assumptions in datasets for simulation have to be described in advance. There are three types of network in the simulation, which are random, small-world, and scale-free types. Each network is assumed as a parameterized cluster network, and a Web service is a transformation between two clusters. Each cluster contains its parameters, and it is also called node. In other words, the input and output parameters of a Web service can be generated and selected from each two domain cluster nodes according to argument  $\eta$ ,  $M_p$ , and the generation rules of WSBen, and a Web service generated in the network could be regarded as an edge. WSBen provides a set of functions to simplify the generation of test data for WSC algorithm. It generates Web services according to the parameter cluster network which users specify. In our simulation, we assume that there are 100 clusters in network and the parameter condense rate is 0.8. The configurations for three types of network model in our experiment platform are as follows.

- (I) Random Network: Barabasi-Albert (100, 0.06). The model creates 100 nodes in the graph and each edge in the graph is with probability 0.06 to be chosen.
- (II) Small-World Network: Newman-Watts-Strogatz (100, 6, 0.1). The initialization is a ring graph with 6 nodes. Each node adds to the graph and constructs edges connected to each other with probability 0.1, until there are 100 nodes in this graph
- (III) Scale-Free Network: Erdos-Reyi (100, 6). There are 6 nodes with no edge in the initial graph. Each node adds 6 edges until reach 100 nodes. Each added edge is preferentially attached to existing nodes with high degrees.

For each network, there are 10 different sizes in each of test data types, which sizes are 10,000 to 100,000, respectively. Thus, there are 30 test sets (three frameworks multiplied by ten different test sizes) in our large-scale WSC simulation.

#### **4.3. Evaluation Criteria**

Effectiveness, efficiency, and feasibility are three evaluations, which are used to test our proposed approach. We use diverse sizes of Web services and three types of Web service networks to measure the scalability and robustness of our approach. The evaluation metrics are as follows.

- (I)  $\#T$ : it measures the time that is required by the algorithm to find a fully matched or approximate solution. In other words, it is a measure of computational efficiency.
- (II)  $\#C$ : the number of Web services in a solution of WSC problem, which also stands for composite cost. It is a measure of cost and effectiveness.
- (III)  $\#L$ : the number of nested levels of composite Web services in a solution of WSC problem. It is also a measure of cost and effectiveness.
- (IV)  $\#P$ : it measures the difference of input parameters between the final solution and user request. It means that there exists a completely correct solution if the  $\#P$  rate is 100%, otherwise, the solution is an approximate answer with some missing input parameters, and user can add those missing input parameters to achieve the goal. Precision (described in (3.1)) is a measure of effectiveness.

#### **4.4. Simulation Results**

In this section, we show the efficiency, effectiveness, and feasibility of the proposed algorithm in three cases. We utilize diverse sizes of Web services and different type of network topology to observe the scalability and robustness of our proposed algorithm. Some related results are illustrated in the below sections. The three test datasets of our experiments deal with the networks of random, small, and scale-free type. We compare the proposed backward strategy with the forward strategy in the previous study [7] to observe the results. The evaluation criteria are four indexes:  $\#T$ ,  $\#C$ ,  $\#L$ , and  $\#P$  which are described in Section 4.3.

*Case 1 (Random Network).* Table 2 shows the results of five requests of random network with  $|W| = 10,000$  Web services in a test dataset. The results of criterion  $\#P$  mean that both Backward and Forward strategies can find solutions in all cases ( $\#P = 1$ ). Regarding  $\#L$ , Backward and Forward strategies have no difference in finding solutions. In the criterion of  $\#T$ , the computational efficiency of Forward is better than Backward, because of generating

**Table 2:** Results of random network with  $|W| = 10000$ .

Test request	Backward				Forward			
	#L	#C	#T	#P	#L	#C	#T	#P
r1	8	18	2.725	1	8	262	1.033	1
r2	8	14	3.028	1	8	237	1.05	1
r3	7	9	1.916	1	7	220	1.066	1
r4	7	10	2.191	1	7	245	1.072	1
r5	9	16	4.141	1	9	241	1.062	1
Avg.	<b>7.8</b>	<b>13.4</b>	<b>2.8</b>	<b>1</b>	<b>7.8</b>	<b>241</b>	<b>1.056</b>	<b>1</b>

**Table 3:** Results of small world network with  $|W| = 10000$ .

Test request	Backward				Forward			
	#L	#C	#T	#P	#L	#C	#T	#P
r1	14	14	1.599	1	14	183	0.863	1
r2	11	11	1.016	1	11	175	0.769	1
r3	12	12	1.985	1	12	156	0.746	1
r4	10	10	2.419	1	10	174	0.716	1
r5	16	16	1.854	1	16	181	0.903	1
Avg.	<b>12.6</b>	<b>12.6</b>	<b>1.776</b>	<b>1</b>	<b>12.6</b>	<b>173.8</b>	<b>0.8</b>	<b>1</b>

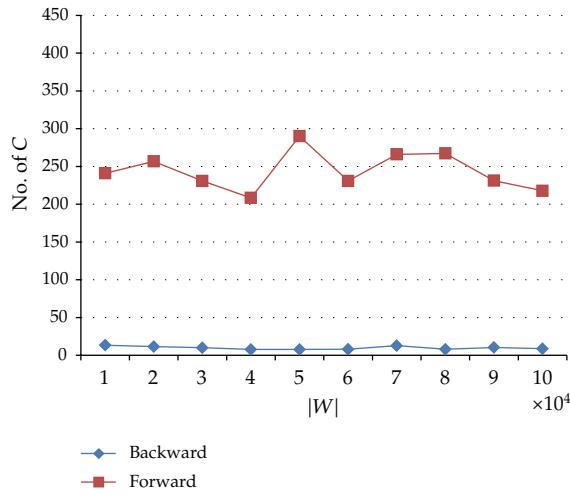
final solutions by a backward search to expand graph layers is very time consuming. It still falls within an acceptable time frame in our proposed approach even in the large-scale (10000 Web services) environment. In the experiment with #C Web services, our proposed Backward approach outperforms the Forward. Backward strategy uses less 20 services to fulfill the request in all cases, but the Forward strategy requires more than 200 Web services to find a solution.

*Case 2 (Small World Network).* Table 3 shows the results of five test requests of a small world network with  $|W| = 10,000$  Web services in a test dataset. The results of criterion #P mean that both our proposed Backward and the Forward strategies still can find solutions in all cases ( $#P = 1$ ). Regarding #L, the result of Backward strategy is as good as the Forward strategy. Our proposed Backward strategy takes more a little time than the Forward strategy in #T, but processing time is still acceptable. The reason is to generate final solutions by a backward search to expand graph layers is very time consuming. Finally, in the experiment with #C, it shows it has produced much better performance than the Forward strategy.

*Case 3 (Scale-Free Network).* Table 4 shows the result of five test requests of a scale-free network which contains  $|W| = 10,000$  Web services in a test dataset. The results of criterion #P mean that Forward strategy still can satisfy all requests ( $#P = 1$ ), but it requires more than 200 Web services to obtain the solution in service cost #C. In the more complex scale-free network, although Backward strategy finds the fully matched solution is impossible in some cases, an approximate solution can be found and replaced, which use much less services to satisfy the request. Regarding #T and #L, the result shows that Backward strategy are a bit better than Forward in #T, but both produce almost the same performance in these two criteria.

**Table 4:** Results of scale-free network with  $|W| = 10000$ .

Test request	Backward				Forward			
	#L	#C	#T	#P	#L	#C	#T	#P
r1	4	11	1.232	0.933	4	244	2.48	1
r2	4	6	3.151	1	4	343	1.654	1
r3	5	11	2.886	0.778	5	356	2.824	1
r4	—	—	—	—	4	313	1.414	1
r5	4	10	0.203	0.814	4	281	2.122	1
Avg.	4.25	9.5	1.868	0.8812	4.4	316	2.3808	1

**Figure 8:** The average cost of finding solution in random network.

As shown in Figures 8, 9, and 10, the proposed algorithm has much better usage of Web services in all cases in terms of obtaining the solution. Because of the aim of the forward algorithm, which is to reach the goal as quick as possible, it expands the search space in planning graph for services composition problem no matter how many redundant Web services are produced. However, the proposed backward algorithm has no redundant Web services existed in the solution, because its backward strategy searches what it needs to reach the initial state. Figure 10 shows average cost of searching solution in Scale-Free Network. It can be observed that the forward algorithm represents an unstable circumstance in Scale-Free Network, when the size of a test dataset becomes large. Nevertheless, our backward algorithm is still to appear stable and effective results. On average, our algorithm reduces to 94% service cost for finding the solutions. From the above experiment results, we have a simple deduction that the backward search will continue to display the stable and smooth results in different types of network topology, even if the sizes of Web services continue to increase.

In experiments, three types of network topology and diverse sizes of Web services are utilized to evaluate these two algorithms. The main findings about the proposed algorithm from the experiments are given as follows. In effectiveness, the experiment results show our proposed backward algorithm has 94% better effectiveness compared to the forward algorithm in most cases. In few cases, it uses little more levels to obtain the solution, but it

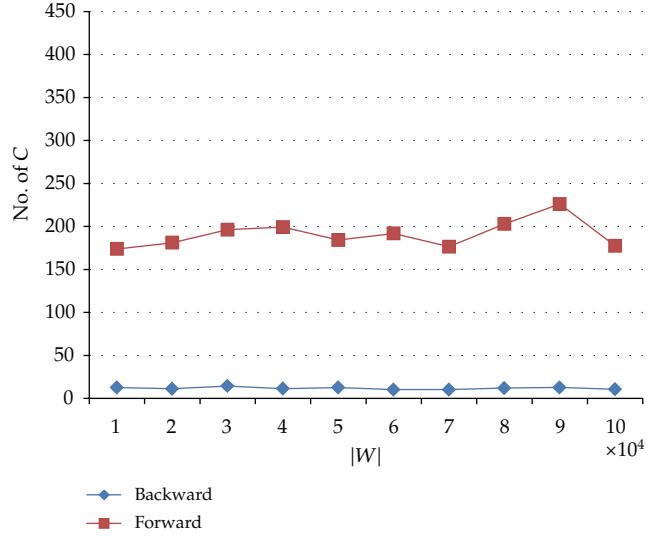


Figure 9: The average cost of finding solution in small world network.

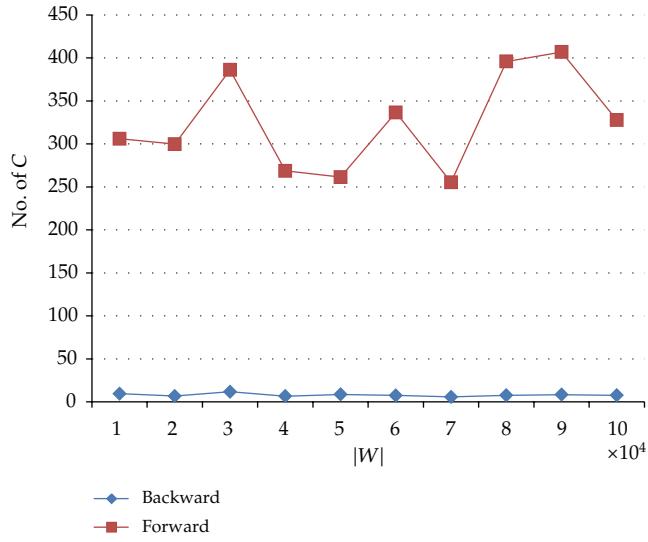


Figure 10: The average cost of finding solution in scale-free network.

can get low-cost solutions. As confirmed by the experiment results, our proposed algorithm can also get very high-precise solutions in most cases. In efficiency, although the forward algorithm has better performance than the backward algorithm, the cost of solution is very high. The proposed backward algorithm is efficient when the sizes of Web service are less than 50,000. Therefore, the effect of this cost-effective approach for large-scale WSC problem is exhibited.

## 5. Conclusion

In this study, we proposed a backward planning graph mechanism for Web service composition on cloud environment. It utilizes a planning graph based on a backward search to find multiple feasible solutions and recommends the best composite solution based on their service costs. We also validated that the proposed algorithm can improve the error-prone problem of service composition and the redundant Web service involved in large-scale service composition problem. Therefore, the algorithm based on the backward planning graph search, which is capability of recommending multiple service composition and remove the redundant services. As the experiment results in this paper, we proved that our proposed backward algorithm had a better cost-effectiveness than the forward search algorithm in terms of service cost. The proposed algorithm is able to recommend approximate solutions of service composition using very few Web services, because it has higher quality of relationships between services. In other words, we can decrease the amount of cost of Web services and remain acceptable planning graph levels and execution time.

In the future, we will study how to improve a greedy algorithm in order to expand the solution tree. To obtain right combinations is a very important issue for the design of algorithm. It cannot only help to decrease wrong combinations, but also to improve the effectiveness and efficiency of algorithm. Moreover, we can add more predictable restrictions to prune the huge combination tree nodes for our algorithm efficiency. If there are some more predictable restrictions and composition information, then that will help us to make more appropriate decisions to find the solutions. Moreover, because the Semantic Similarity Module in the proposed approach is optional, this module has been not analyzed in the experiments. There is a need to have an environment that can help to validate semantic association of service compositions. To design a semantic experiment environment should be undertaken determining how the semantic can influence the effectiveness of service composition.

## Acknowledgment

This research work was supported by the National Science Council of the Republic of China under the Grant NSC-99-2410-H-009-036-MY3. This research is carried out as a part of GREENet which was supported by a Marie Curie International Research Staff Exchange Scheme Fellowship within the 7th European Community Framework Programme under grant agreement no. 269122.

## References

- [1] H. Haas and A. Brown, "Web Services Glossary," <http://www.w3.org/TR/ws-gloss/>.
- [2] D. Booth, H. Haas, F. McCabe et al., "Web Services Architecture," <http://www.w3.org/TR/ws-arch/>.
- [3] M. Kuzu and N. K. Cicekli, "Dynamic planning approach to automated web service composition," *Applied Intelligence*, vol. 36, no. 1, pp. 1–28, 2010.
- [4] S. Wang, W. Shen, and Q. Hao, "Agent based workflow ontology for dynamic business process composition," in *Proceedings of the 9th International Conference on Computer Supported Cooperative Work in Design (CSCWD '05)*, pp. 452–457, Coventry, UK, 2005.
- [5] L. Qiu, L. Chang, F. Lin, and Z. Shi, "Context optimization of AI planning for semantic Web services composition," *Service Oriented Computing and Applications*, vol. 1, no. 2, pp. 117–128, 2007.
- [6] A. Alves, A. Arkin, S. Askary et al., "Web Services Business Process Execution Language Version 2.0," OASIS WSBPEL TC, 2007.

- [7] X. Zheng and Y. Yan, "An efficient syntactic web service composition algorithm based on the planning graph model," in *Proceedings of the IEEE International Conference on Web Services (ICWS '08)*, pp. 691–699, September 2008.
- [8] Y. Yan and X. Zheng, "A planning graph based algorithm for semantic web service composition," in *Proceedings of the IEEE Joint Conference on E-Commerce Technology (CEC '08) and Enterprise Computing, E-Commerce and E-Services (EEE '08)*, pp. 339–342, July 2008.
- [9] J. Peer, *Web Service Composition as AI Planning—a Survey*, University of St.Gallen, St. Gallen, Switzerland, 2005.
- [10] A. L. Blum and M. L. Furst, "Fast planning through planning graph analysis," *Artificial Intelligence*, vol. 90, no. 1-2, pp. 281–300, 1997.
- [11] S. Kambhampati and M. B. Do, "Planning as constraint satisfaction: Solving the planning graph by compiling it into CSP," *Artificial Intelligence*, vol. 132, no. 2, pp. 151–182, 2001.
- [12] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall Press, Upper Saddle River, NJ, USA, 2009.
- [13] Y. Dimopoulos, B. Nebel, and J. Koehler, "Encoding planning problems in nonmonotonic logic programs," in *Proceedings of the 4th European Conference on Planning: Recent Advances in AI Planning (ECP '97)*, pp. 169–181, Toulouse, France, 1997.
- [14] S. C. Oh and D. Lee, "WSBen: A web services discovery and composition benchmark toolkit," *International Journal of Web Services Research*, vol. 6, no. 1, pp. 1–19, 2009.
- [15] X. F. Wang and G. Chen, "Complex networks: Small-world, scale-free and beyond," *IEEE Circuits and Systems Magazine*, vol. 3, no. 1, pp. 6–20, 2003.

## Research Article

# An Optimal Classification Method for Biological and Medical Data

**Yao-Huei Huang,<sup>1</sup> Yu-Chien Ko,<sup>2</sup> and Hao-Chun Lu<sup>3</sup>**

<sup>1</sup> Institute of Information Management, National Chiao Tung University, Management Building 2, 1001 Ta-Hsueh Road, Hsinchu 300, Taiwan

<sup>2</sup> Department of Information Management, Chu-Hua University, No. 707, Section 2, WuFu Road, Hsinchu 300, Taiwan

<sup>3</sup> Department of Information Management, College of Management, Fu Jen Catholic University, No. 510, Jhongjheng Road, Sinjhuang, Taipei 242, Taiwan

Correspondence should be addressed to Yao-Huei Huang, yaohuei.huang@gmail.com

Received 25 October 2011; Revised 25 January 2012; Accepted 28 January 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Yao-Huei Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a union of hyperspheres by the mixed-integer nonlinear program to classify biological and medical datasets. A classifying program with nonlinear terms uses piecewise linearization technique to obtain a global optimum. The numerical examples illustrate that the proposed method can obtain the global optimum more effectively than current methods.

## 1. Introduction

Classification techniques have been widely applied in the biological and medical research domains [1–5]. Either objects classification or patterns recognition for biological and medical datasets necessarily demands an optimum accuracy for saving patients' lives. However, cancer identification with the supervised learning technique does not take a global view in identifying species or predicting survivals. The improvement should cover the whole scope to give implications instead of only considering the efficiency for diagnosis. This research aims to extract features from whole datasets in terms of induction rules.

In the given dataset with several objects, in which each object has some attributes and belongs to a specific class, classification techniques are used to find a rule of attributes that appropriately describes the features of a specified class. The techniques have been studied over the last four decades, including decision tree-based methods [6–11], hyperplane-based methods [12–14], and machine learning-based methods [14–17].

To assess the effects of these classifying techniques, three criteria are used for evaluating the quality of inducing rules based on the study of Li and Chen [3].

- (i) *Accuracy*. The rule fitting a class should not cover the objects of other classes. The accuracy of a rule should be the higher the better.
- (ii) *Support*. A good rule of fitting a class should be supported by most of the objects of the same class.
- (iii) *Compact*. A good rule should be expressed in a compact way. That is, the fewer the number of rules, the better the rules are.

This study proposes a novel method to induce rules with high rates of accuracy, support, and compactness based on global optimization techniques, which have become more and more useful in biological and medical researches.

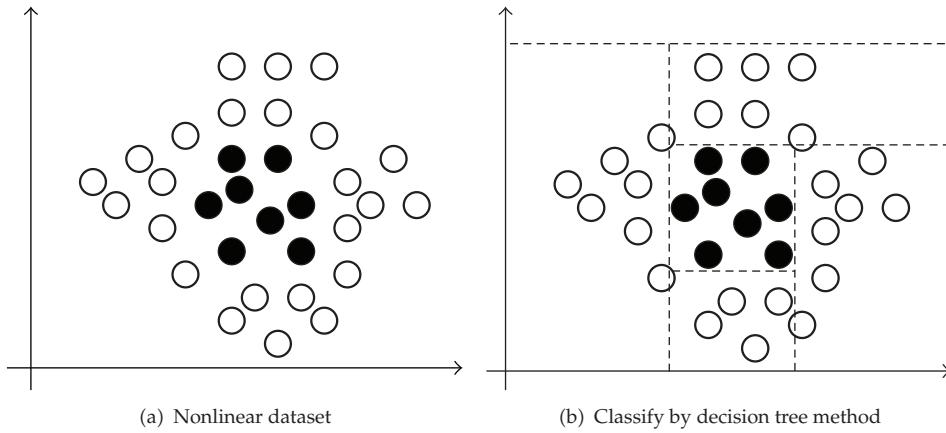
The rest of this paper is organized as follows. Section 2 gives an overview of the related literatures. Two types of mathematical models and a classification algorithm are proposed in Section 3. The numerical examples demonstrate the effectiveness of the proposed method in Section 4. Finally, the main conclusions of this study and future work are drawn in Section 5.

## 2. Literature Review

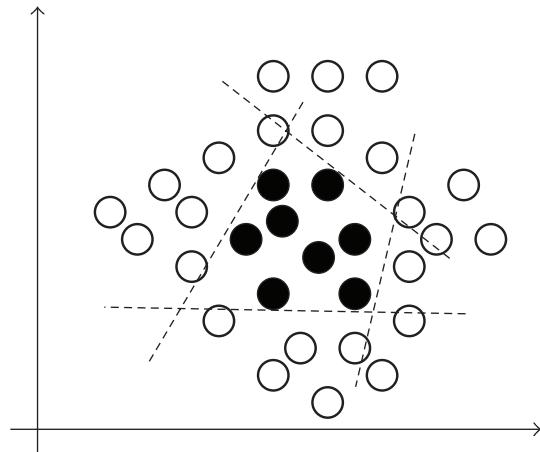
Currently, two well-known methods are used to induce classification rules. The first method is the decision tree-based method, which has been developed in the last few decades [6–10]. It is widely applied to fault isolation of an induction motor [18] to classify normal or tumor tissues [19], skeletal maturity assessment [20], proteomic mass spectra classification [21], and other cases [22, 23]. Although the decision tree-based method assumes that all classes can be separated by linear operations, the inducing rules will suffer if the boundaries between the classes are nonlinear. In fact, the linearity assumption prohibits practical applications because many biological and medical datasets have complicated nonlinear interactions between attributes and predicted classes.

Consider the classification problem with two attributes as shown in Figure 1, where “○” represents a first-class object, and “●” represents a second-class object. Figure 1 depicts a situation in which a nonlinear relationship exists between the objects of two classes. Decision tree method focuses on inducing classification rules for the objects, as shown in Figure 1(b), in which the decision tree method requires four rectangular regions to classify the objects.

The second is the support vector hyperplane method, which conducts feature selection and rule extraction from the gene expression data of cancer tissue [24]; it is also applied in other applications [12–14, 25]. The technique separates observations of different classes by multiple hyperplanes. As the number of decision variables is required to express the relationship between each training datum and hyperplane, and the separating hyperplane is assumed a nonlinear programming problem, the training speed becomes slow for a large number of training data. Additionally, similar hypersphere support vector methods have been developed by Lin et al. [26], Wang et al. [27], Gu and Wu [28], and Hifi and M'Hallah [29] for classifying objects. In classification algorithms, they partition the sample space using the sphere-structured support vector machine [14, 30]. However, these methods need to form a classification problem as a nonlinear nonconvex program, which makes reaching an optimal solution difficult. Taking Figure 1 as an example, a hyperplane-based method requires four hyperplanes to discriminate the objects, as shown in Figure 2.



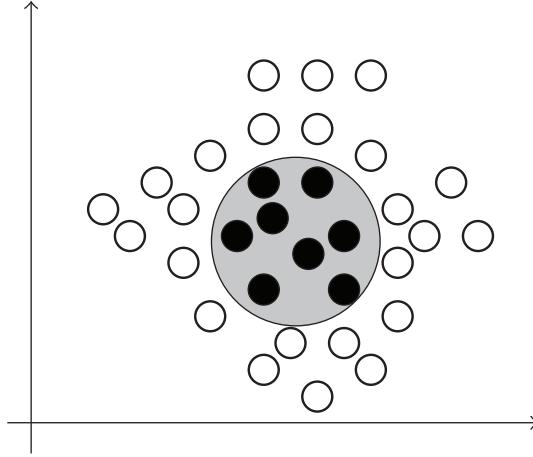
**Figure 1:** Classifying the objects of two classes.



**Figure 2:** Classify by hyperplane method method.

As previously mentioned, many biological and medical datasets have complicated boundaries between attributes and classes. Both decision tree-based methods and hyperplane-based methods find only the rules with high accuracy, which either cover only a narrow part of the objects or require numerous attributes to explain a classification rule. Although these methods are computationally effective for deducing the classifications rules, they have two limitations as follows.

- (i) Decision tree-based methods are heuristic approaches that can only induce feasible rules. Moreover, decision tree-based methods split the data into hyperrectangular regions using a single variable, which may generate a large number of branches (i.e., low rates of compactness).
  - (ii) Hyperplane-based methods use numerous hyperplanes to separate objects of different classes and divide the objects in a dataset into indistinct groups. The method may generate a large number of hyperplanes and associated rules with low rates of compactness.



**Figure 3:** Classify by hypersphere.

Therefore, this study proposes a novel hypersphere method to induce classification rules based on a piecewise linearization technique. The technique reformulates the original hypersphere model by a piecewise linearization approach using a number of binary variables and constraints in the number of piecewise line segments. As the number of break points used in the linearization process increases, the error in linear approximation decreases, and an approximately global optimal solution of the hypersphere model can be obtained. That is, the proposed method is an optimization approach that can find the optimal rules with a high rate of accuracy, support, and compactness. The concept of the hypersphere method is depicted in Figure 3, in which only one circle is required to classify the objects. All objects of class “●” are covered by a circle, and those not covered by this circle belong to class “○.”

### 3. The Proposed Models and Algorithm

As the classification rules directly affect the rates of accuracy, support, and compactness, we formulate two models to determine the highest accuracy rate and support rate, respectively. To facilitate the discussion, the related notations are introduced first:

$a_{i,j}$ :  $j$ 'th attribute value of the  $i$ 'th object,

$h_{t,k,j}$ :  $j$ 'th center value of the  $k$ 'th hypersphere for class  $t$ ,

$r_{t,k}$ : radius of the  $k$ 'th hypersphere for class  $t$ ,

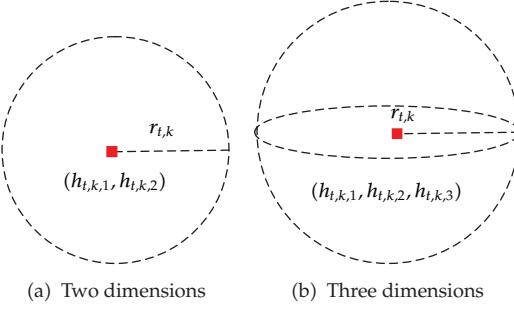
$n(t)$ : number of objects for class  $t$ ,

$c_i$ :  $i$ 'th object belonging to class  $c_i \in \{1, 2, \dots, g\}$ ,

$m$ : number of attributes,

$R_t$ : a rule describing class  $t$ .

Based on these notations, we propose two types of classification models as follow.



**Figure 4:** The concept of hypersphere method.

### 3.1. Two Types of Classification Models

Considering the object  $x_i$  and hypersphere  $S_{t,k}$ , and normalizing  $a_{i,j}$  (i.e., to express its scale easily), we then have the following three notations.

*Notation 1.* Normalization rescales all  $a_{i,j}$  as  $a'_{i,j}$ . The following is a normalizing formula:

$$a'_{i,j} = \frac{a_{i,j} - \underline{a}_j}{\bar{a}_j - \underline{a}_j}, \quad (3.1)$$

where  $0 \leq a'_{i,j} \leq 1$ ,  $\bar{a}_j$  is the largest value of attribute  $j$ , and  $\underline{a}_j$  is the smallest value of attribute  $j$ .

*Notation 2.* A general form for expressing an object  $x_i$  is written as

$$x_i = (a'_{i,1}, a'_{i,2}, \dots, a'_{i,m}; c_i), \quad (3.2)$$

where  $c_i$  is the class index of object  $x_i$ .

*Notation 3.* A general form for expressing a hypersphere  $S_{t,k}$  is written as

$$S_{t,k} = (h_{t,k,1}, h_{t,k,2}, \dots, h_{t,k,m}; r_{t,k}), \quad (3.3)$$

where  $S_{t,k}$  is the  $k$ 'th hypersphere for class  $t$ .

We use two and three dimensions (i.e., two attributes and three attributes) as visualizations to depict clearly a circle and a sphere, respectively (Figure 4). Figure 4(a) denotes the centroid of the circle as  $(h_{t,k,1}, h_{t,k,2})$  and the radius of the circle as  $r_{t,k}$ . They are extended to three dimensions called sphere (Figure 4(b)); in  $m$  dimensions (i.e.,  $m$  attributes),  $m > 3$ , which are then called hyperspheres.

To find each center and the radius of the hypersphere, the following two nonlinear models are considered. The first model looks for a support rate as high as possible while the accuracy rate is fixed to 1, as shown in Model 1.

*Model 1.* One has the following:

$$\begin{aligned}
 & \text{Maximize} \quad \sum_{i \in I^+} u_{t,i,k} \\
 & \text{subject to} \quad \sum_{j=1}^m (a'_{i,j} - h_{t,k,j})^2 \leq (1 - u_{t,i,k})M + r_{t,k}^2, \quad \forall i \in I^+, \\
 & \quad \sum_{j=1}^m (a'_{i',j} - h_{t,k,j})^2 > r_{t,k}^2, \quad \forall i' \in I^-, \\
 & \quad u_{t,i,k} \in \{0, 1\} \quad \forall i \in I^+, r_{t,k} \geq 0, \text{ and } M \text{ is big enough constant,}
 \end{aligned} \tag{3.4}$$

where  $I^+$  and  $I^-$  are the two sets for all objects expressed, respectively, by

$$I^+ = \{i \mid i = 1, 2, \dots, n, \text{ where object } i \in \text{class } t\}, \tag{3.5}$$

$$I^- = \{i' \mid i' = 1, 2, \dots, n, \text{ where object } i' \notin \text{class } t\}. \tag{3.6}$$

Referring to Li and Chen [3], the rates of accuracy and support of  $R_t$  in Model 1 can be specified by the following definitions.

*Definition 3.1.* The accuracy rate of a rule  $R_t$  for Model 1 is  $AR(R_t) = 1$ .

*Definition 3.2.* The support rate of a rule  $R_t$  for Model 1 is specified as follows.

- (i) If  $\sum_{k \in K} u_{t,i,k} \geq 1$  for all  $i$  belonging to class  $t$ , then  $U_{t,i} = 1$ ; otherwise  $U_{t,i} = 0$ , where  $K$  indicates the hypersphere set for class  $t$ .
- (ii)

$$SR(R_t) = \frac{\sum_{i \in \text{class } t} U_{t,i}}{n(t)}, \tag{3.7}$$

where  $n(t)$  indicates the number of objects belonging to class  $t$ .

The second model looks for an accuracy rate as high as possible while the support rate is fixed to 1, as shown in Model 2.

*Model 2.* One has the following:

$$\begin{aligned}
 & \text{Maximize} \quad \sum_{i' \in I^-} v_{t,i',k} \\
 & \text{subject to} \quad \sum_{j=1}^m (a'_{i,j} - h_{t,k,j})^2 \leq r_{t,k}^2, \quad \forall i \in I^+
 \end{aligned}$$

$$\sum_{j=1}^m \left( a'_{i,j} - h_{t,k,j} \right)^2 > (v_{t,i',k} - 1)M + r_{t,k}^2, \quad \forall i' \in I^-,$$

$$v_{t,i',k} \in \{0, 1\}, \quad \forall i' \in I^-, \quad r_{t,k} \geq 0,$$
(3.8)

where  $I^+$  and  $I^-$  are the two sets expressed by (3.5) and (3.6), respectively.

Similarly, the rates of accuracy and support of  $R_t$  in Model 2 can be considered as follows.

*Definition 3.3.* The accuracy rate of a rule  $R_t$  of Model 2 is denoted as  $AR(R_t)$  and is specified as follows.

- (i) If  $\sum_{k \in K} v_{t,i',k} = 0$  belongs to class  $t$ , then  $V_{t,i'} = 1$  for all  $i'$ ; otherwise,  $V_{t,i'} = 0$ , where  $K$  represents the hypersphere set for class  $t$ .
- (ii)

$$AR(R_t) = \frac{\|R_t\| - \sum_{i' \in \text{class } t} V_{t,i'}}{\|R_t\|}, \quad (3.9)$$

where  $\|R_t\|$  represents the number of total objects covered by  $R_t$ .

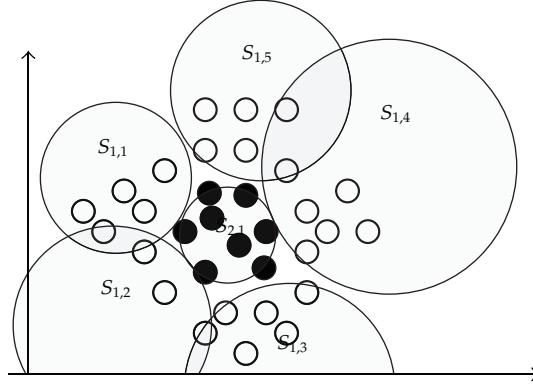
*Definition 3.4.* The support rate of a rule  $R_t$  of Model 2 is denoted as  $SR(R_t)$ , and  $SR(R_t) = 1$ .

*Definition 3.5.* The compactness rate of a set of rules  $R_1, \dots, R_g$ , denoted as  $CR(R_1, \dots, R_g)$ , is expressed as follows:

$$CR(R_1, \dots, R_g) = \frac{g}{\sum_{t=1}^g US_t}, \quad (3.10)$$

where  $US_t$  means the number of hyperspheres and unions of hyperspheres for class  $t$ . A union of hyperspheres indicates that the object is covered by different hyperspheres, as shown in Figure 5. Take Figure 5 for an example, in which there are two classes. The objects of class “○” are covered by two unions of the circles (i.e.,  $S_{1,1} \cup S_{1,2} \cup S_{1,3}$  and  $S_{1,4} \cup S_{1,5}$ ), and the objects of class “●” are covered by one circle (i.e.,  $S_{2,1}$ ). Therefore,  $US_1 = 2$ ,  $US_2 = 1$ , and  $CR(R_1, R_2) = 2/3$ .

Moreover, Models 1 and 2 are separable nonlinear programs solvable to find an optimal solution by linearizing the quadratic terms  $h_{t,k,j}^2$ . The piecewise linearization technique is discussed as follows.



**Figure 5:** Classify by hypersphere method.

**Proposition 3.6** (referring to Beale and Forrest [31]). Denote approximate function  $L(f(x))$  as a piecewise linear function (i.e., linear convex combination) of  $f(x)$ , where  $b_l, l = 1, 2, \dots, q$  represents the break points of  $L(f(x))$ .  $L(f(x))$  is expressed as follows:

$$f(x) \approx L(f(x)) = \sum_{l=1}^q f(b_l) w_l, \quad (3.11)$$

$$x = \sum_{l=1}^q w_l b_l, \quad (3.12)$$

$$\sum_{l=1}^q w_l = 1, \quad (3.13)$$

where  $w_l \geq 0$ , and (3.13) is a special-ordered set of type 2 (SOS2) constraint (reference to Beale and Forrest [31]).

Note that the SOS2 constraint is a set of variables in which at most two variables may be nonzero. If two variables are nonzero, they must be adjacent in the set.

*Notation 4.* According to Proposition 3.6, let  $f(x) = h_{t,k,j}^2$ .  $f(x)$  is linearized by the Proposition 3.6 and is expressed as  $L(h_{t,k,j}^2)$ .

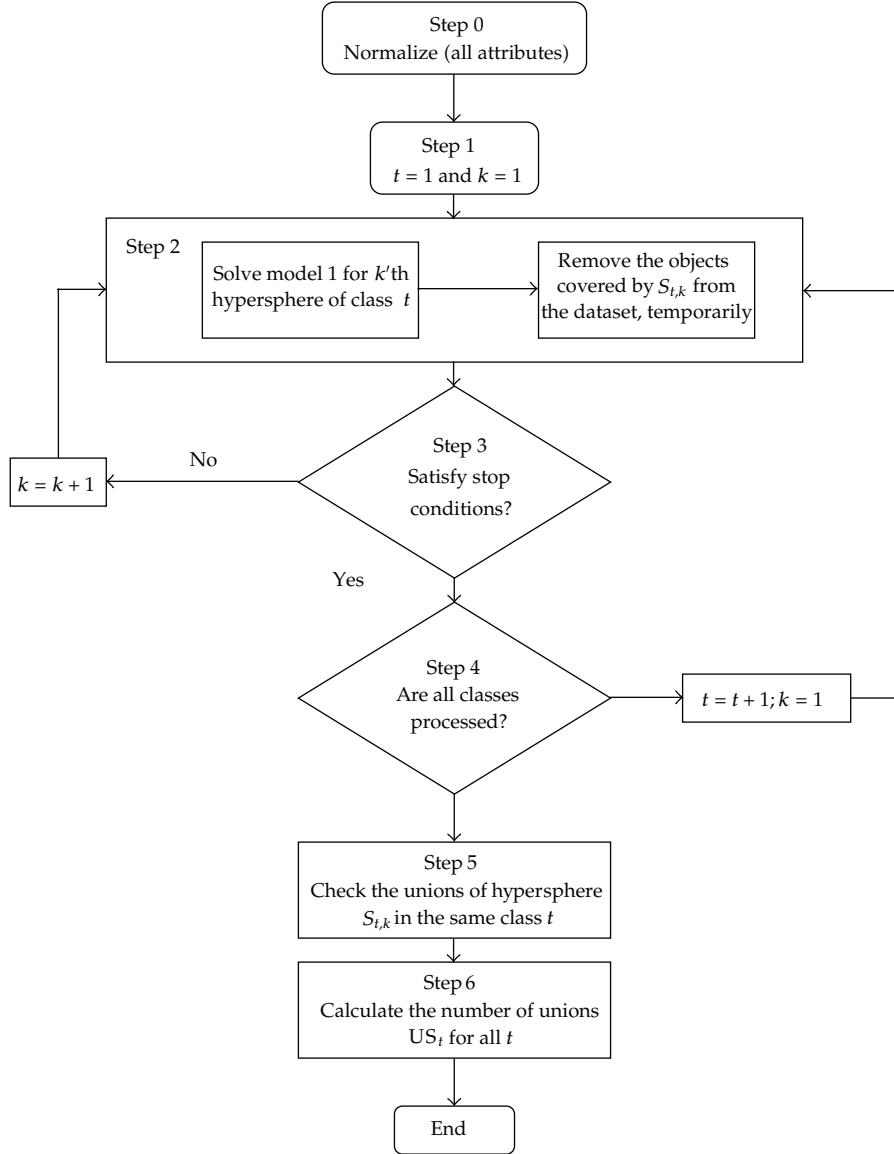
### 3.2. Solution Algorithm

A proposed algorithm is also presented to seek the highest accuracy rate or the highest support rate, as described as follows.

*Algorithm 3.7.*

*Step 1.* Normalize all attributes (i.e., rescale  $a'_{i,j} = (a_{i,j} - \underline{a}_j)/(\bar{a}_j - \underline{a}_j)$  to be  $0 \leq a'_{i,j} \leq 1$ ).

*Step 2.* Initialization:  $t = 1$  and  $k = 1$ .



**Figure 6:** Flowchart of the proposed algorithm.

*Step 3.* Solve Model 1 (or Model 2) to obtain the  $k$ 'th hypersphere of class  $t$ . Remove the objects covered by  $S_{t,k}$  from the dataset temporarily.

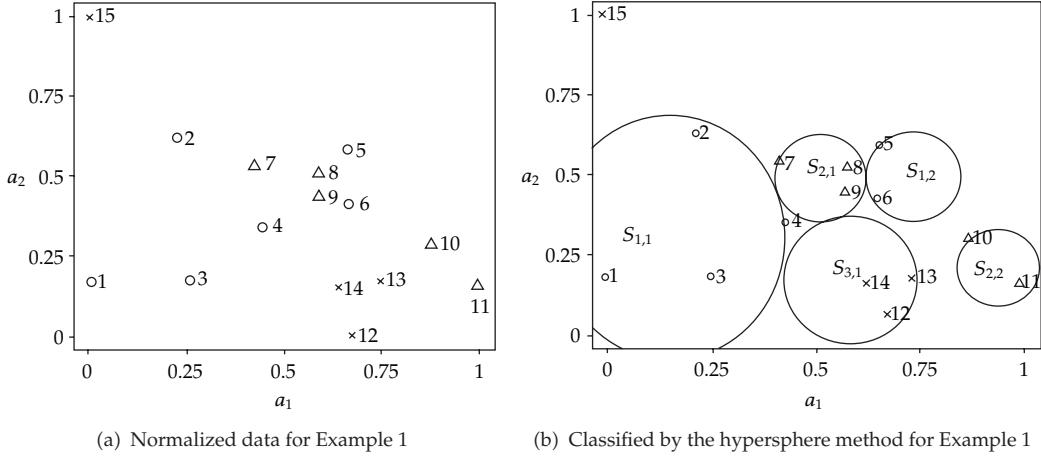
*Step 4.* Let  $k = k + 1$ , and resolve Model 1 (or Model 2) until all objects in class  $t$  are assigned to the hyperspheres of same class.

*Step 5.* Let  $k = 1$  and  $t = t + 1$ , and reiterate Step 3 until all classes are processed.

*Step 6.* Check the independent hyperspheres and unions of hyperspheres  $S_{t,k}$  in the same class  $t$ .

**Table 1:** Dataset of Example 1.

Object	$a_{i,1}$	$a_{i,2}$	$c_i$	Object	$a_{i,1}$	$a_{i,2}$	$c_i$	Object	$a_{i,1}$	$a_{i,2}$	$c_i$
$x_1$	6	8	1	$x_2$	12	20	1	$x_3$	13	8	1
$x_4$	18	12.5	1	$x_5$	24	19	1	$x_6$	24	14.5	1
$x_7$	17.5	17.5	2	$x_8$	22	17	2	$x_9$	22	15	2
$x_{10}$	30	11	2	$x_{11}$	33.5	7.5	2	$x_{12}$	24.5	3.5	3
$x_{13}$	26.5	8	3	$x_{14}$	23.5	7.5	3	$x_{15}$	6	30	3

(Symbols of  $c_i$ ) 1:  $\circ$  2:  $\Delta$  3:  $\times$ .**Figure 7:** Visualization of Example 1.

*Step 7.* Calculate and record the number of independent hyperspheres and unions of hyperspheres in  $US_t$ , and iterate  $t$  until all classes are done.

According to this algorithm, we can obtain the optimal rules to classify objects most efficiently. The process of the algorithm is depicted in Figure 6.

### 3.3. Operation of a Simple Example

Consider a dataset  $T$  in Table 1 as an example, which has object  $i$ , two attributes  $(a_{i,1}, a_{i,2})$ , and an index of classes  $(c_i)$  for  $i = 1, 2, \dots, 15$ . The dataset  $T$  is expressed as  $T = \{x_i \mid (a_{i,1}, a_{i,2}; c_i) \forall i = 1, 2, \dots, 15\}$ . There are the domain values of  $c_i \in \{1, 2, 3\}$ . As there are only two attributes, these 15 objects can be plotted on a two-dimensional space after normalizing them, as shown in Figure 7(a).

This example can be solved by the proposed algorithm as follows.

*Step 1.* Normalize all attributes (i.e.,  $a'_{i,1} = (a_{i,1} - 6) / (33.5 - 6)$  and  $a'_{i,2} = (a_{i,2} - 3.5) / (30 - 3.5)$ ).

*Step 2.* Initialization:  $t = 1$  and  $k = 1$ .

**Table 2:** Centroid points for the Iris data set by the proposed method.

Rule number	Union of spheres	$S_{t,k}$	$h_{t,k,1}$	$h_{t,k,2}$	$h_{t,k,3}$	$h_{t,k,4}$	$r_{t,k}$
$R_1$	$S_{1,1} \cup S_{2,1} \cup S_{2,2} \cup S_{2,3}$	$S_{1,1}$	0.366	0.807	0.000	0.000	0.6205
		$S_{2,1}$	0.557	0.320	0.205	0.460	0.2540
		$S_{2,2}$	0.575	0.581	0.626	0.515	0.0612
$R_3$	$S_{3,1} \cup S_{3,2}$	$S_{3,1}$	0.248	0.000	2.226	2.151	4.8087
		$S_{3,2}$	0.329	0.187	0.650	0.613	0.0330

**Table 3:** Comparing results for the Iris flower data set ( $R_1, R_2, R_3$ ).

Items	Proposed method	Decision tree	Hyperplane support vector
$AR(R_1, R_2, R_3)$	(1,1,1)	(1,0.98,0.98)	(1,0.98,0.96)
$SR(R_1, R_2, R_3)$	(1,0.98,0.98)	(1,0.98,0.98)	(1,0.96,0.98)
$CR$	1	0.5	0.1875

*Step 3.* The classification model (i.e., Model 1) is linearly formulated as follows:

$$\begin{aligned}
& \text{Maximize} \quad \sum_{i \in I^+} u_{t,i,k} \\
& \text{subject to} \quad \sum_{j=1}^m \left( a'_{ij}^2 - 2a'_{ij} h_{t,k,j} + L(h_{t,k,j}^2) \right) \leq (1 - u_{t,i,k})M + r_{t,k}^2, \quad \forall i \in I^+, \\
& \quad \sum_{j=1}^m \left( a'_{i',j}^2 - 2a'_{i',j} h_{t,k,j} + L(h_{t,k,j}^2) \right) > r_{t,k}^2, \quad \forall i' \in I^-, \\
& \quad u_{t,i,k} \in \{0, 1\}, \quad r_{t,k}^2 \geq 0,
\end{aligned} \tag{3.14}$$

where  $I^+ = \{x_1, x_2, \dots, x_6\}$  and  $I^- = \{x_7, x_8, \dots, x_{15}\}$ . The optimal solution of the  $(h_{t,k,1}, h_{t,k,2}, r_{t,k}) = (0.047, 0.265, 0.15749)$  for  $S_{1,1}$ , where  $S_{1,1}$  covers objects 1–4. We then temporarily remove these objects covered by  $S_{1,1}$ .

*Step 4.*  $k = k + 1$ : the optimal solution of the  $(h_{t,k,1}, h_{t,k,2}, r_{t,k}) = (0.736, 0.5, 0.0138)$  for  $S_{1,2}$ , where  $S_{1,2}$  covers objects 5–6. Class 1 is then done.

*Step 5.* As  $t = t + 1$ ,  $k = 1$ , and Steps 3 and 4 are iterated, we then, respectively, have optimal solutions for  $S_{t,k}$  as follows. The results are shown in Figure 7(b).

- (i)  $(h_{2,1,1}, h_{2,1,2}, r_{2,1}) = (0.514, 0.469, 0.0127)$ , where  $S_{2,1}$  covers objects 7–9.
- (ii)  $(h_{2,2,1}, h_{2,2,2}, r_{2,2}) = (0.929, 0.210, 0.0251)$ , where  $S_{2,2}$  covers objects 10–11.
- (iii)  $(h_{3,1,1}, h_{3,1,2}, r_{3,1}) = (0.583, 0.188, 0.0436)$ , where  $S_{3,1}$  covers objects 12–14.

*Step 6.* Check and calculate the unions of hypersphere  $S_{t,k}$  for all  $k$  in class  $t$  (i.e., Initial  $t = 1$ ).

*Step 7.* As  $t = t + 1$ , mark the number of unions of class  $t$  into  $US_t$  and iterate Step 6 until  $t = g$ .

## 4. Numerical Examples

This study shows how the experimental results evaluate the performance, including accuracy, support, and compactness rates, and compares the proposed model with different methods using CPLEX [32]. All tests were run on a PC equipped with an Intel Pentium D 2.8 GHz CPU and 2 GMB RAM. Three datasets were tested in our experiments as follows:

- (i) Iris Flower dataset introduced by Sir Ronald Aylmer Fisher (1936),
- (ii) European barn swallow (*Hirundo rustica*) dataset obtained by trapping individual swallows in Stirlingshire, Scotland, between May and July 1997 [1, 3],
- (iii) the highly selective vagotomy (HSV) patient dataset of F. Raszeja Memorial Hospital in Poland [3, 33, 34].

### 4.1. Iris Flower Dataset

The Iris Flower dataset contains 150 objects. Each object is described by four attributes (i.e., sepal length, sepal width, petal length, and petal width) and is classified by one of three classes (i.e., setosa, versicolor, and virginica). By solving the proposed method, we induced six hyperspheres (i.e.,  $S_{1,1} \in$  Class 1,  $S_{2,1}, S_{2,2}, S_{2,3} \in$  Class 2, and  $S_{3,1}, S_{3,2} \in$  Class 3). The induced classification rules are reported in Table 2. Table 2 also lists a hypersphere and two unions (i.e.,  $S_{1,1}, S_{2,1} \cup S_{2,2} \cup S_{2,3}$ , and  $S_{3,1} \cup S_{3,2}$ ) of hyperspheres with centroid points and radii.

Rule  $R_1$  in Table 2 contains a hypersphere  $S_{1,1}$ , which implies that

- (i) “if  $(a'_{i,1} - 0.366)^2 + (a'_{i,2} - 0.807)^2 + (a'_{i,3} - 0)^2 + (a'_{i,4} - 0)^2 \leq 0.6205$ , then object  $x_i$  belongs to class 1.”

Rule  $R_2$  in Table 2 contains a union of three hyperspheres (i.e.,  $S_{2,1} \cup S_{2,2} \cup S_{2,3}$ ) which implies that

- (i) “if  $(a'_{i,1} - 0.557)^2 + (a'_{i,2} - 0.32)^2 + (a'_{i,3} - 0.205)^2 + (a'_{i,4} - 0.46)^2 \leq 0.254$ , then object  $x_i$  belongs to class 2,” or
- (ii) “if  $(a'_{i,1} - 0.575)^2 + (a'_{i,2} - 0.581)^2 + (a'_{i,3} - 0.626)^2 + (a'_{i,4} - 0.515)^2 \leq 0.0612$ , then object  $x_i$  belongs to class 2,” or
- (iii) “if  $(a'_{i,1} - 0.423)^2 + (a'_{i,2} - 0.261)^2 + (a'_{i,3} - 0.352)^2 + (a'_{i,4} - 0.49)^2 \leq 0.1388$ , then object  $x_i$  belongs to class 2.”

Rule  $R_3$  in Table 2 contains a union of two hyperspheres (i.e.,  $S_{3,1} \cup S_{3,2}$ ), which implies that

- (i) “if  $(a'_{i,1} - 0.248)^2 + (a'_{i,2} - 0)^2 + (a'_{i,3} - 2.226)^2 + (a'_{i,4} - 2.151)^2 \leq 4.8087$ , then object  $x_i$  belongs to class 3,” or
- (ii) “if  $(a'_{i,1} - 0.329)^2 + (a'_{i,2} - 0.187)^2 + (a'_{i,3} - 0.65)^2 + (a'_{i,4} - 0.613)^2 \leq 0.033$ , then object  $x_i$  belongs to class 3.”

Comparing the proposed method with both decision tree [3] and hyperplane methods [35] in deducing the classification rules for the Iris Flower dataset, Table 3 lists the experimental result.

**Table 4:** Centroid points for the Swallow data set by the proposed method.

Rule number	Union of spheres	$S_{t,k}$	$h_{t,k,1}$	$h_{t,k,2}$	$h_{t,k,3}$	$h_{t,k,4}$	$h_{t,k,5}$	$h_{t,k,6}$	$h_{t,k,7}$	$h_{t,k,8}$	$r_{t,k}$
$R_1$	$S_{1,1} \cup S_{1,2} \cup S_{1,3} \cup S_{1,4}$	$S_{1,1}$	0.607	0.110	0.806	0.000	0.000	0.406	1.077	1.163	1.780
		$S_{1,2}$	0.483	0.000	0.236	0.328	0.000	0.793	0.280	0.931	0.948
		$S_{1,3}$	0.588	0.000	1.179	0.000	0.000	0.653	1.037	0.982	1.879
		$S_{1,4}$	0.414	0.000	0.658	0.000	0.227	0.085	0.563	1.224	1.197
$R_2$	$S_{2,1} \cup S_{2,2} \cup S_{2,3} \cup S_{2,4} \cup S_{2,5}$	$S_{2,1}$	0.358	0.667	0.371	1.403	0.360	1.530	0.634	0.114	2.825
		$S_{2,2}$	0.198	1.102	0.697	0.000	0.000	1.734	0.110	0.595	2.632
		$S_{2,3}$	0.532	0.422	0.323	1.676	1.009	0.000	0.180	0.020	2.458
		$S_{2,4}$	0.528	0.000	0.430	0.876	0.579	0.461	0.435	0.040	0.694
		$S_{2,5}$	0.659	1.408	0.516	0.000	0.382	0.000	0.173	0.064	1.570

**Table 5:** Comparing results for the Swallow data set ( $R_1, R_2$ ).

Items	Proposed method	Decision tree	Hyperplane support vector
$AR(R_1, R_2)$	(1, 1)	(0.97, 1)	(0.97, 1)
$SR(R_1, R_2)$	(1, 0.97)	(0.97, 1)	(0.97, 1)
$CR$	1	0.3	0.1

The accuracy rates of ( $R_1, R_2, R_3$ ) in the proposed method are (1,1,1), as Model 1 has been solved. This finding indicates that none of objects in class 2 or class 3 are covered by  $S_{1,1}$ , none of objects in classes 1 or 3 are covered by  $S_{2,1} \cup S_{2,2} \cup S_{2,3}$ , and none of the objects in classes 1 or 2 are covered by  $S_{3,1} \cup S_{3,2}$ . The support rate of ( $R_1, R_2, R_3$ ) in the proposed method is (1,0.98,0.98), indicating that all objects in class 1 are covered by  $S_{1,1}$ , 98% of the objects in class 2 are covered by  $S_{2,1}, S_{2,2}$ , and  $S_{2,3}$ , and 98% of the objects in class 3 are covered by  $S_{3,1}$  and  $S_{3,2}$ . The compactness rate of rules  $R_1, R_2$ , and  $R_3$  is computed as  $CR(R_1, R_2, R_3) = 3/3 = 1$ . Finally, we determine the following.

- (i) Although all three methods perform very well in the rates of accuracy and support, the proposed method has the best performance for the accuracy of classes 2 and 3 (i.e.,  $R_2$  and  $R_3$ ).
- (ii) The proposed method has the best compactness rate.

#### 4.2. Swallow Dataset

The European barn swallow (*Hirundo rustica*) dataset was obtained by trapping individual swallows in Stirlingshire, Scotland, between May and July 1997. This dataset contains 69 swallows. Each object is described by eight attributes, and it belongs to one of two classes (i.e., the birds are classified by the gender of individual birds).

Here, we also used Model 1 to induce the classification rules. Table 4 lists the optimal solutions (i.e., centroid and radius) for both rules  $R_1$  and  $R_2$ .

The result of the decision tree method, which is referred to in Li and Chen [3], is listed in Table 5, where  $AR(R_1, R_2) = (0.97, 1)$ ,  $SR(R_1, R_2) = (0.97, 1)$ , and  $CR = 0.3$ .

The result of the hyperplane method, referred to in Chang and Lin [35], is also listed in Table 5, where  $AR(R_1, R_2) = (0.97, 1)$ ,  $SR(R_1, R_2) = (0.97, 1)$ , and  $CR = 0.1$ .

We compared the three methods in Table 5 to show that the proposed method can induce rules with better or equivalent values of  $AR$  and  $SR$ . In fact, the proposed method also has the best compactness rate.

#### **4.3. HSV Dataset**

The HSV dataset contains 122 patients classified into four classes, with each patient having 11 preoperating attributes. To maximize the support rate with respect to the proposed method (i.e., Model 1), the proposed method generated seven hyperspheres and three unions of hyperspheres. The centroids and radii of the hyperspheres are reported in Table 6, and a comparison with other methods is reported in Table 7.

Using the decision tree method in the HSV dataset generates 24 rules. In addition, the hyperplane method deduces 45 hyperplanes for the HSV dataset. Table 7 also shows that the proposed method can find rules with the highest rates (i.e.,  $AR$ ,  $SR$ , and  $CR$ ) compared with the other two methods.

#### **4.4. Limitation of the Proposed Method**

The hypersphere models are solved by one of the most powerful mixed-integer program software CPLEX [32] running in a PC. Based on optimization technique, the results of the numerical examples illustrate that the usefulness of the proposed method is better than that of the current methods, including the decision tree method and the hyperplane support vector method. As the solving time of the hypersphere model, which is linearized, mainly depends on the number of binary variables and constraints, solving the reformulated hypersphere model from the proposed algorithm takes about one minute for each dataset (i.e., in Sections 4.1 and 4.3), in which using eight piecewise line segments linearizes the nonlinear nonconvex term (i.e.,  $L(h_{t,k,j}^2)$ ) of Model 1.

The computing time for solving a linearized hypersphere program grows rapidly as the numbers of binary variables and constraints increase. Also, the computing time of the proposed method is slower than that of the decision tree method and hyperplane method, especially for large datasets or a great number of piecewise line segments. In the further study, utilizing a mainframe-version optimization software [36–38], integrating metaheuristic algorithms, or using distributed computing techniques can enhance solving speed to conquer this problem.

### **5. Conclusions and Future Work**

This study proposes a novel method for deducing classification rules, which can find the optimal solution based on a hypersphere domain. The optimization technique for finding classification rules is approached to optimal. Results of the numerical examples illustrate that the usefulness of the proposed method is better than that of the current methods, including the decision tree method and the hyperplane method. The proposed method is guaranteed to find an optimal rule, but the computational complexity grows rapidly by increasing the problem size. More investigation and research are required to enhance further

**Table 6:** Centroid points for the HSV data by the proposed method.

Rule numbers	Union of spheres	$S_{k,l}$	$h_{t,k,1}$	$h_{t,k,2}$	$h_{t,k,3}$	$h_{t,k,4}$	$h_{t,k,5}$	$h_{t,k,6}$	$h_{t,k,7}$	$h_{t,k,8}$	$h_{t,k,9}$	$h_{t,k,10}$	$h_{t,k,11}$	$r_{t,k}$
$R_1$	$S_{1,1}$	$S_{1,1}$	0.504	0.528	0.366	0.288	0.850	0.848	0.351	-0.282	0.405	0.234	0.313	1.576
	$S_{1,2}$	$S_{1,2}$	0.469	0.395	0.312	0.133	-0.274	-0.309	0.134	0.590	0.813	1.000	-0.847	3.044
	$S_{1,3}$	$S_{1,3}$	0.287	0.867	0.458	0.189	0.977	0.267	0.017	-0.846	0.462	-1.000	1.000	4.249
	$S_{1,2} \cup S_{1,8}$	$S_{1,4}$	0.586	-0.400	0.605	0.467	-0.167	-0.631	0.268	1.000	0.526	1.000	0.120	3.365
	$S_{1,5}$	$S_{1,5}$	0.296	-0.511	-1.000	0.536	-0.263	0.552	0.954	0.635	-0.117	-1.000	0.600	4.608
	$S_{1,6}$	$S_{1,6}$	0.775	0.678	-0.092	0.340	0.366	0.455	0.451	1.000	0.422	0.295	0.470	1.560
	$S_{1,7}$	$S_{1,7}$	0.525	0.296	0.194	0.124	-0.945	-0.467	0.127	1.000	0.097	-0.897	0.654	4.113
	$S_{1,8}$	$S_{1,8}$	0.000	0.140	0.109	0.250	0.193	0.168	0.063	0.142	0.181	0.119	0.082	0.089
$R_2$	$S_{2,1}$	$S_{2,1}$	0.513	0.056	0.143	0.078	0.533	0.439	0.180	0.167	-0.284	-0.464	0.749	1.450
	$S_{2,2}$	$S_{2,2}$	0.533	0.381	0.139	0.568	-0.373	-0.005	0.080	0.831	0.293	0.215	0.369	1.293
	$S_{2,3}$	$S_{2,3}$	0.000	-0.450	0.709	0.553	0.896	1.000	0.157	-0.447	0.068	-0.580	1.000	3.579
	$S_{2,4}$	$S_{2,4}$	0.000	0.862	0.543	-0.066	0.389	0.451	0.106	0.394	-0.004	0.042	-0.014	0.409
$R_3$	$S_{3,1} \cup S_{3,2}$	$S_{3,1}$	0.624	0.507	-1.000	0.147	0.840	1.000	0.827	-0.585	0.199	0.831	-0.380	4.277
	$S_{3,2}$	$S_{3,2}$	0.534	0.483	-0.088	0.365	-0.364	0.270	0.437	0.785	0.786	0.682	-1.000	3.003
	$S_{3,3}$	$S_{3,3}$	0.000	0.210	0.630	0.750	-1.000	0.244	0.388	1.000	0.474	-0.604	0.475	3.162
$R_4$	$S_{4,1} \cup S_{4,2} \cup S_{4,3}$	$S_{4,1}$	0.551	0.374	0.637	0.256	0.865	0.944	0.315	-0.831	-0.485	-0.676	0.979	4.330
	$S_{4,2}$	$S_{4,2}$	0.717	0.254	0.548	-0.718	0.118	0.730	-0.547	0.498	-0.580	-0.464	0.821	3.287
	$S_{4,3}$	$S_{4,3}$	0.527	1.000	0.533	0.089	-0.152	-1.000	0.366	0.046	0.547	0.947	-0.436	2.943
	$S_{4,4}$	$S_{4,4}$	0.489	0.625	-0.209	0.522	0.644	0.582	0.615	1.000	0.306	-0.401	0.931	2.059
	$S_{4,5}$	$S_{4,5}$	-0.011	0.491	-0.409	0.155	-0.766	-0.227	-0.107	1.000	0.303	-0.680	-0.185	2.843

**Table 7:** Comparing results for the HSV data set ( $R_1, R_2, R_3, R_4$ ).

Items	Proposed method	Decision tree	Hyperplane support vector
$AR(R_1, R_2, R_3, R_4)$	(1,1,1,1)	(0.93,0.81,0.7,0.71)	(0.9,1,1,0.9)
$SR(R_1, R_2, R_3, R_4)$	(0.99,1,1,1)	(0.93,0.72,0.78,0.71)	(0.9,0.72,0.67,0.69)
$CR$	0.4	0.17	0.09

the computational efficiency of globally solving large-scale classification problems, such as running mainframe-version optimization software, integrating meta-heuristic algorithms, or using distributed computing techniques.

## Acknowledgments

The authors wish to thank the editor and the anonymous referees for providing insightful comments and suggestions, which have helped them improving the quality of the paper. This work was supported by the National Science Council of Taiwan under Grants NSC 100-2811-E-009-040-, NSC 99-2221-E-030-005-, and NSC 100-2221-E-030-009-.

## References

- [1] M. J. Beynon and K. L. Buchanan, "An illustration of variable precision rough set theory: the gender classification of the European barn swallow (*Hirundo rustica*)," *Bulletin of Mathematical Biology*, vol. 65, no. 5, pp. 835–858, 2003.
- [2] H. L. Li and C. J. Fu, "A linear programming approach for identifying a consensus sequence on DNA sequences," *Bioinformatics*, vol. 21, no. 9, pp. 1838–1845, 2005.
- [3] H. L. Li and M. H. Chen, "Induction of multiple criteria optimal classification rules for biological and medical data," *Computers in Biology and Medicine*, vol. 38, no. 1, pp. 42–52, 2008.
- [4] C. W. Chu, G. S. Liang, and C. T. Liao, "Controlling inventory by combining ABC analysis and fuzzy classification," *Computers and Industrial Engineering*, vol. 55, no. 4, pp. 841–851, 2008.
- [5] J.-X. Chen, "Peer-estimation for multiple criteria ABC inventory classification," *Computers & Operations Research*, vol. 38, no. 12, pp. 1784–1791, 2011.
- [6] E. B. Hunt, J. Marin, and P. J. Stone, *Experiments in Induction*, Academic Press, New York, NY, USA, 1966.
- [7] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, Wadsworth Statistics/Probability Series, Wadsworth Advanced Books and Software, Belmont, Calif, USA, 1984.
- [8] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
- [9] J. R. Quinlan, "Simplifying decision trees," *International Journal of Man-Machine Studies*, vol. 27, no. 3, pp. 221–234, 1987.
- [10] J. R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufman, San Mateo, Calif, USA, 1993.
- [11] H. Kim and W. Y. Loh, "Classification Trees with Unbiased Multiway Splits," *Journal of the American Statistical Association*, vol. 96, no. 454, pp. 589–604, 2001.
- [12] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, USA, 1995.
- [13] R. M. Rifkin, *Everything old is new again: a fresh look at historical approaches in machine learning*, Ph.D. thesis, Massachusetts Institute of Technology, Ann Arbor, MI, USA, 2002.
- [14] S. Katagiri and S. Abe, "Incremental training of support vector machines using hyperspheres," *Pattern Recognition Letters*, vol. 27, no. 13, pp. 1495–1507, 2006.
- [15] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

- [16] C. H. Chu and D. Widjaja, "Neural network system for forecasting method selection," *Decision Support Systems*, vol. 12, no. 1, pp. 13–24, 1994.
- [17] C. Giraud-Carrier, R. Vilalta, and P. Brazdil, "Guest editorial: introduction to the special issue on meta-learning," *Machine Learning*, vol. 54, no. 3, pp. 187–193, 2004.
- [18] D. Pomorski and P. B. Perche, "Inductive learning of decision trees: application to fault isolation of an induction motor," *Engineering Applications of Artificial Intelligence*, vol. 14, no. 2, pp. 155–166, 2001.
- [19] H. Zhang, C. N. Yu, B. Singer, and M. Xiong, "Recursive partitioning for tumor classification with gene expression microarray data," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 98, no. 12, pp. 6730–6735, 2001.
- [20] S. Aja-Fernández, R. De Luis-García, M. Á. Martín-Fernández, and C. Alberola-López, "A computational TW3 classifier for skeletal maturity assessment. A Computing with Words approach," *Journal of Biomedical Informatics*, vol. 37, no. 2, pp. 99–107, 2004.
- [21] P. Geurts, M. Fillet, D. de Seny et al., "Proteomic mass spectra classification using decision tree based ensemble methods," *Bioinformatics*, vol. 21, no. 14, pp. 3138–3145, 2005.
- [22] C. Olaru and L. Wehenkel, "A complete fuzzy decision tree technique," *Fuzzy Sets and Systems*, vol. 138, no. 2, pp. 221–254, 2003.
- [23] M. Pal and P. M. Mather, "An assessment of the effectiveness of decision tree methods for land cover classification," *Remote Sensing of Environment*, vol. 86, no. 4, pp. 554–565, 2003.
- [24] Z. Chen, J. Li, and L. Wei, "A multiple kernel support vector machine scheme for feature selection and rule extraction from gene expression data of cancer tissue," *Artificial Intelligence in Medicine*, vol. 41, no. 2, pp. 161–175, 2007.
- [25] H. Nunez, C. Angulo, and A. Catala, "Rule-extraction from support vector machines," in *Proceedings of the European Symposium on Artificial Neural Networks*, pp. 107–112, 2002.
- [26] Y. M. Lin, X. Wang, W. W. Y. Ng, Q. Chang, D. S. Yeung, and X. L. Wang, "Sphere classification for ambiguous data," in *Proceedings of the International Conference on Machine Learning and Cybernetics*, pp. 2571–2574, August 2006.
- [27] J. Wang, P. Neskovic, and L. N. Cooper, "Bayes classification based on minimum bounding spheres," *Neurocomputing*, vol. 70, no. 4–6, pp. 801–808, 2007.
- [28] L. Gu and H. Z. Wu, "A kernel-based fuzzy greedy multiple hyperspheres covering algorithm for pattern classification," *Neurocomputing*, vol. 72, no. 1–3, pp. 313–320, 2008.
- [29] M. Hifi and R. M'Hallah, "A literature review on circle and sphere packing problems: models and methodologies," *Advances in Operations Research*, vol. 2009, Article ID 150624, 22 pages, 2009.
- [30] M. Zhu, Y. Wang, S. Chen, and X. Liu, "Sphere-structured support vector machines for multi-class pattern recognition," in *Proceedings of the 9th International Conference (RSFDGrC '03)*, vol. 2639 of *Lecture Notes in Computer Science*, pp. 589–593, May 2003.
- [31] E. M. L. Beale and J. J. H. Forrest, "Global optimization using special ordered sets," *Mathematical Programming*, vol. 10, no. 1, pp. 52–69, 1976.
- [32] IBM/ILOG, "Cplex 12.0 reference manual," 2010, <http://www.ilog.com/products/cplex/>.
- [33] D. C. Dunn, W. E. G. Thomas, and J. O. Hunter, "An evaluation of highly selective vagotomy in the treatment of chronic duodenal ulcer," *Surgery Gynecology and Obstetrics*, vol. 150, no. 6, pp. 845–849, 1980.
- [34] K. Slowinski, "Rough classification of HSV patients," in *Intelligent Decision Support-Handbook of Applications and Advances of the Rough Sets Theory*, R. Slowinski, Ed., pp. 77–94, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1992.
- [35] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machines," 2010, <http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html>.
- [36] C. Than, R. Sugino, H. Innan, and L. Nakhleh, "Efficient inference of bacterial strain trees from genome-scale multilocus data," *Bioinformatics*, vol. 24, no. 13, pp. i123–i131, 2008.
- [37] M. Rockville, "Large scale computing and storage requirements for biological and environmental research," Tech. Rep., Ernest Orlando Lawrence Berkeley National Laboratory, Berkeley, Calif, USA, 2009.
- [38] X. Xie, X. Fang, S. Hu, and D. Wu, "Evolution of supercomputers," *Frontiers of Computer Science in China*, vol. 4, no. 4, pp. 428–436, 2010.

## Research Article

# A Modified PSO Algorithm for Minimizing the Total Costs of Resources in MRCPSP

**Mohammad Khalilzadeh,<sup>1</sup> Fereydoon Kianfar,<sup>1</sup>  
Ali Shirzadeh Chaleshtari,<sup>1</sup> Shahram Shadrokh,<sup>1</sup>  
and Mohammad Ranjbar<sup>2</sup>**

<sup>1</sup> Department of Industrial Engineering, Sharif University of Technology, Tehran 11365-8639, Iran

<sup>2</sup> Department of Industrial Engineering, Faculty of Engineering, Ferdowsi University of Mashhad, Mashhad 9177948974, Iran

Correspondence should be addressed to Mohammad Khalilzadeh, mo.kzadeh@gmail.com

Received 6 September 2011; Revised 4 December 2011; Accepted 30 December 2011

Academic Editor: Yi-Chung Hu

Copyright © 2012 Mohammad Khalilzadeh et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We introduce a multimode resource-constrained project scheduling problem with finish-to-start precedence relations among project activities, considering renewable and nonrenewable resource costs. We assume that renewable resources are rented and are not available in all periods of time of the project. In other words, there is a mandated ready date as well as a due date for each renewable resource type so that no resource is used before its ready date. However, the resources are permitted to be used after their due dates by paying penalty costs. The objective is to minimize the total costs of both renewable and nonrenewable resource usage. This problem is called multimode resource-constrained project scheduling problem with minimization of total weighted resource tardiness penalty cost (MRCPSP-TWRTPC), where, for each activity, both renewable and nonrenewable resource requirements depend on activity mode. For this problem, we present a metaheuristic algorithm based on a modified Particle Swarm Optimization (PSO) approach introduced by Tchomté and Gourgand which uses a modified rule for the displacement of particles. We present a prioritization rule for activities and several improvement and local search methods. Experimental results reveal the effectiveness and efficiency of the proposed algorithm for the problem in question.

## 1. Introduction

The resource-constrained project scheduling problem (RCPSP) is the scheduling of project activities subject to precedence relations as well as renewable resource constraints with the objective of minimizing the makespan of the project. Each nonpreemptive activity in RCPSP can be done in a single mode. For more information on RCPSP and solution methods, we refer to Demeulemeester and Herroelen [1]. In the multimode RCPSP (MRCPSP), a set of

allowable modes can be defined for each activity which is characterized by a constant duration and associated resource requirements. In this paper we consider MRCPSp with the objective of minimizing total costs of all resources. Two types of resources, renewable and nonrenewable, are considered. Nonrenewable resource cost of an activity is a function of its resource requirements, determined by its modes. The limited renewable resources are rented and each renewable resource is available in a predetermined sequential time period specified by its ready time and due date and is not available before the ready time. However, each renewable resource can be used after its due date with tardiness penalty cost. As the cost of renting for each renewable resource is fixed, there is no need to incorporate it into the objective function and only tardiness penalty cost is considered for each renewable resource. The MRCPSp under minimization of total costs of resources (RCPSP-TWRTPC) is an applicable problem and a modified version of the MRCPSp in which all assumptions and constraints of the MRCPSp are held, but the objective function is different. We assume that there are a few renewable resources such as very expert human resources with high skill levels, particular types of cranes, and tunnel boring machines that should be leased from other companies providing these types of resources. Since these limited renewable resources are employed in other projects, there is a dictated ready date as well as a due date for each of them such that no resource can be accessible before its ready date, but these resources are allowed to be used after their due dates by paying penalty cost, depending on the resource type. Also, we suppose that there are a few nonrenewable resources like budget, materials, energy, or other resources which are consumed during the project.

Ranjbar et al. [2] studied this problem with single mode for each activity and availability of one unit for each type of renewable resource, without considering nonrenewable resources. They called this problem *resource-constrained project scheduling problem, minimization of total weighted resource tardiness penalty cost* (RCPSP-TWRTPC), which is an extended form of *resource-constrained project scheduling problem* (RCPSP). They developed a metaheuristic-based GRASP algorithm together with a branch and bound procedure to solve the problem.

The problem we have studied here is a generalization of the problem introduced by Ranjbar et al. [2] with more realistic viewpoint of resource costs by considering both renewable and nonrenewable resources cost. We call this problem *multimode resource-constrained project scheduling problem, minimization of total weighted resource tardiness penalty cost* (MRCPSp-TWRTPC).

Several exact and heuristic methods have been presented for MRCPSp. For instance, we can point to branch and cut method introduced by Heilmann [3] and branch and bound method developed by Zhu et al. [4] as two of the most powerful exact methods. Zhang et al. [5] presented classical particle swarm optimization (PSO) methods for multimode resource-constrained project scheduling problems to minimize the duration of construction projects. Lova et al. [6] suggested a heuristic algorithm based on priority rules and thereafter a hybrid genetic algorithm [7] to solve MRCPSp. Jarboui et al. [8] applied a combinatorial particle swarm optimization for solving multimode resource-constrained project scheduling problem. Ranjbar et al. [9] developed scatter search algorithm to tackle this problem, using path relinking methodology as its local search method. Van Peteghem and Vanhoucke [10] proposed genetic algorithm to solve preemptive and nonpreemptive multimode resource-constrained project scheduling problems. Kazemi and Tavakkoli-Moghaddam [11] developed a multimode particle swarm optimization which combines with genetic operator to solve a biobjective multimode resource-constrained project scheduling problem with positive and negative cash flows. The difference between this study and the previous papers is that they all considered the minimization of project makespan as objective function, but, in our

problem, the objective function is minimization of total costs of renewable and nonrenewable resources.

MRCPS-TWRTPC is a generalization of the RCPSP problem, and, considering the NP-hardness of RCPSP [12, 13], the MRCPS-TWRTPC problem is NP-hard as well, and hence, metaheuristic method is the practical approach. In the remainder of this paper, we introduce a metaheuristic-based PSO algorithm for solving this problem. PSO, in its present form, dates back to 1990s; however, in this short period, PSO has shown good performance in a variety of application domains, particularly in the constrained optimization problems. Many researchers studied PSO widely and proposed several modifications. In this paper, we use a modified PSO algorithm developed and used by Tchomté and Gourgand for solving RCPSP efficiently [14].

The rest of this paper is organized as follows. In the next section, MRCPS-TWRTPC is described in detail and is formulated in a mathematical model. In Section 3, a description of the PSO algorithm and its modifications is presented, and in Section 4 an algorithm based on modified PSO introduced by Tchomté and Gourgand [14] is explained. Section 5 is the experimental analysis. Finally, Section 6 concludes the work.

## 2. Problem Description

In MRCPS-TWRTPC, a project is to be scheduled in order to minimize its total costs. Resources available for completing project activities can be classified as either renewable or nonrenewable. Activity  $j$  may have a number of execution modes  $M_j$ . Each activity mode specifies the activity duration and the activity requirements for the certain amount of renewable and nonrenewable resources. Each type of limited renewable resource is rented for a fixed time interval, starting from its ready time and ending with its due date, and is not available before its ready time but can be used after its due date with tardiness penalty cost. Nonrenewable resources are not limited. All activities are ready at the beginning of the project, and no preemption is permitted. If an activity is started under a specific mode, the activity mode cannot be changed. Activity  $j$  executed in mode  $m$  has duration  $d_{jm}$  and requires  $r_{jmk}$  units of renewable resource  $k$  and  $n_{jmk}$  units of nonrenewable resource  $k$ . The project network is depicted by an activity on node (AON) representation with finish-to-start precedence relations and zero time lag. Dummy activities 1 and n correspond to start and completion of the project. The list of activities is topologically numbered; that is, each predecessor of every activity has a smaller number than the number of activity itself. Also, we define the earliest and latest start time of activity  $j$  by  $EST_j$  and  $LST_j$ , respectively.  $EST_j$ s and  $LST_j$ s are computed by CPM forward and backward passes using the mode with shortest duration for each activity and assigning  $LST_n = LFT_n = T$ , where  $T$  is an upper bound for project makespan determined by any valid method, such as the sum of the longest duration of entire project activities plus the ready times of renewable resources. Consequently, each activity  $j$  can only be performed in time period  $[EST_j, LST_j]$ .

We define problem parameters as follows:

$n$ : number of project activities,

$NR$ : number of nonrenewable resources,

$c_k$ : unit cost of nonrenewable resource  $k$ ,

$R$ : number of renewable resources,

- $R_k$ : renewable resource  $k$  availability,
- $r_k$ : ready time of renewable resource  $k$ ,
- $d_k$ : due date of renewable resource  $k$ ,
- $p_k$ : tardiness penalty cost of renewable resource  $k$  for each period,
- $M_j$ : number of modes of activity  $j$ ,
- $P_j$ : the set of predecessors of activity  $j$ ,
- $d_{jm}$ : duration of activity  $j$  under mode  $m$ ,
- $r_{jmk}$ : renewable resource  $k$  requirement for executing activity  $j$  under mode  $m$ ,
- $n_{jmk}$ : nonrenewable resource  $k$  requirement for executing activity  $j$  under mode  $m$ ,
- $\text{EST}_j$ : earliest start time of activity  $j$ ,
- $\text{LST}_j$ : latest start time of activity  $j$ ,
- $T$ : upper bound of the project makespan.

We also define the decision variables as follows:

$$x_{jmt} = \begin{cases} 1, & \text{if activity } j \text{ is started under mode } m \text{ in period } \tau, \\ 0, & \text{otherwise,} \end{cases} \quad (2.1)$$

$$y_{k\tau} = \begin{cases} 1, & \text{if renewable resource } k \text{ is used in period } \tau, \\ 0, & \text{otherwise.} \end{cases}$$

$l_k$ : is the renewable resource  $k$  tardiness, determined by  $l_k = \max\{0, CP_k - d_k\}$ , where  $CP_k$  is the release time of resource  $k$  by the project.

The mixed integer programming model for this problem can be formulated as follows:

$$\text{Min } \sum_{k=1}^{NR} c_k \left( \sum_{j=1}^n \sum_{m=1}^{M_j} n_{jmk} \sum_{\tau=\text{EST}_j}^{\text{LST}_j} x_{jmt} \right) + \sum_{k=1}^R p_k \cdot l_k \quad (2.2)$$

$$\text{S.t. } \sum_{m=1}^{M_j} \sum_{\tau=\text{EST}_j}^{\text{LST}_j} x_{jmt} = 1, \quad j = 1, 2, \dots, n, \quad (2.3)$$

$$\sum_{m=1}^{M_i} \sum_{\tau=\text{EST}_i}^{\text{LST}_i} (\tau + d_{im}) x_{imt} \leq \sum_{m=1}^{M_j} \sum_{\tau=\text{EST}_j}^{\text{LST}_j} \tau x_{jmt}, \quad j = 1, 2, \dots, n, \quad i \in P_j, \quad (2.4)$$

$$\sum_{j=1}^n \sum_{m=1}^{M_j} r_{jm} \sum_{z=\tau-d_{jm}+1}^{\tau} x_{jmz} \leq R_k \cdot y_{k\tau}, \quad k = 1, 2, \dots, R, \quad \tau = 1, 2, \dots, T, \quad (2.5)$$

$$\sum_{\tau=1}^{r_k-1} y_{k\tau} = 0, \quad k = 1, 2, \dots, R, \quad (2.6)$$

$$\tau \cdot y_{k\tau} - d_k \leq l_k, \quad k = 1, 2, \dots, R, \quad \tau = d_k, d_k + 1, \dots, \text{LST}_n, \quad (2.7)$$

$$x_{jm\tau} \in \{0, 1\}, \quad j = 1, 2, \dots, n, \quad m = 1, 2, \dots, M_j, \quad \tau = \text{EST}_j, \dots, \text{LST}_j, \quad (2.8)$$

$$y_{k\tau} \in \{0, 1\}, \quad k = 1, 2, \dots, R, \quad \tau = 0, \dots, \text{LST}_n, \quad (2.9)$$

$$l_k \geq 0, \quad k = 1, 2, \dots, R. \quad (2.10)$$

In the above model, objective function (2.2) is project cost minimization in which the first and second terms are total costs of using nonrenewable resources and total penalty costs of renewable resources tardiness, respectively. Constraint set (2.3) ensures that each activity  $j$  is started under one of its modes within its specified start time periods, that is,  $[\text{EST}_j, \text{LST}_j]$ . Constraint set (2.4) forces precedence relationship between activities. Constraint (2.5) limit renewable resource usage. According to constraint (2.6), renewable resources cannot be used before their ready times and their tardiness periods are determined by constraint (2.7). Finally, constraint sets (2.8), (2.9), and (2.10) are nonfunctional ones.

### 3. Particle Swarm Optimization

The PSO algorithm is relatively recent, evolutionary, and population-based metaheuristic, originally developed by Kennedy and Eberhart [15] and redefined by Shi and Eberhart [16]. In spite of the early stages of the PSO method, it has found broad applications in combinatorial and constrained optimization domains and is currently a main research topic. PSO inspired from the social behavior of natural swarms exploits a swarm of particles for search space that are updated from iteration to iteration. Each particle corresponds to a candidate solution assessed by the objective function of the problem in question and is considered as a point in an  $n$ -dimension space. The status of a particle is represented by its position and velocity [15]. The  $n$ -dimensional position for particle  $i$  at iteration  $t$  can be represented as  $X_{i,t} = (x_{i1}^t, x_{i2}^t, \dots, x_{in}^t)$ . Similarly, the velocity is also an  $n$ -dimension vector for particle  $i$  at iteration  $t$  which can be denoted as  $V_{i,t} = (v_{i1}^t, v_{i2}^t, \dots, v_{in}^t)$ . Using the fitness evaluation, each particle remembers the best position it has perceived so far, referred to as  $P_{i,t}$ , and the best position of all particles, the best position of the best particle in the entire swarm, referred to as  $G_t$ . The position vector of each particle at iteration  $t$  is updated using (3.1), and the particle moves to its new position:

$$X_{i,t+1} = X_{i,t} + V_{i,t}. \quad (3.1)$$

Velocity vector is also updated by(3.2)

$$V_{i,t+1} = c_1 \cdot V_{i,t} + c_2 \cdot r_2 \cdot (P_{i,t} - X_{i,t}) + c_3 \cdot r_3 \cdot (G_t - X_{i,t}). \quad (3.2)$$

This vector is a function of three components: previous velocity of the particle, the best experience of the particle, also, the entire swarm's best experiences up to the current iteration which are called inertia, cognition part, and social part, respectively [16].

Updating process continues until the termination criterion is met which usually is the maximal number of generations, processing time, or the best particle position of the whole swarm that cannot improve further after a predefined number of generations.

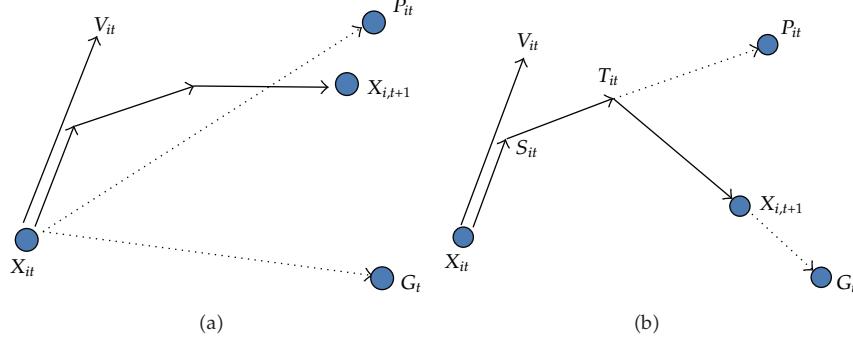
In (3.2),  $r_2$  and  $r_3$  are real random numbers with uniform distribution which are usually selected from the interval  $[0, 1]$ .  $c_2$  and  $c_3$  are known constants as learning factors, showing the significance of local and global best experiences, respectively. Also,  $c_1$  is defined as a positive inertia weight which was first introduced by Shi and Eberhart [17]. This parameter can be specific for each particle [18]. Liu et al. [19] and Shi and Eberhart [20] introduced time-decreasing inertia weight.

The PSO parameters analyses have been the subject of several researches. For instance, Tchomté and Gourgand [14] determined some conditions for parameters to ensure that each particle converges to some equilibrium point after enough number of iterations.

Although PSO has been originally designed for solving continuous problems, it can be used for solving discrete problems as well. Different techniques have been designed to use this algorithm for combinatorial optimization problems such as the ones introduced by Jarboui et al. [8], Clerc [21], and Kennedy and Eberhart [22].

In this paper, we use the modified PSO approach which was introduced by Tchomté and Gourgand [14] as an extension of PSO that integrates a new displacement rule of the particles. The computational results of their algorithm showed that their PSO algorithm outperformed all state-of-the-art algorithms in solving RCPSP, and this is the reason for selecting their approach for our problem. We describe this modified PSO method in the following.

A metaheuristic algorithm should be able to explore search space effectively and efficiently. PSO algorithm should be intelligent enough to both intensively explore regions of the search space with high-quality solutions and to diversely move to unexplored regions of the search space. These two techniques that were introduced by Glover and Laguna [23] are known as intensification and diversification methods, respectively. Tchomté and Gourgand [14] analyzed particle trajectories and modified particle position updating rules. The idea originated from the PSO applications in which particles basically move from their current positions toward the best local and global positions ( $P_{i,t}, G_t$ ), but the particles do not get close enough to  $P_{i,t}$  and  $G_t$ . As a result, diversification is performed well, but intensification is not. Consequently, Tchomté and Gourgand [14] proposed a new particle displacement rule to improve the intensification process by letting each particle visit two positions  $S_{i,t}$  and  $T_{i,t}$  before moving from current position,  $X_{i,t}$ , to the next position  $X_{i,t+1}$ . First, the inertia has influences on the position by making the particle move from  $X_{i,t}$  to  $S_{i,t}$ . Then the cognition part moves the particle to  $T_{i,t}$  and finally under social part affect the particle to reach its new position,  $X_{i,t+1}$ , at the next iteration. Adapted from Tchomté and Gourgand [14], particle displacement in the classical PSO and this modified PSO has been shown in Figures 1(a) and 1(b), respectively.



**Figure 1:** Particle displacement in the swarm: (a) classical PSO, (b) modified PSO.

For this purpose, the particle position updating rule is

$$\begin{aligned}
 S_{i,t} &= X_{i,t} + c_1 \cdot V_{i,t}, \\
 T_{i,t} &= S_{i,t} + c_2 \cdot r_2 \cdot (P_{i,t} - S_{i,t}), \\
 X_{i,t+1} &= T_{i,t} + c_3 \cdot r_3 \cdot (G_t - T_{i,t}), \\
 X_{i,t+1} &= X_{i,t} + c_1 \cdot (1 - c_2 \cdot r_2) \cdot (1 - c_3 \cdot r_3) \cdot V_{i,t} \\
 &\quad + c_2 \cdot r_2 \cdot (1 - c_3 \cdot r_3) \cdot (P_{i,t} - X_{i,t}) + c_3 \cdot r_3 \cdot (G_t - X_{i,t}). \tag{3.3}
 \end{aligned}$$

Then, in each time step  $t$ , velocity vector is updated as follows:

$$V_{i,t+1} = \alpha \cdot V_{i,t} + \beta \cdot (P_{i,t} - X_{i,t}) + \gamma \cdot (G_t - X_{i,t}), \tag{3.4}$$

$$\alpha = c_1 \cdot (1 - r_2 \cdot c_2) \cdot (1 - r_3 \cdot c_3), \tag{3.5}$$

$$\beta = c_2 \cdot (1 - r_3 \cdot c_3), \tag{3.5}$$

$$\gamma = c_3.$$

Tchomté and Gourgand [14] showed that the necessary conditions for coefficients so that the particles converge to the equilibriums are satisfying (3.6) plus (3.7) or (3.6) plus (3.8):

$$\phi > 0, \quad \phi - 2 * (c_1 + 1) < 0, \quad c_1 < 1, \tag{3.6}$$

where  $\phi = (c_2 + c_3)/2$ ,

$$0 < c_1 < 0.9, \quad 0 < c_2 < 2, \quad 0 < c_3 < 2, \tag{3.7}$$

$$0 < c_1 < 0.9, \quad 2 \leq c_2 < 4, \quad 2 \leq c_3 < 4. \tag{3.8}$$

#### 4. Modified PSO for MRCPS-TWRTPC

In this section, we present a modified PSO algorithm, using the approach of Tchomté and Gourgand [14], for solving MRCPS-TWRTPC. Algorithm 1 shows the pseudocode. In this algorithm the  $i$ th particle position at iteration  $t$  is represented by the  $n$ -dimensional vector

```

(1) Do Preprocessing
(2) Generate initial particle swarm
(3) While termination criterion is met do
(4)   While all particles have been evaluated do
(5)     Determine activities priorities
(6)     Schedule activities based on their modes and priorities using the parallel
          schedule generation and delay local search
(7)     While schedule is improved do
(8)       Improve schedule by Mode Assignment Modification—Part I
(9)       Improve schedule by Local Left Shift
(10)      End while
(11)      Improve schedule by Mode Assignment Modification—Part II
(12)      Compute corresponding cost of the generated schedule
(13)    End while
(14)    Update the local and global best solutions if necessary
(15)    Update position and velocity of each particle according to (3.3) and (3.4), respectively
(16)  End while
(17) Report the global best solution

```

**Algorithm 1:** Pseudocode of modified PSO algorithm for MRCPS-TWRTPC.

$X_{i,t} = (x_{i1}^t, x_{i2}^t, \dots, x_{in}^t)$  in which  $x_{ij}^t$ ,  $j = 1, 2, 3, \dots, n$  is the mode assignment to the  $j$ th activity at iteration  $t$  and is an integer in the interval  $[1, M_j]$ . A feasible schedule of the project is constructed from each  $X_{i,t}$ . For this purpose, first the activities are prioritized, see Section 4.3. Then, using single-pass parallel schedule generation scheme, the activities are scheduled, see Section 4.4. Certain local search and improvement procedures are applied on this solution to reach a better schedule, see Sections 4.5, 4.6, 4.7, and 4.8.

Each particle's fitness value is determined by calculating total cost of the final schedule. Then, if necessary, local and/or global best positions are updated, see Section 4.9. If termination criterion is not met, particle positions and velocity vectors are updated by (3.3) and (3.4), respectively, for the next iteration, see Section 4.10. Different parts of this algorithm are described in more details as follows.

#### 4.1. Preprocessing

Sprecher et al. [24] introduced several preprocessing rules in order to reduce feasible space of MRCPS. Later, these rules have been used in other articles such as Lova et al. [6], Peteghem and Van Vanhouck [10], and Hartmann and Briskorn [25]. Considering the similarities between MRCPS-TWRTPC and MRCPS, we apply two of these rules to our proposed problem. One is the *nonexecutable mode elimination* rule for an activity. For a *nonexecutable mode*, the amount of the resource needed for executing the activity is more than the resource availability. Another method is *inefficient mode elimination* method. A given mode is *inefficient* for an activity if there is another mode for which the activity duration is less, and that activity can be accomplished with less total amount of both renewable and nonrenewable resources. Therefore, activities modes are analyzed one by one and nonexecutable and inefficient modes are deleted.

#### 4.2. Generating Initial Particle Swarm

Initial particle swarm is generated randomly. Here, component  $j$ ,  $j = 1, \dots, n$ , of either position or velocity vectors for each particle is generated randomly from the ranges  $[1, M_j]$  and

$[-M_j, M_j]$ , respectively, as there is no nonrenewable resource constraint in the problem; moreover, all nonexecutable modes have been deleted, initial mode assignments are feasible, and no modification is needed.

### 4.3. Activity Priority for Scheduling

In order to generate a solution in MRCPSP-TWRTPC, two issues are to be decided: activities mode assignment and scheduling of activities. By specifying mode assignment for a solution, the cost of nonrenewable resources is determined and fixed. Then, scheduling of activities is performed with the objective of minimizing the total cost of renewable resource tardiness penalties. Therefore, the priorities of activities for scheduling are determined by renewable resources. Our procedure for this purpose is as follows.

First, we define the set of activities which need renewable resource  $k$  as the *activity set of resource  $k$*  ( $\text{ASR}_k$ ). Each activity in this set may have immediate or nonimmediate predecessors that may not be a member of this set. We define the set of these predecessors which are not members of  $\text{ASR}_k$  as *activities predecessors of resource  $k$*  ( $\text{APR}_k$ ). Then, the pairs of  $\text{ASR}_k$  and  $\text{APR}_k$ ,  $k = 1, \dots, R$ , are prioritized by index  $k$  using the heuristic that activities in  $\text{ASR}_k$  and  $\text{APR}_k$  for the resource which has more potential of causing tardiness penalty should receive higher priority of being scheduled. To access the potential of the  $k$ th resource tardiness penalty cost, we note that this penalty cost is equal to  $P_k \times \max\{0, CP_k - d_k\}$ , where  $CP_k$  is the release time of resource  $k$  in the project, and hence  $P_k \times (CP_k - d_k)$  is a good measure for prioritizing the resources for this purpose. Now the question is how to access  $CP_k$  without knowing the schedule. We use the following procedure to access  $CP_k$ ,  $k = 1, \dots, R$ .

Since no activity can start sooner than the ready time of all the resources it needs, in order to take into account the resource ready time, we add one dummy node  $k$  for each resource  $k$ ,  $k = 1, \dots, R$ , to the project network. Each dummy activity  $R$  is single mode with no resource requirement, and the duration of  $r_k$ . All these dummy activities can be scheduled at time zero. Then, for any activity  $j$  which needs renewable resource  $k$ , we set dummy activity  $k$  as one of its predecessors. So execution of activity  $j$  is not possible before the time  $r_k$ . Note that dummy activity  $k$  is a member of  $\text{APR}_k$ . The length of the critical path of subnetwork  $k$  is denoted by  $CP_k$  and is considered as a relative measure of  $CP_k$  for  $k = 1, \dots, R$ .  $CP_k$  is computed using CPM method. After computation of  $CP_k$  for all resources, the parts of activity sets  $\text{ASR}_k$  and  $\text{APR}_k$ ,  $k = 1, \dots, R$ , are prioritizing using the value of  $P_k \times (CP_k - d_k)$ .

In our procedure for prioritizing activities for scheduling, we give more priority to the activities in  $\text{APR}_k$  than activities in  $\text{ASR}_k$ , since activities in  $\text{ASR}_k$  cannot be scheduled unless activities in  $\text{APR}_k$  are scheduled.

Finally, for each resource  $k$ , it is necessary to prioritize the activities belonging to each set of  $\text{ASR}_k$  and  $\text{APR}_k$ . We notice when a set of activities using a renewable resource are to be scheduled, we actually deal with an RCPSP, because activities under specified modes are to be scheduled in order to execute within the shortest possible time. Therefore, in order to prioritize activities of a set of  $\text{ASR}_k$  or  $\text{APR}_k$ , we apply one of the most efficient heuristics to scheduling of activities in RCPSP. Lova et al. [6] compared a number of most efficient heuristics for prioritization of activities in RCPSP and found out that activities prioritization based on nondecreasing order of the sum of the latest start and finish time (LSTLFT) gained the best results among single-pass heuristics. This has been among the best multipass methods as well. Multipass methods need much more computation times than single-pass methods, but they usually result in negligible improvement of the solution; hence, we use the LSTLFT method with single-pass scheduling in order to prioritize activities of  $\text{ASR}_k$  and  $\text{APR}_k$ .

#### 4.4. Scheduling Activities

We use parallel schedule generation scheme for scheduling activities, since it fits well with *delay local search* which we will use. For more details on parallel schedule generation scheme see Section 6.3.2.1.2 of Demeulemeester and Herroelen [1].

Parallel schedule generation scheme repeats over the separate decision points at which activities can be added to the schedule. A decision point in time horizon corresponds with the completion times of already scheduled activities; thus, at most  $n$  decision points need to be considered. At each decision point, the unscheduled activities whose predecessors have been accomplished are taken into consideration in the order of priority list and are scheduled if no resource conflict exists at that time instant.

In this problem, renewable resources have ready time and availability of them differs before and after these times and some new activities may be able to be scheduled after these times. Therefore, we consider ready times in addition to the completion times of activities for choosing new point in time horizon. A new point in time horizon is the closest point to the current point among the ready times of renewable resources and the completion time of scheduled activities.

#### 4.5. Delay Local Search

This local search method was used by Chen et al. [26] for the RCPSP problem to escape from local minimum. This method performs like the mutation operator in genetic algorithm and can delay scheduling each activity in spite of its priority to let other activities be scheduled sooner and some resources asked by selected activity will be retained for other activities.

In order to use resources more efficiently, scheduling some activities are delayed in spite of their priority. So other activities can be scheduled sooner. If these selected activities are not delayed, they could delay other project activities for a rather long time. Therefore, each activity is delayed if  $q \leq q_0$ , where  $q$  is randomly selected form uniform distribution  $[0, 1]$  and  $q_0$  ( $0 < q_0 < 1$ ) is the probability of delay and called “delay rate.”

Considering the efficiency of this delay local search in shortening project makespan shown in Chen et al. [26], we apply this method to scheduling activities.

#### 4.6. Mode Assignment Modification-Part I

After scheduling activities, the current schedule may have a set of activities with positive free floats. We call this set FFA. For each  $j \in FFA$ , it may be possible to change the mode of activity  $j$  and reschedule it, using its free float, such that its finish time is not increased and hence no other activity is affected. The mode change and rescheduling of activity  $j$  can reduce nonrenewable resource costs, but it can also change  $CP_k$ , release time of resource  $k$ ,  $k = 1, \dots, R$ . The change of  $CP_k$  can change renewable resource  $k$  tardiness,  $k = \max\{0, CP_k - d_k\}$ , and its cost,  $p_k l_k$ . If we set the availability of resource  $k$ ,  $k = 1, \dots, R$ , for the periods after  $\max\{CP_k, d_k\}$  equal to zero and then reschedule activity  $j$ , we are sure that this scheduling is not going to increase renewable resource tardiness penalties. Considering the above points, we define “mode assignment modification-part I” as follows and use it as a local search procedure in our algorithm.

- (1) For the current schedule find the set FFA by forward pass computation.
- (2) Set the availability of resource  $k$ ,  $k = 1, \dots, R$ , for the periods after  $\max\{CP_k, d_k\}$  equal to zero.

- (3) For each  $j \in FFA$ , as we go through forward pass, consider the least nonrenewable resource cost mode of activity  $j$ . If it is not its current mode, reschedule activity  $j$  in this mode using its free float and considering resource constraints, without increasing its finish time. If this rescheduling is not possible, check the next least nonrenewable resource cost mode of activity  $j$ .

#### **4.7. Local Left Shift Improvement**

Mode assignment modification-part I can reduce the renewable resource requirements of the project for certain periods under the resulted schedule. This should be expected since usually the mode with less nonrenewable resources has longer duration and requires less renewable resources per period. Hence, the possibility of local left shift of certain activities exists. Therefore, we perform the standard local left shift method after the mode assignment modification-part I.

After performing local left shift, we might be able to modify some of the activities mode assignment again by mode assignment modification-part I. Hence, these two procedures are used one after another until no improvement can be achieved in the schedule.

#### **4.8. Mode Assignment Modification-Part II**

We consider the set of project activities with no successors and call this set NSA. The direct predecessor activities of dummy activity  $n$  make NSA. For any  $j \in NSA$  if we change the mode of activity  $j$  and reschedule it, the schedule of no other activity changes and the value of cost change of the project is  $\Delta c = \Delta NRC_j + \sum_{k=1}^R p_k \Delta l_k$ , where  $\Delta NRC_j$  is the change of nonrenewable resource cost of mode for activity  $j$  and  $\Delta l_k$  is the change of  $k$ th resource tardiness. Since activity  $j$  has no successor,  $\Delta l_k$  can be computed easily. If the value of change effect on total cost is negative, the mode change for activity  $j \in NSA$  is justifiable. Considering above points, we define the following local search procedure as “mode assignment modification-part II” and use it in our algorithm.

- (1) Let  $NSA = \{j \mid \text{activity } j \text{ is the direct predecessor of dummy activity } n\}$ .
- (2) For each  $j \in NSA$ , consider some modes for activity  $j$ , in which nonrenewable resources requirement cost is less and this cost saving is more than the probable increase in the penalty cost of renewable resources. Compute  $\Delta c$  for each of them. Considering renewable resource constraints, if the least  $\Delta c$  is negative, its corresponding mode replaces the current mode of activity  $j$ .

#### **4.9. Updating the Local and Global Best Solutions**

As mentioned earlier, the PSO algorithm stores the best local solution obtained by each particle and the best global solution obtained by the entire swarm. Therefore, after evaluating all the particles of the swarm at iteration, the best local solution for each particle up to the current iteration is compared with the current solution of the particle. If the current solution has lower total cost, the best local solution of the particle is replaced by the current solution of the particle.

Similarly, the best global solution is updated.

#### **4.10. Updating Particles Position and Velocity**

To update position of the particles to generate new solutions for the next iteration, firstly, particles velocity is updated using (3.4). In this process each element of the velocity vector which is more than  $M_i$  is changed to  $M_i$  and each element of the velocity vector which is less than  $-M_i$  is changed to  $-M_i$ . Then, particles positions are updated using (3.3). Similar to the procedure for updating each particle velocity, each element of the new position vector which is more than  $M_i$  is changed to  $M_i$  and each element of the new position vector which is less than 1 is changed to 1. As the elements of position vector determine the mode assignment of activities and should be integer, each noninteger element is rounded to the closest integer.

### **5. Experimental Analysis**

In this section, we present experimental analysis of the algorithm. All programs have been coded and executed on C#.NET 2008 platform on a PC with Core 2 Duo 2.53 GHz CPU and 3 GB RAM.

#### **5.1. Sample Problems**

We used sample problems library of PSPLIB [27] and selected three sets of multimode project scheduling problems, j10, j16, j20 as the small size problems and j30 as the medium size problem. In addition, two sets of large size problems, j60 and j90, were generated with the same parameters as j30, but with 60 and 90 activities, respectively. Also, in order to observe the effect of resources in the problem, two extra sets of project scheduling problems were generated by Kolisch et al. [28], which we call j30\_r4\_n4 and j60\_r4\_n4. All the parameters in these sets are similar to those in sets j30 and j60, respectively, but instead of having 2 renewable and 2 nonrenewable resources, there exist 4 resources of each type.

Discrete uniform distribution has been used in the related literature of the project scheduling; for example, Ranjbar et al. [2] and Khalilzadeh et al. [29] used discrete uniform distribution for resource ready dates and due date. Hence, in this paper we have used discrete uniform distribution to select the parameters. The unit cost of nonrenewable resources were randomly selected from discrete uniform distribution (2,6), the unit penalty cost of renewable resource tardiness were randomly chosen from discrete uniform distribution (10,30). The ready times of renewable resources were randomly generated from discrete uniform distribution (0,15), and finally, the renewable resource due dates were randomly picked from discrete uniform distribution (5,15) plus the amount of their ready time.

#### **5.2. Parameters Tuning**

There exist five parameters, delay rate ( $q_0$ ), the number of particles in the swarm  $P$ , and also,  $c_1$ ,  $c_2$ , and  $c_3$  in our PSO algorithm. For setting these parameters we used 3-point factorial design as shown in Table 1. As mentioned in Section 3, parameters  $c_1$ ,  $c_2$ , and  $c_3$  ought to be chosen in the ranges given by (3.6) and (3.7), or (3.6) and (3.8). For each of these two ranges, three points have been selected.

A set of 100 test instances was randomly selected from the set j16 and solved with the CPU time limit of 150 milliseconds. Subsequently, around 4 to 30 iterations were executed, depending on the number of particles in the swarm.

**Table 1:** Parameter settings.

Parameter	Levels
$q_0$	0.2–0.4–0.6
$P$	10–30–60
$c_1$	0.2–0.45–0.7
	Satisfying relations (3.6) and (3.7)
$c_2$	0.5–1–1.5
$c_3$	0.5–1–1.5
	Satisfying relations (3.6) and (3.8)
	2.5–3–3.5
	2.5–3–3.5

**Table 2:** Algorithm validity assessment.

Problems set	Average CPU time (millisecond)	Average $d$	Standard deviation of $d$
J10 (536 problems)	278	1.19	2.18
J16 (550 problems)	446	3.64	4.75
J20 (554 problems)	543	4.70	5.65
J30 (640 problems)	825	7.40	8.13
J60 (640 problems)	1818	10.56	11.06
J90 (640 problems)	2916	11.76	12.21
J30-r4-n4 (640 problems)	1509	7.78	8.73
J60-r4-n4 (640 problems)	3302	11.69	12.56

Each test instance was run for all  $3^5 \times 2 = 486$  permutations of parameter values, a total of 48,600 problems. The tuned values of the parameters are  $q_0 = 0.4$ ,  $P = 60$ ,  $c_1 = 0.2$ ,  $c_2 = 0.5$ ,  $c_3 = 1$ .

### 5.3. Algorithm Validity

As our research is new, we could not find any solved problem in the literature. So we developed some instance problems whose optimum objective function values are in hand. Then we solved these instances with our algorithm and tested the results. In order to generate these instances, we used sample problems introduced in Section 5.1; however, we modified the due dates of renewable resources as follows. We generated a random feasible schedule for each instance, after assigning the least nonrenewable cost mode to each activity of the project. In this schedule, we determined the release time of each renewable resource. Subsequently, we set the due date for each renewable resource equal to its release time; hence, this schedule has zero tardiness penalty cost and its objective function value is equal to the cost of nonrenewable resources. As we assigned the least nonrenewable cost modes to the activities, this schedule is optimal and the optimum value of its objective function is available.

All sample problems were modified with the above procedure and solved with the PSO algorithm. The termination criterion was generation of 600 schedules. In order to assess the validity of the algorithm,  $d$ , the percent deviation of the objective function value from optimum, computed for each test problem solved, where  $d = 100 \times (Z_p - Z_{\text{opt}})/Z_{\text{opt}}$ ,  $Z_p$  is the objective function value of the best solution achieved by the PSO algorithm and  $Z_{\text{opt}}$ , the optimal objective function value of the instance. Table 2 shows the average and standard deviation of  $d$  for each problem set. The low values of average and standard deviation of  $d$  reveal good performance of the algorithm with small CPU time, although we do not have any standard for comparison.

**Table 3:** Algorithm robustness check.

Problem set	120 schedules		900 schedules	
	Average CPU time (millisecond)	Average percent deviation	Average CPU time (millisecond)	Average percent deviation
j10	69	6.18	352	5.17
j16	112	4.74	541	5.01
j20	135	4.14	648	4.14
J30	216	4.17	1039	3.65
J60	471	3.44	2220	3.35
J90	776	2.13	3652	2.08
J30-r4-n4	410	4.13	1893	3.29
J60-r4-n4	880	2.77	4085	2.76

#### 5.4. Algorithm Robustness

In order to check the robustness of the PSO algorithm, we have used the algorithm for several instances. Each instance has been solved several times, and  $d$  the percent deviation of objective function value for each instance has been computed.

From each set of problems, 15 randomly chosen instances have been used and each instance has been solved 30 times using the PSO algorithm, and each time done under two termination criteria, generation of 120 and 900 schedules.

To check the robustness of the algorithm,  $d'$  for each instance has been computed, where  $d' = 100 \times Sd(Z_i) / E(Z_i)$ ,  $Z_i$  is the objective function value of the best solution achieved by solving the problem for  $i$ th run,  $Sd(Z_i)$  standard deviation of  $Z_i$ , and  $E(Z_i)$  is the mean of  $Z_i$ . The average of  $d'$  for the problems of each set has been shown in Table 3.

#### 5.5. Improvement Methods Performance Assessment

In this section, we assess the performance of the improvement methods introduced in Sections 4.6 and 4.8. The first one is the mode assignment improvement method-part I along with local left shift, and the other one is the mode assignment improvement method-part II. In order to assess each of these improvement methods, we remove each one from the original algorithm to get two simplified algorithms, each of which does not have one of the two improvement methods. We use 30 instances from each problem set and solve them with the main algorithm, also, with the two simplified algorithms. Subsequently, compare the results gained by simplified algorithms with the results of the original algorithm. All instances have been solved by three algorithms under three different termination criteria, to generation of 120, 600, and 900 schedules. We compute  $d''$  for each instance to compare the results, where  $d'' = 100 \times (Z_s - Z_p) / Z_p$ ,  $Z_s$  is the final solution objective function value obtained by the simplified PSO algorithm,  $Z_p$ , final solution objective function value obtained by original PSO, and  $d''$  is the percent deviation of  $Z_s$  from  $Z_p$ .

Mean and standard deviation of  $d''$  for each set have been computed. Table 4 shows the effect of deleting mode assignment improvement method-part I and local left shift and we can see that, in all cases, the performance of the algorithm deteriorates remarkably and in most cases the mean CPU time, in millisecond, considerably increases if this local search is deleted.

**Table 4:** Performance assessment of improvement part I.

Problem set	120 schedules				600 schedules				900 schedules			
	Average CPU time of original algorithm (millisecond)	Average $d$	Standard deviation of $d$	Average CPU time of original algorithm (millisecond)	Average $d$	Standard deviation of $d$	Average CPU time of original algorithm (millisecond)	Average $d$	Standard deviation of $d$	Average CPU time of simplified algorithm (millisecond)	Average $d$	Standard deviation of $d$
J10	87	85	6.43	14.13	270	335	7.27	15.77	385	487	15.90	27.79
J16	121	133	10.32	16.65	707	523	4.54	9.11	605	791	6.66	12.75
J20	157	171	10.61	15.71	515	654	10.43	17.49	725	983	12.63	21.84
J30	233	247	5.41	9.93	787	1005	8.58	14.75	1120	1488	10.41	14.92
J60	499	541	10.46	12.29	1658	2189	8.95	10.72	2419	3257	10.33	13.26
J90	819	886	10.05	11.57	2676	3531	11.23	11.83	3960	5293	11.82	13.04
J30-r4-n4	437	476	9.43	13.34	1423	1964	8.81	13.55	2082	2909	11.71	13.81
J60-r4-n4	887	1003	12.12	13.94	2950	7089	12.72	15.03	4294	6084	13.13	14.78

**Table 5:** Performance assessment of improvement part II.

Problem set	120 schedules				600 schedules				900 schedules			
	Average CPU time of original algorithm (millisecond)	Average $d$	Standard deviation of $d$	Average CPU time of original algorithm (millisecond)	Average $d$	Standard deviation of $d$	Average CPU time of original algorithm (millisecond)	Average $d$	Standard deviation of $d$	Average CPU time of simplified algorithm (millisecond)	Average $d$	Standard deviation of $d$
J10	87	112	0.61	8.65	270	429	11.92	3.32	385	646	4.37	14.77
J16	121	175	2.45	7.62	707	676	5.47	10.20	605	1032	2.04	8.66
J20	157	220	1.50	7.41	515	842	0.38	5.85	725	1231	0.66	7.96
J30	233	341	-1.52	8.06	787	1313	1.68	5.29	1120	1922	2.16	6.63
J60	499	730	-0.54	3.19	1658	2858	-0.55	3.74	2419	4208	0.68	4.25
J90	819	1167	0.06	3.55	2676	4575	-0.73	4.32	3960	6752	-0.93	2.84
J30-r4-n4	437	618	-0.70	6.74	1423	2413	-0.04	3.61	2082	3611	0.39	4.27
J60-r4-n4	887	1322	0.77	3.89	2950	5102	1.34	4.70	4294	7505	-0.62	3.45

Table 5 shows the effect of deleting mode assignment improvement method-part II. We can see that in most cases the performance of the algorithm deteriorates, and in all cases the average CPU time increases remarkably. In the following, we explain the reason for increasing the average CPU time.

## 6. Conclusions

In this paper, we introduced MRCPS-TWRTPC problem as a resource-oriented cost minimization project scheduling problem considering both renewable and nonrenewable resource costs. We formulated and mathematically modeled this problem as mixed integer programming model and discussed its NP-hardness. Subsequently, we developed a metaheuristic algorithm to tackle the proposed project scheduling problem. We briefly reviewed the applications of the PSO algorithm for solving combinatorial and constrained optimization problems. Thereafter, we applied a modified PSO algorithm including modified updating rules for particles velocity and position. In order to generate feasible schedules, we used the PSO algorithm for activity mode assignment and developed a novel heuristic technique to prioritize activities for parallel scheduling scheme. Two improvement heuristics, delay local search and local left shift, in line with two mode assignment modification methods, were implemented to improve the solutions. The computational results revealed proper algorithm robustness in solving different instances especially with high number of iterations. Also, the validity analysis showed small deviations from the optimal solutions for the test instances in reasonable solving time. Finally, we assessed two improvement methods used in our algorithm to demonstrate their good performance.

## References

- [1] E. Demeulemeester and W. S. Herroelen, *Project Scheduling, A Research Handbook*, Kluwer Academic, Dordrecht, The Netherlands, 2001.
- [2] M. Ranjbar, M. Khalilzadeh, F. Kianfar, and K. Etminani, "An optimal procedure for minimizing total weighted resource tardiness penalty costs in the resource-constrained project scheduling problem," *Computers and Industrial Engineering*, vol. 62, no. 1, pp. 264–270, 2012.
- [3] R. Heilmann, "A branch-and-bound procedure for the multi-mode resource-constrained project scheduling problem with minimum and maximum time lags," *European Journal of Operational Research*, vol. 144, no. 2, pp. 348–365, 2003.
- [4] G. Zhu, J. F. Bard, and G. Yu, "A branch-and-cut procedure for the multimode resource-constrained project-scheduling problem," *INFORMS Journal on Computing*, vol. 18, no. 3, pp. 377–390, 2006.
- [5] H. Zhang, C. M. Tam, and H. Li, "Multimode project scheduling based on particle swarm optimization," *Computer-Aided Civil and Infrastructure Engineering*, vol. 21, no. 2, pp. 93–103, 2006.
- [6] A. Lova, P. Tormos, and F. Barber, "Multi-mode resource constrained project scheduling: scheduling schemes, priority rules and mode selection rules," *Inteligencia Artificial*, vol. 10, no. 30, pp. 69–86, 2006.
- [7] A. Lova, P. Tormos, M. Cervantes, and F. Barber, "An efficient hybrid genetic algorithm for scheduling projects with resource constraints and multiple execution modes," *International Journal of Production Economics*, vol. 117, no. 2, pp. 302–316, 2009.
- [8] B. Jarboui, N. Damak, P. Siarry, and A. Rebai, "A combinatorial particle swarm optimization for solving multi-mode resource-constrained project scheduling problems," *Applied Mathematics and Computation*, vol. 195, no. 1, pp. 299–308, 2008.
- [9] M. Ranjbar, B. De Reyck, and F. Kianfar, "A hybrid scatter search for the discrete time/resource trade-off problem in project scheduling," *European Journal of Operational Research*, vol. 193, no. 1, pp. 35–48, 2009.
- [10] V. Van Peteghem and M. Vanhoucke, "A genetic algorithm for the preemptive and non-preemptive multi-mode resource-constrained project scheduling problem," *European Journal of Operational Research*, vol. 201, no. 2, pp. 409–418, 2010.

- [11] F. S. Kazemi and R. Tavakkoli-Moghaddam, "Solving a multi-objective multi-mode resource-constrained project scheduling problem with particle swarm optimization," *International Journal of Academic Research*, vol. 3, no. 1, pp. 103–110, 2011.
- [12] J. Błażewicz, J. K. Lenstra, and A. H. G. Rinnooy Kan, "Scheduling subject to resource constraints: classification and complexity," *Discrete Applied Mathematics*, vol. 5, no. 1, pp. 11–24, 1983.
- [13] P. Brucker, A. Drexel, R. Mohring, K. Neumann, and E. Pesch, "Resource-constrained project scheduling: notation, classification, models and methods," *European Journal of Operational Research*, vol. 113, pp. 3–41, 1999.
- [14] S. Kemmoé Tchomté and M. Gourgand, "Particle swarm optimization: a study of particle displacement for solving continuous and combinatorial optimization problems," *International Journal of Production Economics*, vol. 121, no. 1, pp. 57–67, 2009.
- [15] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, Piscataway, NJ, USA, December 1995.
- [16] Y. Shi and R. Eberhart, "Modified particle swarm optimizer," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '98)*, pp. 69–73, Piscataway, NJ, USA, May 1998.
- [17] Y. Shi and R. Eberhart, "Parameter selection in particle swarm optimization," in *Annual Conference Evolutionary Programming*, San Diego, Calif, USA, 1998.
- [18] C. Y. Tsai and S. W. Yeh, "A multiple objective particle swarm optimization approach for inventory classification," *International Journal of Production Economics*, vol. 114, no. 2, pp. 656–666, 2008.
- [19] S. Liu, J. Tang, and J. Song, "Order-planning model and algorithm for manufacturing steel sheets," *International Journal of Production Economics*, vol. 100, no. 1, pp. 30–43, 2006.
- [20] Y. Shi and R. Eberhart, "Empirical study of particle swarm optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 945–1950, Piscataway, NJ, USA, 1999.
- [21] M. Clerc, "Discrete particle swarm optimization," in *New Optimization Techniques in Engineering*, G. C. Onwubolu and B. V. Babu, Eds., pp. 204–219, Springer, Berlin, Germany, 2004.
- [22] Kennedy and R. C. Eberhart, "Discrete binary version of the particle swarm algorithm," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 5, pp. 4104–4108, IEEE Computer Society, Washington, DC, USA, 1997.
- [23] F. Glover and M. Laguna, "Tabu search," in *Handbook of Combinatorial Optimization*, D.-Z. Du and P. M. Pardalos, Eds., vol. 3, pp. 621–757, Kluwer Academic, Boston, Mass, USA, 1998.
- [24] A. Sprecher, S. Hartmann, and A. Drexl, "An exact algorithm for project scheduling with multiple modes," *OR Spektrum*, vol. 19, no. 3, pp. 195–203, 1997.
- [25] S. Hartmann and D. Briskorn, "A survey of variants and extensions of the resource-constrained project scheduling problem," *European Journal of Operational Research*, vol. 207, no. 1, pp. 1–14, 2010.
- [26] R. M. Chen, C. L. Wu, C. M. Wang, and S. T. Lo, "Using novel particle swarm optimization scheme to solve resource-constrained scheduling problem in PSPLIB," *Expert Systems with Applications*, vol. 37, no. 3, pp. 1899–1910, 2010.
- [27] R. Kolisch and A. Sprecher, "PSPLIB—a project scheduling problem library," *European Journal of Operational Research*, vol. 96, no. 1, pp. 205–216, 1997.
- [28] R. Kolisch, A. Sprecher, and A. Drexl, "Characterization and generation of a general class of resource-constrained project scheduling problems," *Management Science*, vol. 41, pp. 693–1703, 1995.
- [29] M. Khalilzadeh, F. Kianfar, and M. Ranjbar, "A Scatter Search Algorithm for RCPSP with discounted weighted earliness-tardiness costs," *Life Science Journal*, vol. 8, no. 2, pp. 634–640, 2011.

*Letter to the Editor*

## **Comment on “Highly Efficient Sigma Point Filter for Spacecraft Attitude and Rate Estimation”**

**Baiqing Hu, Lubin Chang, An Li, and Fangjun Qin**

*Department of Navigation Engineering, Naval University of Engineering, Wuhan 430033, China*

Correspondence should be addressed to Lubin Chang, changlubin@163.com

Received 13 September 2011; Accepted 3 January 2012

Copyright © 2012 Baiqing Hu 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 light of the intuition that a better symmetrical structure can further increase the numerical accuracy, the paper by Fan and Zeng (2009) developed a new sigma point construction strategy for the unscented Kalman filter (UKF), namely, geometric simplex sigma points (GSSP). This comment presents a different perspective from the standpoint of the numerical integration. In this respect, the GSSP constitutes an integration formula of degree 2 with equal weights. Then, we demonstrate that the GSSP can be derived through the orthogonal transformation from the basic points set of degree 2. Moreover, the method presented in this comment can be used to construct more accurate sigma points set for certain dynamic problems.

With the intuition that a better symmetry property provides a better numerical behavior [1], addressed the construction strategies to make the best symmetric structure in simplex sigma point set and derived the so-called geometric simplex sigma points (GSSP) for Euclidean geometric space. As compared with the previously exiting simplex sigma points set, the GSSP has a symmetric structure and a lower computational expense, is numerically more accurate, and can be used in a variety of 3-dimensional modeled dynamic problems.

In this comment we will show that the GSSP can also be derived from the integration rule of degree 2. Embedding the Gaussian assumption in the Bayesian filter we can reach the idea that the functional recursion of the Bayesian filter reduces to an algebraic recursion operating only on conditional means and covariances which share the same structure of Gaussian weighted integrals whose integrands are all of the form *nonlinear function*  $\times$  *Gaussian density*. The multidimensional integrals are usually intractable for systems involving nonlinearity, so the recursive estimation problem boils down to how to compute the integrals using approximate methods. There are many well-known numerical integration methods such as Gauss-Hermite quadrature, cubature rules, fully symmetric integration rule, and central-difference-based methods that can be used to handle such integrals [2–4]. The unscented transformation (UT) used in the traditional unscented Kalman filter (UKF) can be interpreted as either fully symmetric integration rule or cubature rule of degree 3. The

simplex UT can also be interpreted as a numerical integration formula of degree 2 [3]. Next we will focus on the numerical integration formula of degree 2 in order to derive the GSSP.

Before getting involved in further details, we first introduce some definitions when constructing the exact monomials rule as follows [3, 4].

*Definition 1.* Consider the monomials of the form  $\prod_{i=1}^d x_i^{\alpha_i}$ , where the powers  $\alpha_i$  are nonnegative integers and  $\sum_{i=1}^d \alpha_i \leq p$ , a rule said to have precision  $p$  if it can integrate such monomials accurately and it is not exact for monomials of degree  $p + 1$ .

The numerical integration formulas are conducted by approximating the integrals with the weighted sum of an elaborately chosen set of points as follows [5–7]:

$$\int_{R^n} g(x) \cdot W(x) \approx \sum_k \alpha_k g(\chi_k), \quad (1)$$

where  $R^n$  is a region in an  $n$ -dimensional, real, Euclidean space,  $x = (x_1, x_2, x_3, \dots, x_n)$  is the state variable,  $\alpha_k$  are constants, and  $\chi_k$  are points in the space. The integral weight is a Gaussian distribution as discussed above. Since an arbitrary Gaussian distribution with mean  $\mu$  and covariance  $\Sigma$  can always be transformed into the unit Gaussian distribution as (see prove in [4])

$$\int g(x) \cdot N(x; \mu, \Sigma) dx = \int g(A\xi + \mu) \cdot N(\xi; 0, I) d\xi, \quad (2)$$

where  $AA^T = \Sigma$  and  $I$  is the identity matrix, we can start by considering the multidimensional unit Gaussian integral. Based on Definition 1, we can construct a rule, which is exact up to degree 2 by determining the weighted points set  $\chi_k$  such that it is exact for selections  $g_i(\xi) = 1$ ,  $g_i(\xi) = \xi_i$ ,  $g_{i,j}(\xi) = \xi_i \xi_j$ ,  $i \neq j$ , and  $g_{i,i}(\xi) = \xi_i^2$ . The true values of the integrals are

$$I_0 = \int 1 \cdot N(\xi; 0, I) d\xi = 1, \quad (3)$$

$$\begin{aligned} I_1 &= \int \xi_i \cdot N(\xi; 0, I) d\xi = 0, \quad i = 1, 2, \dots, n, \\ I_2 &= \int \xi_i^2 \cdot N(\xi; 0, I) d\xi = 1, \quad i = 1, 2, \dots, n, \\ I_{1 \times 1} &= \int \xi_i \xi_j \cdot N(\xi; 0, I) d\xi = 0, \quad i \neq j = 1, 2, \dots, n. \end{aligned} \quad (4)$$

In [5] Stroud had proved that  $n + 1$  is the minimum number of points for equally weighted degree 2 formulas. Let us define  $n + 1$  equally weighted points

$$\chi = (\chi_1, \chi_2, \dots, \chi_{n+1}), \quad (5)$$

where  $\chi_k = (\chi_{k,1}, \chi_{k,2}, \dots, \chi_{k,n})^T$ ,  $k = 1, 2, \dots, n+1$ . In order to calculate (4) accurately using these points through (1), we can get the following equations:

$$\begin{aligned} \frac{1}{n+1} \sum_{k=1}^{n+1} \chi_k &= 0, \\ \frac{1}{n+1} \sum_{k=1}^{n+1} \chi_k \cdot \chi_k^T &= I_n, \end{aligned} \quad (6)$$

where  $I_n$  is the  $n$ -dimensional identity matrix. Any equally weighted points set that fulfills (6) can approximate the unit Gaussian integral accurately up to degree 2. References [6, 7] have presented a basic points set that fulfills such conditions with the form as

$$\begin{aligned} \chi_{k,2r-1} &= \sqrt{2} \cos \frac{2rk\pi}{n+1}, \\ \chi_{k,2r} &= \sqrt{2} \sin \frac{2rk\pi}{n+1}, \end{aligned} \quad (7)$$

$r = 1, 2, \dots, [n/2]$ , and if  $n$  is odd,  $\chi_{k,n} = (-1)^k$ .  $[n/2]$  is the greatest integer not exceeding  $n/2$ . When  $n = 3$ , the basic points set is

$$S_1 = [\chi_1 \mid \chi_2 \mid \chi_3 \mid \chi_4] = \begin{bmatrix} 0 & -\sqrt{2} & 0 & \sqrt{2} \\ \sqrt{2} & 0 & -\sqrt{2} & 0 \\ -1 & 1 & -1 & 1 \end{bmatrix}. \quad (8)$$

Next we will give a theorem through which we can get the GSSP from the basic points set of degree 2.

**Theorem 1.** Assume that  $n+1$  equally weighted points set as that in (5) constitutes an integration formula of degree 2.  $A$  is an  $n \times n$  orthogonal matrix. Then,  $A\chi$  also constitutes an integration formula of degree 2.

*Proof.* By defining a matrix

$$M = \begin{bmatrix} \chi_{1,1} & \chi_{2,1} & \chi_{3,1} & \cdots & \chi_{n+1,1} \\ \chi_{1,2} & \chi_{2,2} & \chi_{3,2} & \cdots & \chi_{n+1,2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \chi_{1,n} & \chi_{2,n} & \chi_{3,n} & \cdots & \chi_{n+1,n} \end{bmatrix}, \quad (9)$$

we can rewrite (6) as

$$MM^T = (n+1)I_n, \quad (10)$$

where  $I_n$  is the  $n$ -dimensional identity matrix.  $A$  is an orthogonal matrix, so

$$AM(AM)^T = AMM^T A^T = (n+1)I_n AA^T = (n+1)I_n. \quad (11)$$

Hence,  $A\chi$  also fulfills (6) which completes the proof.  $\square$

For the 3-dimensional Euclidean space, there are many orthogonal matrixes. Here, we use the direction cosine matrix (DCM) which is widely used in the practical systems such as guidance and navigation [8]. The DCM that rotates an angle  $\phi$  about  $u = [0 \ 0 \ 1]^T$  is

$$C(\phi) = \begin{bmatrix} \cos \phi & \sin \phi & 0 \\ -\sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (12)$$

By a simple computation, it is obvious that

$$S_2 = C\left(\frac{3\pi}{4}\right)S_1 = \begin{bmatrix} 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & 1 \end{bmatrix}. \quad (13)$$

Since all the points share equal weight,  $S_2$  is virtually just the GSSP derived in [1].

Up to this point we have derived the GSSP through the numerical integration formulas method. Compared with the intuitionistic method in [1], our method is more principled in mathematical terms. Although Theorem 1 is proposed for integration formula of degree 2, it can be generalized for different degrees, that is, the orthogonal transformation on the numerical integration formula will not change its accurate degree. Reference [7] also presented the points set of degree 3, that is,

$$\gamma = (\gamma_1, \gamma_2, \dots, \gamma_{2n}), \quad (14)$$

where  $\gamma_k = (\gamma_{k,1}, \gamma_{k,2}, \dots, \gamma_{k,n})^T$ ,  $k = 1, 2, \dots, 2n$  with

$$\begin{aligned} \gamma_{k,2r-1} &= \sqrt{2} \cos\left(\frac{(2r-1)k\pi}{n}\right), \\ \gamma_{k,2r} &= \sqrt{2} \sin\left(\frac{(2r-1)k\pi}{n}\right), \end{aligned} \quad (15)$$

$r = 1, 2, \dots, [n/2]$ , and if  $n$  is odd,  $\gamma_{k,n} = (-1)^k$ .  $[n/2]$  is the greatest integer not exceeding  $n/2$ . It can be proven that the points set (14) can be derived through orthogonal transformation on the cubature points set with the form [4]

$$\lambda = [\sqrt{n}e_k \ -\sqrt{n}e_k], \quad k = 1, \dots, n, \quad (16)$$

where  $e_i$  denotes a unit vector to the direction of coordinate axis  $i$ .

As can be seen from (16) the distance of the cubature point from the mean is proportional to  $\sqrt{n}$ . So, for high-dimensional problems, the cubature points set bears the nonlocal sampling problem [9–11]. For many kinds of nonlinearities (such as exponents or trigonometric functions), this can lead to significant difficulties. In contrast, the points set (14) does not bear such nonlocal sampling problem. Under this condition, the points set (14) is more accurate than the cubature points set (16). Therefore, to this respect, we can derive different sigma points set which may be more accurate for certain dynamic problems by the presented method.

## References

- [1] C.-S. Fan and Y. Zheng, "Highly efficient sigma point filter for spacecraft attitude and rate estimation," *Mathematical Problems in Engineering*, vol. 2009, Article ID 507370, 23 pages, 2009.
- [2] K. Ito and K. Xiong, "Gaussian filters for nonlinear filtering problems," *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 910–927, 2000.
- [3] Y.-X. Wu, D.-W. Hu, M.-P. Wu, and X.-P. Hu, "A numerical-integration perspective on Gaussian filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 2910–2921, 2006.
- [4] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1254–1269, 2009.
- [5] A. H. Stroud, "Numerical integration formulas of degree two," *Mathematics and Computers in Simulation*, vol. 14, pp. 21–26, 1960.
- [6] A. H. Stroud, "Remarks on the disposition of points in numerical integration formulas," *Mathematical Tables and Other Aids to Computation*, vol. 11, pp. 257–261, 1957.
- [7] D. Xiu, "Numerical integration formulas of degree two," *Applied Numerical Mathematics*, vol. 58, no. 10, pp. 1515–1520, 2008.
- [8] D. H. Titterton and J. L. Weston, *Strapdown Inertial Navigation Technology*, The Institution of Electrical Engineers, 2nd edition, 2004.
- [9] S. J. Julier, "The scaled unscented transformation," in *Proceedings of the American Control Conference*, vol. 6, pp. 4555–4559, Anchorage, Alaska, USA, May 2002.
- [10] S. Chen, H. Tong, Z. Wang, S. Liu, M. Li, and B. Zhang, "Improved generalized belief propagation for vision processing," *Mathematical Problems in Engineering*, vol. 2011, Article ID 416963, 12 pages, 2011.
- [11] S. Y. Chen, "Kalman filter for robot vision: a survey," *IEEE Transactions on Industrial Electronics*, vol. 59, 2012.

## Research Article

# A Filter Algorithm with Inexact Line Search

**Meiling Liu,<sup>1</sup> Xueqian Li,<sup>2</sup> and Qinmin Wu<sup>3</sup>**

<sup>1</sup> Department of Mathematics, Tongji University, Shanghai 200092, China

<sup>2</sup> Business School, University of Shanghai for Science and Technology, Shanghai 200093, China

<sup>3</sup> School of Management, Fudan University, Shanghai 200433, China

Correspondence should be addressed to Meiling Liu, lium2004@163.com

Received 25 October 2011; Accepted 18 December 2011

Academic Editor: Dongdong Ge

Copyright © 2012 Meiling 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.

A filter algorithm with inexact line search is proposed for solving nonlinear programming problems. The filter is constructed by employing the norm of the gradient of the Lagrangian function to the infeasibility measure. Transition to superlinear local convergence is showed for the proposed filter algorithm without second-order correction. Under mild conditions, the global convergence can also be derived. Numerical experiments show the efficiency of the algorithm.

## 1. Introduction

Fletcher and Leyffer [1] proposed filter methods in 2002 offering an alternative to traditional merit functions in solving nonlinear programming problems (NLPs). The underlying concept is that a trial point is accepted provided that there is a sufficient decrease of the objective function or the constraint violation. Filter methods avoid the difficulty of determining a suitable value of the penalty parameter in the merit function. The promising numerical results in [1] led to a growing interest in filter methods in recent years. Two variants of trust-region filter sequential quadratic programming (SQP) method were proposed by Fletcher et al. [2, 3]. Chin and Fletcher [4] developed filter method to sequential linear programming strategy that takes equality-constrained quadratic programming steps. Ribeiro et al. [5] proposed a general filter algorithm that does not depend on the particular method used for the step of computation. Ulbrich [6] argued superlinear local convergence of a filter SQP method. Ulbrich et al. [7] and Wächter and Biegler [8] applied filter technique to interior method and achieved the global convergence to first-order critical point. Wächter and Biegler [9, 10] proposed a line-search filter method and applied it to different algorithm framework. Gould et al. [11] and Shen et al. [12] developed new multidimensional filter technique.

Su and Pu [13] extended the monotonicity of the filter technique. Nie [14] applied filter method to solve nonlinear complementarity problems. In this paper, the global convergence is analyzed widely. However, it has been noted by Fletcher and Leyffer [1] that the filter approach can suffer from the Maratos effect as that of a penalty function approach. By the Maratos effect, a full step can lead to an increase of both infeasibility measure and objective function in filter components even if arbitrarily close to a regular minimizer. This makes the full step unacceptable for the filter and can prohibit fast local convergence.

In this paper, we propose a filter algorithm with inexact line-search for nonlinear programming problems that ensures superlinear local convergence without second-order correction steps. We use the norm of the gradient of the Lagrangian function in the infeasibility measure in the filter components. Moreover, the new filter algorithm has the same global convergence properties as that of the previous works [2, 3, 9]. In addition, since the sufficient decrease conditions in an SQP framework can usually make the algorithm complex and time-consuming, the presented method is a line-search method without using SQP steps. An inexact line-search criterion is used as the sufficient reduction conditions. In the end, numerical experiences also show the efficiency of the new filter algorithm.

This paper is organized as follows. For the main part of the paper, the presented techniques will be applied to general NLP. In Section 2, we state the algorithm mechanism. The convergent properties are shown in Section 3. The global and superlinear convergence are proved. Furthermore, the Maratos effect is avoided. Finally, Section 4 shows the effectiveness of our method under some numerical experiences.

## 2. Inexact Line-Search Filter Approach

### 2.1. The Algorithm Mechanism

We describe and analyze the line-search filter method for NLP with equality constraints. State it as

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{subject to} \quad & c_i(x) = 0, \quad i \in E, \end{aligned} \tag{2.1}$$

where the objective function  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  and the constraints  $c_i$  are assumed to be continuously differentiable, and  $E = \{i \mid i = 1, 2, \dots, m\}$ .

The corresponding Lagrangian function is

$$L(x, \lambda) = f(x) + \lambda^T c(x), \tag{2.2}$$

where the vector  $\lambda$  corresponds to the Lagrange multiplier. The Karush-Kuhn-Tucker (KKT) conditions for (2.1) are

$$\begin{aligned} \nabla f(x) + \lambda^T \nabla c(x) &= 0, \\ c(x) &= 0. \end{aligned} \tag{2.3}$$

For a given initial estimate  $x_0$ , the line-search algorithm generates a sequence of iterates  $x_k$  by  $x_{k+1} = x_k + \alpha_k d_k$  as the estimates of the solution for (2.1). Here, the search direction  $d_k$  is computed from the linearization at  $x_k$  of the KKT conditions (2.3):

$$\begin{pmatrix} W_k & \nabla c(x_k) \\ \nabla c(x_k) & 0 \end{pmatrix} \begin{pmatrix} d_k \\ \lambda^+ \end{pmatrix} = - \begin{pmatrix} \nabla f(x_k) \\ c(x_k) \end{pmatrix}, \quad (2.4)$$

where the symmetric matrix  $W_k$  denotes the Hessian  $\nabla_{xx}^2 L(x_k, \lambda_k)$  of (2.2) or a positive definite approximation to it.

After a search direction  $d_k$  has been computed, the step size  $\alpha_k \in (0, 1]$  is determined by a backtracking line-search procedure, where a decreasing sequence of step size  $\alpha_k$  is tried until some acceptable criteria are satisfied. Generally, the acceptable criteria are constructed by a condition that if the current trial point  $x_k$  can provide sufficient reduction of a merit function. The filter method proposed by Fletcher and Leyffer [1] offers an alternative to merit functions. In this paper, the filter notion is defined as follows.

*Definition 2.1.* A pair  $(V_k, f_k)$  is said to dominate another pair  $(V_l, f_l)$  if and only if both  $V_k \leq V_l$  and  $f_k \leq f_l$ .

Here, we define  $V(x) := \|c(x)\|_2 + \|\nabla L(x)\|_2$  as the  $l_2$  norm of the infeasibility measure. That is, we modify the infeasibility measure in filter, with this modification, the superlinear convergence is possibly derived. Strictly, it is a stationarity measure. However, we still call it infeasibility measure according to its function. In the rest of paper, the norm is always computed by  $l_2$  norm excepting special noting.

*Definition 2.2.* A filter is a list of pairs  $(V_l, f_l)$  such that no pair dominates any other. A point  $(V_k, f_k)$  is said to be acceptable for inclusion in the filter if it is not dominated by any point in the filter.

When a pair  $(V_k, f_k)$  is said to be acceptable to the filter, we also say the iterate  $x_k$  is acceptable to the filter. In filter notion, a trial point  $x_{k+1}$  is accepted if it improves feasibility or improves the objective function. So, it is noted that filter criteria is less demanding than traditional penalty function. When improving optimality, the norm of the gradient of the Lagrangian function will tend to zero, so it offers a more precise analysis for the objective function.

However, this simple filter concept is not sufficient to guarantee global convergence. Fletcher et al. [3] replace this condition by requiring that the next iterate provides at least as much progress in one of the measure  $V$  or  $f$  that corresponds to a small fraction of the current infeasibility measure. Here, we use the similar technique to our filter. Formally, we say that a trial point can be accepted to the current iterate  $x_k$  or the filter if

$$V(x_k) \leq \beta V(x_l) \quad (2.5a)$$

or

$$f(x_k) \leq f(x_l) - \gamma V(x_k), \quad (2.5b)$$

for some fixed constants  $\beta, \gamma \in (0, 1)$ , and  $(V(x_l), f(x_l))$  are points in current filter. In practical implementation, the constants  $\beta$  close to 1 and  $\gamma$  close to 0. However, the criteria (2.5a) and (2.5b) may make a trial point always provides sufficient reduction of the infeasibility measure alone, and not the objective function. To prevent this, we apply a technique proposed in Wächter and Biegler [10] to a different sufficient reduction criteria. The switching condition is

$$\nabla f_k^T d_k < 0, \quad -\alpha_k [\nabla f_k^T d_k]^{e_1} > \delta [V(x_k)]^{e_2}, \quad (2.6)$$

where  $\delta > 0$ ,  $e_2 > 1$ ,  $e_1 \geq 2e_2$ . If the condition (2.6) holds, we replace the filter condition (2.5b) as an inexact line-search condition, that is, the Armijo type condition

$$f(x_{k+1}) \leq f(x_k) + \eta \alpha_k \nabla f_k^T d_k, \quad (2.7)$$

where  $\eta \in (0, 1/2)$  is a constant. If (2.6) holds but not (2.7), the trial points are still determined by (2.5a) and (2.5b).

If a trial point  $x_k$  can be accepted at a step size by (2.7), we refer to  $x_k$  as an  $f$  type iterate and the corresponding  $\alpha_k$  as an  $f$  step size.

## 2.2. The Algorithm Analysis

By the right part of switching condition (2.6), it ensures that the improvement to the objective function by the Armijo condition (2.7) is sufficiently large compared to the current infeasibility measure  $V(x_k)$ . Thus, if iterate points remote from the feasible region, the decrease of the objective function can be sufficient. By setting  $e_2 > 1$ , the progress predicted by the line model  $-\alpha_k [\nabla f_k^T d_k]^{e_1}$  of  $f$  can be a power of the infeasibility measure  $V(x)$ . The choice of  $e_1 \geq 2e_2$  makes it possible that a full step can be accepted by (2.7) when it closes to a local solution.

In this paper, we denote the filter as a set  $\mathcal{F}_k$  containing all iterates accepted by (2.5a) and (2.5b). During the optimization, if the  $f$  type switching condition (2.6) holds and the Armijo condition (2.7) is satisfied, the trial point is determined by (2.7) not by (2.5a) and (2.5b), and the value of the objective function is strictly decreased. This can prevents cycling of the algorithm (see [10]).

If the linear system (2.4) is incompatible, no search direction  $d_k$  can be found and the algorithm switches to a feasibility restoration phase. In it, we try to decrease the infeasibility measure to find a new iterate  $x_{k+1}$  that satisfies (2.4) and is acceptable to the current filter. Similarly, in case  $\nabla f_k^T d_k < 0$ , the sufficient decrease criteria for the objective function in (2.5b) may not be satisfied. To inspect where no admissible step size  $\alpha$  can be found and the feasibility restoration phase has to be invoked, we can consider reducing  $\alpha$  and define

$$\alpha_k^{\min} = \begin{cases} \min \left\{ 1 - \beta, -\frac{\gamma V(x_k)}{-\nabla f_k^T d_k}, \frac{\delta [V(x_k)]^{e_2}}{[-\nabla f_k^T d_k]^{e_1}} \right\}, & \text{if } \nabla f_k^T d_k < 0, \\ 1 - \beta, & \text{otherwise.} \end{cases} \quad (2.8)$$

If the trial step size  $\alpha < \alpha_k^{\min}$ , the algorithm turns to the feasibility restoration phase.

By  $\alpha_k^{\min}$ , it is ensured that the algorithm does not switch to the feasibility restoration phase as long as (2.6) holds for a step size  $\alpha < \alpha_k$  and that the backtracking line-search procedure is finite. Thus, for a trial point  $x_k$ , the algorithm eventually either delivers a new iterate  $x_{k+1}$  or reverts to the feasibility restoration phase. Once finding a feasible direction, the algorithm still implements the normal algorithm.

Of course, the feasibility restoration phase may not always be possible to find a point satisfying the filter-accepted criteria and the compatible condition. It may converge to a nonzero local minimizer of the infeasibility measure and indicate that the algorithm fails. In this paper, we do not specify the particular procedure for the feasibility restoration phase. Any method for dealing with a nonlinear algebraic system can be used to implement a feasibility restoration phase.

### 2.3. The Algorithm

We are now in the place to state the overall algorithm.

*Algorithm 2.3.*

*Step 1.* Given: starting point  $x_0$ , constants  $V^{\max} = \max\{10^4, 1.2V(x_0)\}$ ,  $\beta, \gamma \in (0, 1)$ ,  $\eta \in (0, 1/2)$ ,  $\delta > 0$ ,  $e_1 \geq 2e_2$ ,  $e_2 > 1$ ,  $\tau \in (0, 1)$ .

*Step 2.* Initialize:  $\mathcal{F}_0 = \{(V^{\max}, f(x_0))\}$ , the iteration counter  $k \leftarrow 0$ .

*Step 3.* For  $k = 0, 1, 2, \dots$ , stop if  $x_k$  satisfies the KKT conditions (2.3).

*Step 4.* Compute the search direction  $d_k$  from (2.4). If the system (2.4) is incompatible, go to the feasibility restoration phase in Step 7.

*Step 5.* Set  $\alpha_0 = 1$ , compute  $\alpha_k^{\min}$ .

- (1) If  $\alpha_k < \alpha_k^{\min}$ , go to Step 7. Otherwise, compute the new trial point  $x_{k+1} = x_k + \alpha_k d_k$ .
- (2) If the conditions (2.6) and (2.7) hold, accept the trial step and go to Step 6, otherwise set  $x_k = x_{k+1}$ , go to Step 5(3).
- (3) In case where no  $\alpha_k$  make (2.7) hold, if  $x_{k+1}$  can be accepted to the filter, augment the filter by  $\mathcal{F}_{k+1} = \mathcal{F}_k \cup \{(V(x_{k+1}), f(x_{k+1}))\}$ , go to Step 6; Otherwise set  $x_k = x_{k+1}$ , go to Step 5(4).
- (4) Compute  $\alpha_{k+1} = \tau \alpha_k$ , go back to Step 5(1).

*Step 6.* Increase the iteration counter  $k \leftarrow k + 1$  and go back to Step 4.

*Step 7.* Feasibility restoration phase: by decreasing the infeasibility of  $V$  to find a new iterate  $x_{k+1}$  such that (2.4) is compatible. And if (2.7) holds at  $x_{k+1}$ , continue with the normal algorithm in Step 6; if (2.5a) and (2.5b) hold at  $x_{k+1}$ , augment the filter by  $\mathcal{F}_{k+1} = \mathcal{F}_k \cup \{(V(x_{k+1}), f(x_{k+1}))\}$ , and then continue with the normal algorithm in Step 6; if the feasibility restoration phase cannot find such a point, stop with insuccess.

*Remark 2.4.* In contrast to SQP method with trust-region technique, the actual step does not necessarily satisfy the linearization of the constraints.

*Remark 2.5.* Practical experience shows that the filter allows a large degree of nonmonotonicity and this can be advantageous to some problems.

*Remark 2.6.* To prevent the situation in which a sequence of points for which are  $f$  type iterative point with  $V_k \rightarrow \infty$  is accepted, we set an upper bound  $V^{\max}$  on the infeasibility measure function  $V$ .

For further specific implementation details of Algorithm 2.3, see Section 4.

### 3. Convergence Analysis

#### 3.1. Global Convergence

In this section, we give a global convergence analysis of Algorithm 2.3. We refer to the global convergence analysis of Wächter and Biegler [10] in some places. First, state the necessary assumptions.

*Assumption A1.* Let all iterates  $x_k$  are in a nonempty closed and bounded set  $S$  of  $\mathbb{R}^n$ .

*Assumption A2.* The functions  $f$  and  $c$  are twice continuously differentiable on an open set containing  $S$ .

*Assumption A3.* The matrix  $W_k$  is positive definite on the null space of the Jacobian  $\nabla c(x_k)$  and uniformly bounded for all  $k$ , and the Lagrange multiply  $\lambda$  is bounded for all  $k$ .

Assumptions A1 and A2 are the standard assumptions. Assumption A3 plays an important role to obtain the convergence result and ensures that the algorithm is implementable.

For stating conveniently, we define a set  $J = \{i \mid x_i \text{ is accepted to the filter}\}$ . In addition, sometimes, it is need to revise  $W_k$  to keep it positive definite by some updating methods such as damped BFGS formula [15] or revised Broyden's method [16].

From Assumptions A1–A3, we can get

$$\|d_k\| \leq M_d, \quad m_W \|d_k\|^2 \leq d_k^T W_k d_k \leq M_W \|d_k\|^2, \quad \|\lambda_k\| \leq M_\lambda, \quad (3.1)$$

where  $M_d$ ,  $M_\lambda$ ,  $M_W$ , and  $m_W$  are constants.

**Lemma 3.1.** Suppose Assumptions A1–A3 hold, if  $\{x_{k_i}\}$  is a subsequence of iterates for which  $\|d_{k_i}\| > \epsilon_1$  with a constant  $\epsilon_1 > 0$  independent of  $k_i$ , then for constant  $\epsilon_2 = (m_W/2)\epsilon_1$ , if  $V(x_{k_i}) \leq (m_W/2M_\lambda)\epsilon_1$ , then

$$\nabla f(x_{k_i})^T d_{k_i} \leq -\epsilon_2. \quad (3.2)$$

*Proof.* By the linear system (2.4) and (3.1),

$$\begin{aligned} \nabla f_{k_i}^T d_{k_i} &= -\lambda_{k_i}^T \nabla c(x_{k_i}) d_{k_i} - d_{k_i}^T W_{k_i} d_{k_i} \\ &= \lambda_{k_i}^T c(x_{k_i}) - d_{k_i}^T W_{k_i} d_{k_i} \\ &\leq M_\lambda V(x_{k_i}) - m_W \|d_{k_i}\|^2 \end{aligned}$$

$$\begin{aligned}
&\leq M_\lambda \frac{m_W}{2M_\lambda} e_1 - m_W e_1 \\
&= -e_2.
\end{aligned} \tag{3.3}$$

□

Lemma 3.1 shows that the search direction is a descent direction for the objective function when the trial points are sufficiently close to feasible region.

**Lemma 3.2.** Suppose Assumptions A1–A3 hold, and that there exists an infinite subsequence  $\{x_{k_i}\}$  of  $\{x_k\}$  such that conditions (2.6) and (2.7) hold. Then

$$\lim_{k \rightarrow \infty} V(x_{k_i}) = 0. \tag{3.4}$$

*Proof.* From Assumptions A1 and A2, we know that  $\nabla f$  is bounded. Hence, it has with (3.1) that there exists a constant  $M_m > 0$  such that

$$|\nabla f_k^T d_k| \leq M_m. \tag{3.5}$$

By (2.6) it has

$$\delta[V(x_{k_i})]^{e_2} < -\alpha_{k_i} [\nabla f_{k_i}^T d_{k_i}]^{e_1} \leq M_m^{e_1} \alpha_{k_i}. \tag{3.6}$$

As  $1 - 1/e_1 > 0$ , we have

$$\left(\frac{\delta}{M_m^{e_1}}\right)^{1-1/e_1} [V(x_{k_i})]^{e_2 - e_2/e_1} < \alpha_{k_i}^{1-1/e_1}. \tag{3.7}$$

Then by (2.7) and (3.7),

$$\begin{aligned}
f(x_{k_i+1}) - f(x_{k_i}) &\leq \eta \alpha_{k_i} \nabla f_{k_i}^T d_{k_i} \\
&< -\eta \delta^{1/e_1} \alpha_{k_i}^{1-1/e_1} [V(x_{k_i})]^{e_2/e_1} \\
&< -\eta \delta^{1/e_1} \left(\frac{\delta}{M_m^{e_1}}\right)^{1-1/e_1} [V(x_{k_i})]^{e_2}.
\end{aligned} \tag{3.8}$$

Hence, for  $c_1 := \eta \delta^{1/e_1} (\delta/M_m^{e_1})^{1-1/e_1}$ , an integer  $K$  and all  $j = 1, 2, \dots$ ,

$$f(x_{K+j}) = f(x_K) + \sum_{k_i=K}^{K+j-1} (f(x_{k_i+1}) - f(x_{k_i})) < f(x_K) - c_1 \sum_{k_i=K}^{K+j-1} [V(x_{k_i})]^{e_2}. \tag{3.9}$$

Since  $f(x_{K+j})$  is bounded below as  $j \rightarrow \infty$ , the series on the right hand side in the last line of (3.8) is bounded, then implies the conclusion. □

**Lemma 3.3.** Let  $\{x_{k_i}\} \subset \{x_k\}$  be an infinite subsequence of iterates so that  $(V(x_{k_i}), f(x_{k_i}))$  is entered into the filter. Then

$$\lim_{i \rightarrow \infty} V(x_{k_i}) = 0. \quad (3.10)$$

*Proof.* Here, we refer to the proof of [2, Lemma 3.3]. If the conclusion is not true, there exists an infinite subsequence  $\{k_j\} \subset \{k_i\} \subset J$  such that

$$V(x_{k_j}) \geq \epsilon, \quad (3.11)$$

for all  $j$  and for some  $\epsilon > 0$ . This means that no other  $(V, f)$  pair can be added to the filter at a later stage within the region

$$[V_{k_j} - (1 - \beta)\epsilon, V_{k_j}] \times [f_{k_j} - \gamma\epsilon, f_{k_j}], \quad (3.12)$$

or with the intersection of this region with

$$S_0 = [0, V^{\max}] \times [f^{\min}, \infty] \quad (3.13)$$

for some constants  $f^{\min} \leq f(x_k)$ . Now, the area of each of these regions is  $(1 - \beta)\gamma\epsilon^2$ . Hence, the set  $S_0 \cup \{(V, f) \mid f \leq M_f\}$  is completely covered by at most a finite number of such regions, for any  $M_f \geq f^{\min}$ . Since the pairs  $(V_{k_j}, f_{k_j})$  keep on being added to the filter,  $f_{k_j}$  tends to infinity when  $i$  tends to infinity. Without loss of generality, assume that  $f_{k_{j+1}} \geq f_{k_j}$  for all  $j$  is sufficiently large. But (2.5a) and (3.11) imply that

$$V_{k_{j+1}} \leq \beta V_{k_j}, \quad (3.14)$$

so  $V_{k_j} \rightarrow 0$ , which contradicts (3.11). Then, this latter assumption is not true and the conclusion follows.  $\square$

Next, we show that if  $\{x_k\}$  is bounded, there exists at least one limit point of the iterative points is a first-order optimal point for (2.1).

**Lemma 3.4.** Suppose Assumptions A1–A3 hold. Let  $\{x_{k_i}\}$  be a subsequence with  $\nabla f_{k_i}^T d_{k_i} < -\epsilon_2$  for a constant  $\epsilon_2 > 0$  independent of  $k_i$ . Then, there exists a constant  $\bar{\alpha} > 0$  so that for all  $k_i$  and  $\alpha_{k_i} < \bar{\alpha}$ ,

$$f(x_{k_i} + \alpha_{k_i} d_{k_i}) - f(x_{k_i}) \leq \eta \alpha_{k_i} \nabla f_{k_i}^T d_{k_i}. \quad (3.15)$$

*Proof.* From Assumptions A1 and A2,  $d^T \nabla^2 f(x)d \leq c_f \|d\|^2$  for some constant  $c_f > 0$ . Thus, it follows from the Taylor Theorem and (3.1) that

$$\begin{aligned} f(x_{k_i} + \alpha_{k_i} d_{k_i}) - f(x_{k_i}) - \alpha_{k_i} \nabla f_{k_i}^T d_{k_i} \\ = \alpha_{k_i}^2 d_{k_i}^T \nabla^2 f(y_1) d_{k_i} \leq c_f \alpha_{k_i}^2 \|d_{k_i}\|^2 \leq \alpha_{k_i} (1 - \eta) \epsilon_2 \\ \leq -(1 - \eta) \alpha_{k_i} \nabla f_{k_i}^T d_{k_i}, \end{aligned} \quad (3.16)$$

if  $\alpha_{k_i} \leq \overline{\alpha_{k_i}} := (1 - \eta) \epsilon_2 / c_f M_d^2$ , where  $y_1$  denotes some point on the line segment from  $x_{k_i}$  to  $x_{k_i} + \alpha_{k_i} d_{k_i}$ . Then the conclusion follows.  $\square$

**Lemma 3.5.** Suppose Assumptions A1–A3 hold, and the filter is augmented only a finite number of times, then

$$\lim_{k \rightarrow \infty} \|d_k\| = 0. \quad (3.17)$$

*Proof.* Since the filter is augmented only a finite number of times, there exists an integer  $K_1$  so that for all iterates  $\{x_k\}_{k>K_1}$  the filter is not augmented. If the claim is not true, there must exist a subsequence  $\{x_{k_i}\}$  and a constant  $\epsilon > 0$  so that  $\|d_{k_i}\| \geq \epsilon$  for all  $i$ . Then by Lemma 3.1, it has  $\nabla f_{k_i}^T d_{k_i} \leq -\epsilon_2$  for all  $k_i \geq K_2$ ,  $K_2$  is some integer and  $K_2 > K_1$ . And from Lemmas 3.2 and 3.4, it has  $V(x_{k_i}) \leq \epsilon$  and

$$f(x_{k_i+1}) - f(x_{k_i}) \leq \eta \alpha_{k_i} \nabla f_{k_i}^T d_{k_i} \leq -\alpha_{k_i} \eta \epsilon_2. \quad (3.18)$$

Since  $f(x_{k_i})$  is bounded below and monotonically decreasing for all  $k \geq K_2$ , one can conclude that  $\lim_{i \rightarrow \infty} \alpha_{k_i} = 0$ . This means that for  $k_i > K_2$  the step size  $\alpha = 1$  has not been accepted. So, we can get a  $\alpha_{k_i} < 1$  such that a trial point  $x_{k_i+1} = x_{k_i} + \alpha_{k_i} d_{k_i}$  satisfies

$$(V(x_{k_i+1}), f(x_{k_i+1})) \notin \mathcal{F}_{k_i} \quad (3.19)$$

or

$$f(x_{k_i+1}) - f(x_{k_i}) > \eta \alpha_{k_i} \nabla f_{k_i}^T d_{k_i}. \quad (3.20)$$

Let  $V^{\min} = \min\{V_k \mid (V_k, f_k) \in \mathcal{F}_k\}$ . From Lagrange's Theorem, it has

$$\begin{aligned} V(x_{k_i+1}) &= V(x_{k_i}) + \alpha_{k_i} \nabla V(y_2)^T d_{k_i} \\ &\leq V(x_{k_i}) + \alpha_{k_i} \|\nabla V(y_2)\| \|d_{k_i}\| \\ &\leq V(x_{k_i}) + c_V \alpha_{k_i} M_d, \end{aligned} \quad (3.21)$$

for some constant  $c_V$ , where  $y_2$  denotes some point on the line segment from  $x_{k_i}$  to  $x_{k_i} + \alpha_{k_i} d_{k_i}$ . Since  $\lim_{i \rightarrow \infty} \alpha_{k_i} = 0$  and  $\lim_{i \rightarrow \infty} V(x_{k_i}) = 0$  by Lemmas 3.2 and 3.3, it has  $V(x_{k_i+1}) < V^{\min}$ .

for  $k_i$  sufficiently large, so (3.19) is not true. In case (3.20), since  $\alpha_{k_i} \rightarrow 0$  for sufficiently large  $k_i$ , we have  $\alpha_{k_i} \leq \bar{\alpha}$  with  $\bar{\alpha}$  from Lemma 3.4, that is, (3.20) can not be satisfied. Then the conclusion follows.  $\square$

**Lemma 3.6.** Suppose Assumptions A1–A3 hold. Let  $\{x_{k_i}\}$  be a subsequence of  $\{x_k\}$  with  $\nabla f_{k_i}^T d_{k_i} \leq -\epsilon_2$  for a constant  $\epsilon_2 > 0$  independent of  $k_i$ . Then, there exists trial points can be accepted to the filter.

*Proof.* The mechanisms of Algorithm 2.3 ensure that the first iterate can be accepted to the filter. Next, we can assume that  $(V(x_k), f(x_k))$  is acceptable to the  $k$ th filter and  $(V(x_l), f(x_l)) \in \mathcal{F}_k, l < k$ . If  $\alpha_{k_i} \leq c_2 := \epsilon_2/M_d^2 c_f$ , it has

$$\alpha_{k_i}^2 \leq \frac{\alpha_{k_i} \epsilon_2}{M_d^2 c_f} \leq \frac{-\alpha_{k_i} \nabla f_{k_i}^T d_{k_i}}{c_f \|d_{k_i}\|^2}, \quad (3.22)$$

that is,  $\alpha_{k_i} \nabla f_{k_i}^T d_{k_i} + c_f \alpha_{k_i}^2 \|d_{k_i}\|^2 \leq 0$ , so by (3.16)

$$f(x_{k_i} + \alpha_{k_i} d_{k_i}) \leq f(x_{k_i}). \quad (3.23)$$

Similarly, if  $\alpha_{k_i} \leq c_3 V(x_{k_i}) \leq V(x_{k_i})/\|d\|^2 c_V$ , with  $c_3 := 1/M_d^2 c_V$  and  $c_V$  from Lemma 3.5, it has

$$\begin{aligned} -\alpha_{k_i} V(x_{k_i}) + c_V \alpha_{k_i}^2 \|d_{k_i}\|^2 &\leq 0, \\ V(x_{k_i} + \alpha_{k_i} d_{k_i}) &\leq V(x_{k_i}). \end{aligned} \quad (3.24)$$

Hence, we have

$$\begin{aligned} f(x_{k_i+1}) &= f(x_{k_i} + \alpha_{k_i} d_{k_i}) \leq f(x_{k_i}) \leq f(x_l) - \gamma V(x_l), \\ V(x_{k_i+1}) &= V(x_{k_i} + \alpha_{k_i} d_{k_i}) \leq V(x_{k_i}) \leq \beta V(x_l). \end{aligned} \quad (3.25)$$

The claim then follows from (3.25).  $\square$

The last Lemma 3.6 shows, for case  $V(x_k) > 0$ , Algorithm 2.3 either accepts a new iterate to the filter or switches to the feasibility restoration phase. For case  $V(x_k) = 0$  and the algorithm does not stop at a KKT point, then  $\nabla f_k^T d_k < 0$ ,  $\alpha_k^{\min} = 0$ , and the Armijo condition (2.7) is satisfied for sufficiently small step size  $\alpha_k$ , so an  $f$  type iterate is accepted. Hence, the inner loop in Step 5 always terminates in a finite number of trial steps, and Algorithm 2.3 is well defined.

**Lemma 3.7.** Suppose Assumptions A1–A3 hold. Let  $\{x_{k_i}\}$  be a subsequence with  $\|d_{k_i}\| \geq \epsilon$  for a constant  $\epsilon > 0$  independent of  $k_i$ . Then, there exists an sufficient large integer  $K$  such that for all  $k_i > K$  the algorithm can generate some trial points either be accepted to the filter or be  $f$  type steps.

*Proof.* By Lemmas 3.1, 3.2, and 3.3, there exist constants  $\epsilon_1, \epsilon_2 > 0$  so that

$$V(x_{k_i}) \leq \epsilon_1, \quad \nabla f_{k_i}^T d_{k_i} < -\epsilon_2 \quad (3.26)$$

for all  $k_i > K$ .

If  $V(x_{k_i}) = 0$ , the  $f$  type switching condition (2.6) is true, there must exist iterates for which are  $f$  type iterates. For the remaining iterates with  $V(x_{k_i}) > 0$ , if

$$V(x_{k_i}) < \min \left\{ \frac{\bar{\alpha}}{c_3}, \frac{c_2}{c_3}, \left[ \frac{\tau c_3 \epsilon_2^{e_1}}{\delta} \right]^{1/e_2-1} \right\}, \quad (3.27)$$

with  $\bar{\alpha}$  from Lemma 3.4 and  $c_2, c_3$  from Lemma 3.6, it implies with  $e_2 > 1$

$$\frac{\delta [V(x_{k_i})]^{e_2}}{\epsilon_2^{e_1}} < \tau c_3 V(x_{k_i}), \quad (3.28)$$

as well as

$$c_3 V(x_{k_i}) < \min \{ \bar{\alpha}, c_2 \}. \quad (3.29)$$

Now choose an arbitrary  $k_i \geq K$  with  $V(x_{k_i}) > 0$  and define

$$c_4 := c_3 V(x_{k_i}) = \min \{ \bar{\alpha}, c_2, c_3 V(x_{k_i}) \}. \quad (3.30)$$

Lemmas 3.4 and 3.6 then imply that a trial step size  $\alpha_{k_i} \leq c_4$  satisfies both

$$\begin{aligned} f(x_{k_i} + \alpha_{k_i} d_{k_i}) &\leq f(x_{k_i}) + \eta \alpha_{k_i} \nabla f_{k_i}^T d_{k_i}, \\ (V(x_{k_i} + \alpha_{k_i} d_{k_i}), f(x_{k_i} + \alpha_{k_i} d_{k_i})) &\in \mathcal{F}_{k_i}. \end{aligned} \quad (3.31)$$

Since  $\alpha > \tau \alpha_k > \tau \alpha_k^{\min}$  by the definition of  $\alpha_k^{\min}$ , the method does not switch to the feasibility restoration phase for those trial step sizes. Then the claim follows.  $\square$

Based on the above lemmas, we can give the main global convergence result.

**Theorem 3.8.** *Suppose Assumptions A1–A3 hold, then*

$$\lim_{k \rightarrow \infty} V(x_k) = 0, \quad (3.32)$$

$$\lim_{k \rightarrow \infty} \|d_k\| = 0, \quad (3.33)$$

that is, there exists a limit point  $\bar{x}$  of  $\{x_k\}$  which is a first-order optimal point for (2.1).

*Proof.* (3.32) follows from Lemmas 3.2 and 3.3.

If the filter is augmented a finite number of times, then the claim (3.33) holds from Lemma 3.5. For either case, there exists a subsequence  $\{x_{k_i}\}$  so that  $k_i \in J$  for all  $i$ . Suppose the conclusion (3.33) is not true, there must exist a subsequence  $\{x_{k_j}\}$  of  $\{x_{k_i}\}$  such that  $\|d_{k_j}\| \geq \epsilon$  for some constant  $\epsilon > 0$ . Hence by Lemmas 3.1 and 3.3, it has  $\nabla f_{k_j}^T d_{k_j} < -\epsilon_2$  and  $\lim_{i \rightarrow \infty} V(x_{k_j}) = 0$  for all  $k_j$ . Then by Lemma 3.7, when  $\alpha < \min \{ \bar{\alpha}, c_2, c_3 V(x_{k_j}) \}$ , the algorithm

can generate a  $f$  type iterate, that is, the filter is not augmented, this contradicts the choice of  $\{x_{k_j}\}$ , so that (3.33) holds.  $\square$

### 3.2. Local Convergence

In this section, we show the local convergence of Algorithm 2.3.

*Assumption A4.* The iterates  $x_k$  converge to a point  $\bar{x}$  that satisfies

$$\begin{aligned} V(\bar{x}) = 0, \quad \nabla_x L(\bar{x}, \bar{\lambda}) = 0, \quad \nabla_{xx} L(\bar{x}, \bar{\lambda}) \text{ is positive definite on } \{d : \nabla c(\bar{x})^T d = 0\}, \\ \nabla c(\bar{x}) \text{ has full-row rank.} \end{aligned} \tag{3.34}$$

*Assumption A5.* There is a neighborhood  $N(\bar{x})$  of  $\bar{x}$  such that  $W_k = \nabla_{xx} L(x_k, \lambda_k)$ , for all  $x_k \in N(\bar{x})$ .

*Remark 3.9.* Under Assumption A4, the point  $\bar{x}$  is a strict local minimum of (2.1).

*Remark 3.10.* Under Assumptions A4 and A5, it is well known that with the choice  $x_{k+1} = x_k + d_k$ , the sequence  $\{x_k\}$  converges  $q$ -superlinearly to  $\bar{x}$  and that the convergence is  $q$ -quadratic if  $\nabla_{xx} f$  and  $\nabla_{xx} c_i$  are lipschitz continuous in a neighborhood of  $\bar{x}$ . That is, for any given  $\zeta \in [0, 1]$ ,  $x_j \in N(\bar{x})$ ,  $j = k, k+1, \dots$ , and  $x_{j+1} = x_j + d_j$ , it has

$$\|d_{j+1}\| \leq \zeta \|d_j\|, \quad \|x_{j+1} - \bar{x}\| \leq \zeta \|x_j - \bar{x}\|. \tag{3.35}$$

We use the proof techniques of local convergence in [6]. In proof, define  $l_\rho(x, \lambda) = L(x, \lambda) + (\rho/2)\|c(x)\|_2^2$  and  $\hat{l}_\rho(x, \lambda) = f(x) + (\rho/2)V(x)$  with  $\rho$  is a parameter.

**Lemma 3.11.** Suppose Assumptions A1–A3 hold. Let  $\bar{x}$  satisfy the Assumption A4. Then, there exist constants  $0 < r < t$ ,  $\rho_0 > 0$  and a neighborhood  $N_\sigma(\bar{x}) = \{x : \|x - \bar{x}\| < \sigma\} \subset N(\bar{x})$  such that

$$\frac{r}{2}\|x - \bar{x}\|^2 + \frac{\rho - \rho_0}{2}\|c(x)\|^2 \leq l_\rho(x, \lambda) - l_\rho(\bar{x}, \bar{\lambda}) \leq t(1 + \rho)\|x - \bar{x}\|^2, \tag{3.36}$$

for all  $\rho \geq \rho_0$  and all  $x \in N_\sigma(\bar{x})$ .

Moreover, possibly after increasing  $t > 0$ ,  $\rho_0 > 0$  and reducing  $\sigma > 0$ , we have that for all  $\rho \geq \rho_0$  and all  $x \in N_\sigma(\bar{x})$

$$\frac{r}{2}\|x - \bar{x}\|^2 + \frac{\rho - \rho_0}{2}V(x) \leq \hat{l}_\rho(x, \lambda) - \hat{l}_\rho(\bar{x}, \bar{\lambda}) \leq t(1 + \rho)\|x - \bar{x}\|^2. \tag{3.37}$$

*Proof.* Let  $x \in N(\bar{x})$ . Using Taylor's Theorem and  $\nabla_x l_\rho(\bar{x}, \bar{\lambda}) = 0, \nabla_\lambda l_\rho(\bar{x}, \bar{\lambda}) = 0$ , we have with some  $(x', \lambda')$  on the line segment between  $(\bar{x}, \bar{\lambda})$  and  $(x, \lambda)$

$$\begin{aligned} l_\rho(x, \lambda) - l_\rho(\bar{x}, \bar{\lambda}) &= \frac{1}{2}(x - \bar{x})^T \nabla_{xx} l_\rho(x', \lambda')(x - \bar{x}) \\ &\quad + (x - \bar{x})^T \nabla_{x\lambda} l_\rho(x', \lambda')(\lambda - \bar{\lambda}). \end{aligned} \quad (3.38)$$

Obviously, it has

$$\nabla_{xx} l_\rho(x', \lambda') = \nabla_{xx} l_{\rho/2}(x', \lambda' + \frac{\rho}{2}c(x')) + \frac{\rho}{2}\nabla c(x')^T \nabla c(x'). \quad (3.39)$$

Under Assumption A4, there exists  $\bar{\rho} > 0$  such that for all  $\rho \geq \bar{\rho}$ ,

$$d^T \nabla_{xx} l_{\rho/2}(\bar{x}, \bar{\lambda}) d \geq 4r\|d\|^2, \quad \forall d \in \mathbb{R}^n, \quad (3.40)$$

with a constant  $r > 0$ , see [15, Theorem 17.5].

Suppose  $\lambda(x)$  is Lipschitz continuous and  $L_y$  is the Lipschitz constant of  $\lambda$ , and  $\bar{\rho} > 0$  is a constant. Let  $\rho_0 := \max\{\bar{\rho}, 4L_y^2/r\}$  for all  $x$  with  $\|x - \bar{x}\| \leq \sigma$ , it has by continuity

$$d^T \nabla_{xx} l_{\rho_0/2}(x, \lambda + \frac{\rho_0}{2}c(x)) d \geq 2r\|d\|^2, \quad \forall d \in \mathbb{R}^n. \quad (3.41)$$

Thus, we obtain for all  $x \in N_\sigma(\bar{x})$  by (3.38), (3.39), and (3.41)

$$l_{\rho_0}(x, \lambda) - l_{\rho_0}(\bar{x}, \bar{\lambda}) \geq r\|x - \bar{x}\|^2 + \frac{\rho_0}{4}\|\nabla c(x')(x - \bar{x})\|^2 + (\lambda - \bar{\lambda})^T \nabla c(x')(x - \bar{x}). \quad (3.42)$$

It is obvious for all  $s > 0$  that

$$\begin{aligned} 2(\lambda - \bar{\lambda})^T \nabla c(x')(x - \bar{x}) &\geq -\frac{1}{s}\|\nabla c(x')(x - \bar{x})\|^2 - s\|\lambda - \bar{\lambda}\|^2 \\ &\geq -\frac{1}{s}\|\nabla c(x')(x - \bar{x})\|^2 - sL_y^2\|\lambda - \bar{\lambda}\|^2. \end{aligned} \quad (3.43)$$

If  $s = r/L_y^2$ ,  $1/s = L_y^2/r \leq \rho^0/4$ , then it has

$$l_{\rho_0}(x, \lambda) - l_{\rho_0}(\bar{x}, \bar{\lambda}) \geq \frac{r}{2}\|x - \bar{x}\|^2 + \frac{\rho_0}{8}\|\nabla c(x')(x - \bar{x})\|^2. \quad (3.44)$$

Since  $c(\bar{x}) = 0, l_\rho(\bar{x}, \bar{\lambda}) = l_{\rho_0}(\bar{x}, \bar{\lambda}) = f(\bar{x})$ , and by (3.44) it has

$$\begin{aligned} l_\rho(x, \lambda) - l_\rho(\bar{x}, \bar{\lambda}) &= l_{\rho_0}(x, \lambda) - l_{\rho_0}(\bar{x}, \bar{\lambda}) + \frac{\rho - \rho_0}{2}\|c(x)\|^2 \\ &\geq \frac{r}{2}\|x - \bar{x}\|^2 + \frac{\rho - \rho_0}{2}\|c(x)\|^2, \end{aligned} \quad (3.45)$$

that is the left inequality in (3.36). For the right inequality in (3.36), it is obvious from (3.38) that for all  $x \in N(\bar{x})$ ,

$$l_\rho(x, \lambda) - l_\rho(\bar{x}, \bar{\lambda}) \leq t(1 + \rho)\|x - \bar{x}\|^2, \quad (3.46)$$

with a constant  $t > 0$ . This proves the inequality (3.36).

Choose  $K$  large enough such that for  $j \geq k \geq K$ ,  $c(x_j) \rightarrow 0$ . We can assume  $\|c(x)\| < 1$ . Then, if  $\rho_0$  is large enough, it has from (3.44) and  $\hat{l}_\rho(\bar{x}, \bar{\lambda}) = l_{\rho_0}(\bar{x}, \bar{\lambda}) = f(\bar{x})$  that

$$\hat{l}_{\rho_0}(x, \lambda) - \hat{l}_{\rho_0}(\bar{x}, \bar{\lambda}) \geq l_{\rho_0}(x, \lambda) - l_{\rho_0}(\bar{x}, \bar{\lambda}) \geq \frac{r}{2}\|x - \bar{x}\|^2. \quad (3.47)$$

By an analogue of (3.45) holds for  $\hat{l}_\rho$ , this proves the left inequality in (3.37).

On the other hand, it has

$$\hat{l}_\rho(x, \lambda) - \hat{l}_\rho(\bar{x}, \bar{\lambda}) = f(x) + \frac{\rho}{2}V(x) - f(\bar{x}). \quad (3.48)$$

Since  $f(x)$  and  $c(x)$  are twice continuously differentiable on closed set  $S$ , we have  $f(x) - f(\bar{x}) = O(\|x - \bar{x}\|^2)$  and  $V(x) = O(\|x - \bar{x}\|^2)$ . This shows the right inequality in (3.37) possibly after increasing  $t$ .  $\square$

**Lemma 3.12.** *Let  $\bar{x}$  satisfy Assumptions A4 and A5. Then for any  $\zeta \in [0, 1]$  and  $M \geq 1$ , there is an index  $K$  such that for all  $k \geq K$ , with*

$$\|x_k - \bar{x}\| \leq M \min_{(V_l, f_l) \in \mathcal{F}_k} \|x_l - \bar{x}\|, \quad (3.49)$$

the points  $x_{j+1}$ ,  $j = k, k+1, \dots$ , with  $\alpha_k = 1$  are acceptable to

$$\mathcal{F}_j := \mathcal{F}_k \cup (V_k, f_k) \cup (V_{k+1}, f_{k+1}) \cup \dots \cup (V_j, f_j). \quad (3.50)$$

*Proof.* Let  $N(\bar{x})$  as in Assumption A5,  $\zeta = 1/2$ ,  $\rho_0$  and  $N_\sigma(\bar{x}) \subset N(\bar{x})$  be given by Lemma 3.11. For all  $k \geq K_1$ ,  $K_1$  is a sufficient large integer, and choose  $\rho \geq \rho_0$  so large that

$$\beta\left(\frac{\rho}{2} - \gamma\right) \geq \beta\frac{\rho}{4}, \quad \frac{\rho - \rho_0}{2} \geq \left(1 - \frac{\beta}{4}\right)\frac{\rho}{2}. \quad (3.51)$$

Then, it has from (3.37) in Lemma 3.11

$$\frac{r}{2}\|x - \bar{x}\|^2 + \left(1 - \frac{\beta}{4}\right)\frac{\rho}{2}V(x) = \hat{l}_\rho(x, \lambda) - \hat{l}_\rho(\bar{x}, \bar{\lambda}) \leq t(1 + \rho)\|x - \bar{x}\|^2. \quad (3.52)$$

Let

$$V' := \min_{(V_j, f_j) \in \mathcal{F}_{K_1}} V_j. \quad (3.53)$$

By  $V' > 0$  and continuity there exists  $0 < \sigma_1 < \sigma$  such that  $V(x) \leq \beta V'$  for all  $x \in N_{\sigma_1}(\bar{x})$ , so the point  $x$  is acceptable to  $\mathcal{F}_{K_1}$ . Since  $x_k \rightarrow \bar{x}$ ,  $x_k \in N_{\sigma_1}(\bar{x})$  for all  $k \geq K_2 > K_1$ ,  $K_2$  is an integer. By (3.35), we can choose  $\sigma_1$  so small that for all  $k \geq K_2$ , the sequence  $\{x_j\}_{j \geq k}$  with  $\alpha_k = 1$  converges linearly with a contraction factor of at least

$$\|x_{j+1} - \bar{x}\| \leq \sqrt{\frac{1}{2} \left(1 - \frac{1-\beta/2}{1-\beta/4}\right) \frac{r}{2M^2t(1+\rho)}} \|x_j - \bar{x}\| \leq \frac{1}{2} \|x_j - \bar{x}\|. \quad (3.54)$$

Suppose an arbitrary  $k_2 \geq K_2$  such that (3.49) holds, and set  $\sigma_2 := \|x_k - \bar{x}\|$ . By (3.54), it has  $x_{k+1} \in N_{\sigma_2}(\bar{x})$  and  $x_{k+1}$  is acceptable to  $\mathcal{F}_{K_1}$ . Next, it is a need to show that  $x_k$  is acceptable with respect to  $(V_l, f_l) \in \mathcal{F}_k \cup (V_k, f_k)$  for  $K_1 \leq l \leq k$ . By (3.49), it has  $x_l \in N_\sigma(\bar{x}) \setminus N_{\sigma_2/M}(\bar{x})$ , so by (3.52)

$$\hat{l}_\rho(x_l, \lambda_l) - \hat{l}_\rho(\bar{x}, \bar{\lambda}) \geq \frac{r}{2M^2} \sigma_2^2 + \left(1 - \frac{\beta}{4}\right) \frac{\rho}{2} V_l. \quad (3.55)$$

And, by (3.52) and (3.54)

$$\hat{l}_\rho(x_{k+1}, \lambda_{k+1}) - \hat{l}_\rho(\bar{x}, \bar{\lambda}) \leq t(1+\rho) \|x_{k+1} - \bar{x}\|^2 \leq \frac{1}{2} \left(1 - \frac{1-\beta/2}{1-\beta/4}\right) \frac{r}{2M^2} \sigma_2^2. \quad (3.56)$$

Next, suppose  $(x, \lambda)$  with  $x \in N_{\sigma_2}(\bar{x})$  is not acceptable to  $(V_l, f_l)$  then

$$V(x) > \beta V_l, \quad f(x) + \gamma V(x) > f_l. \quad (3.57)$$

Thus, it has with (3.51) and (3.52)

$$\begin{aligned} \hat{l}_\rho(x, \lambda) &= f(x) + \frac{\rho}{2} V(x) > f_l + \left(\frac{\rho}{2} - \gamma\right) V(x) \\ &> f_l + \beta \left(\frac{\rho}{2} - \gamma\right) V_l \geq f_l + \beta \left(\frac{\rho}{4}\right) V_l \\ &= \hat{l}_\rho(x_l, \lambda_l) - \left(1 - \frac{\beta}{2}\right) \frac{\rho}{2} V_l \\ &\geq \hat{l}_\rho(x_l, \lambda_l) - \frac{1-\beta/2}{1-\beta/4} (\hat{l}_\rho(x_l, \lambda_l) - \hat{l}_\rho(\bar{x}, \bar{\lambda})). \end{aligned} \quad (3.58)$$

This shows with (3.55) that

$$\begin{aligned} \hat{l}_\rho(x, \lambda) - \hat{l}_\rho(\bar{x}, \bar{\lambda}) &> \left(1 - \frac{1-\beta/2}{1-\beta/4}\right) (\hat{l}_\rho(x_l, \lambda_l) - \hat{l}_\rho(\bar{x}, \bar{\lambda})) \\ &\geq \left(1 - \frac{1-\beta/2}{1-\beta/4}\right) \frac{r}{2M^2} \sigma_2^2. \end{aligned} \quad (3.59)$$

This produces a contradiction to (3.56), so  $x_{k+1}$  is acceptable to  $\mathcal{F}_k \cup (V_k, f_k)$ . Then, the acceptability of  $(V_{j+1}, f_{j+1})$  for  $\mathcal{F}_l$  follows by induction.  $\square$

Next, we show that the sequence  $\{x_j\}_{j \geq k}$  with  $\alpha_k = 1$  can make the sufficient decreasing condition (2.7) hold.

**Lemma 3.13.** *Suppose Assumptions A1–A3 hold. Let  $\bar{x}$  satisfy Assumptions A4 and A5 and let  $K$  be as in Lemma 3.11. Then for all  $k > K$  the sequence  $\{x_j\}_{j \geq k}$  with  $\alpha_j = 1$  satisfies*

$$f(x_{j+1}) \leq f(x_j) + \eta \alpha_j \nabla f_j^T d_j. \quad (3.60)$$

*Proof.* Suppose  $\alpha_j \nabla f_j^T d_j < 0$  and  $\alpha_j [\nabla f_j^T d_j]^{e_1} < -\delta [V(x_j)]^{e_2}$  hold. By  $\alpha_j = 1$ , thus

$$\nabla f_j^T d_j < -(\delta)^{1/e_1} [V(x_j)]^{e_2/e_1}. \quad (3.61)$$

On the other hand, with  $\alpha_j = 1$  the assertion  $f(x_{j+1}) \leq f(x_j) + \eta \nabla f_j^T d_j$  yields

$$\eta \nabla f_j^T d_j \geq \nabla f_j^T d_j + \frac{1}{2} d_j^T \nabla_{xx}^2 f(x') d_j. \quad (3.62)$$

Thus,

$$\nabla f_j^T d_j \leq -\frac{1}{2(1-\eta)} d_j^T \nabla_{xx}^2 f(x') d_j, \quad (3.63)$$

where  $x'$  on the line segment between  $x_j$  and  $x_{j+1}$ .

Obviously, we can prove the conclusion if with  $K > 0$  large enough and for all  $j \geq k \geq K$  the following holds

$$-(\delta)^{1/e_1} [V(x_j)]^{e_2/e_1} \leq -\frac{1}{2(1-\eta)} d_j^T \nabla_{xx}^2 f(x') d_j. \quad (3.64)$$

Since  $c_j + \nabla c_j^T d_j = 0$ ,  $d_j = -\nabla c_j^T (\nabla c_j \nabla c_j^T)^{-1} c_j$ . By Assumption A4,  $\nabla c_j$  has full-row rank, there exists  $c_d > 0$  such that

$$\|d_j\| \leq c_d V(x_j), \quad (3.65)$$

thus,

$$\|d_j\|^2 \leq c_d^2 [V(x_j)]^2. \quad (3.66)$$

Choose  $K$  large enough such that  $V(x_j) \leq 1$  for all  $j \geq k \geq K$ . By  $e_1 > 2e_2$  and (3.66) it has

$$\|d_j\|^2 \leq c_d^2 c_5 [V(x_j)]^{e_2/e_1}, \quad (3.67)$$

**Table 1:** Description on headers.

Header	Description
Problem	The name of the CUTE problem being solved
$n$	Number of variables of the problem
$m$	The total number of constraints
$m_{\text{nl}}$	Number of nonlinear constraints
NIT1	Number of iterations of Algorithm 2.3
NIT2	Number of iterations of algorithm Tri-filter
NIT3	Number of iterations of algorithm SNOPT
NF1	Number of $f$ evaluations of Algorithm 2.3
NF2	Number of $f$ evaluations of algorithm Tri-filter
NF3	Number of $f$ evaluations of algorithm SNOPT

for a constant  $c_5 := 2\delta^{1/e_1}(1 - \eta)/c_f c_d^2$ , we can choose suitable parameters such that the last inequality holds. Thus,

$$\delta^{1/e_1} [V(x_j)]^{e_2/e_1} \geq \frac{1}{2(1 - \eta)} c_f \|d_j\|^2 \geq \frac{1}{2(1 - \eta)} d_j^T \nabla_{xx}^2 f(x') d_j. \quad (3.68)$$

This completes the proof.  $\square$

**Theorem 3.14.** Suppose Assumptions A1–A5 hold. Then, there exists  $K > 0$  such that Algorithm 2.3 takes steps with  $\alpha_k = 1$  for all  $k \geq K$ , that is,

$$x_{k+1} = x_k + d_k. \quad (3.69)$$

In particular,  $x_k$  converges  $q$ -superlinearly to  $\bar{x}$ . If  $\nabla_{xx} f$  and  $\nabla_{xx} c_i$  are Lipschitz continuous in a neighborhood of  $\bar{x}$  then  $x_k$  converges  $q$ -quadratically.

*Proof.* Since Assumptions A4 and A5 hold,  $x_k \rightarrow \bar{x}$  with  $\bar{x}$  satisfying A4. By Lemmas 3.12 and 3.13, the iterate  $(x_{k+1}, \lambda_{k+1}) = (x_k + d_k, \lambda(x_k + d_k))$  is acceptable to the filter  $\mathcal{F}_k \cup (V_k, f_k)$  and satisfies the sufficient decreasing condition (2.7). Thus, the trial iterate  $(x_k + d_k, \lambda(x_k + d_k))$  is accepted by the algorithm and it has

$$x_{k+1} = x_k + d_k, \quad \lambda_{k+1} = \lambda(x_k + d_k). \quad (3.70)$$

That is in both cases the algorithm takes the steps with  $\alpha_k = 1$ . And according to Remark 3.10,  $\{x_k\}$  converges  $q$ -superlinearly to  $\bar{x}$ .  $\square$

## 4. Numerical Experience

In this section, we give some numerical results of Algorithm 2.3. We take some CUTE problems [17], which are available freely on NEOS, to test our algorithm. The test codes are edited in MATLAB. The details about the implementation are described as follows.

**Table 2:** Numerical results of small-scale problems.

Problem	<i>n</i>	<i>m</i>	<i>m<sub>nl</sub></i>	NIT1/NIT2	NF1/NF2
hs017	2	5	3	11/14	14/20
hs019	2	6	4	2/5	5/6
hs024	2	4	2	1/4	2/5
hs037	3	7	6	5/7	6/10
hs042	3	5	4	6/7	8/10
hs043	4	3	0	11/14	12/16
hs046	5	4	2	19/30	41/102
hs047	5	6	3	9/19	28/70
hs049	5	4	2	17/23	27/26
hs056	7	15	11	8/18	16/41
hs059	2	7	4	13/16	25/20
hs063	3	7	5	13/15	30/44
hs071	4	11	9	6/28	12/103
hs076	4	7	4	5/6	6/7
hs077	5	4	2	15/26	25/36
hs078	5	6	3	8/6	18/29
hs079	5	6	3	6/9	10/11
hs092	6	1	1	25/18	35/52
hs098	6	16	12	6/6	6/7
hs099	19	42	28	15/19	25/78
hs104	8	22	16	16/23	33/60
hs106	8	22	21	32/36	12/37
hs108	9	14	10	15/12	13/13
hs111	10	26	23	42/48	50/52
hs112	10	16	16	25/52	25/30
hs113	10	8	1	9/13	15/19
hs114	10	34	23	25/28	23/29
hs116	13	41	35	37/44	30/54
hs117	15	20	15	15/16	15/23
hs118	15	59	59	12/18	22/19
hs119	16	48	40	13/18	16/19

**Table 3:** Numerical results of mid-scale problems.

Problem	<i>n</i>	<i>m</i>	<i>m<sub>nl</sub></i>	NIT1/NIT3	NF1/NF3
Ngone	97	1371	1369	16/72	21/20
Grouping	100	450	350	1/0	4/2
Eigen a2	110	110	55	8/11	5/6
Eigen a	110	110	110	52/226	158/208

- (a) The parameters are set to  $\beta = 0.99$ ,  $\gamma = 0.0001$ ,  $\eta = 0.4$ ,  $\delta = 1$ ,  $\tau = 0.6$ ,  $e_1 = 2.3$ ,  $e_2 = 1.1$ , the termination tolerance  $\epsilon = 1E - 6$ .
- (b) The optimal residual is defined as

$$\text{res} = \max\{\|\nabla f(x_k) + \nabla c(x_k)\lambda\|, V(x_k)\}. \quad (4.1)$$

That is, the algorithm terminates when  $\text{res} < \epsilon$ .

- (c)  $W_k$  is updated by damped BFGS formula [15].

The detailed results of the numerical test on small-scale problems are summarized in Table 2. For comparison purposes, we also give the numerical results of tridimensional line-search filter solver (Tri-filter) in Shen et al. [12] in Table 2. The row headers in Tables 2 and 3 are presented in Table 1.

The results in Table 2 indicate that Algorithm 2.3 has a good effect.

In addition, we also test some mid-scale problems. And we compare the numerical results which are summarized in Table 3 in Algorithm 2.3 and SNOPT solver in Gill et al. [18] since no mid-scale problems are given in trifilter solver.

From Table 3, we find the efficiency of Algorithm 2.3 is also improved significantly. From both Tables 2 and 3, in general, the behavior of the proposed algorithm is rather stable. Finally, we may conclude that, as far as our limited computational experience is concerned, the proposed algorithm is well comparable to trifilter solver and SNOPT solver.

## Acknowledgment

This research is partially supported by the National Natural Science Foundation of China (Grant nos. 10771162 and 91024021).

## References

- [1] R. Fletcher and S. Leyffer, "Nonlinear programming without a penalty function," *Mathematical Programming*, vol. 91, no. 2, pp. 239–269, 2002.
- [2] R. Fletcher, N. I. M. Gould, S. Leyffer, P. L. Toint, and A. Wächter, "Global convergence of a trust-region SQP-filter algorithm for general nonlinear programming," *SIAM Journal on Optimization*, vol. 13, no. 3, pp. 635–659, 2002.
- [3] R. Fletcher, S. Leyffer, and P. L. Toint, "On the global convergence of a filter-SQP algorithm," *SIAM Journal on Optimization*, vol. 13, no. 1, pp. 44–59, 2002.
- [4] C. M. Chin and R. Fletcher, "On the global convergence of an SLP-filter algorithm that takes EQP steps," *Mathematical Programming*, vol. 96, no. 1, Ser. A, pp. 161–177, 2003.
- [5] A. A. Ribeiro, E. W. Karas, and C. C. Gonzaga, "Global convergence of filter methods for nonlinear programming," *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1231–1249, 2008.
- [6] S. Ulbrich, "On the superlinear local convergence of a filter-SQP method," *Mathematical Programming*, vol. 100, no. 1, pp. 217–245, 2004.
- [7] M. Ulbrich, S. Ulbrich, and L. N. Vicente, "A globally convergent primal-dual interior-point filter method for nonlinear programming," *Mathematical Programming*, vol. 100, no. 2, pp. 379–410, 2003.
- [8] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.
- [9] A. Wächter and L. T. Biegler, "Line search filter methods for nonlinear programming: motivation and global convergence," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 1–31, 2005.
- [10] A. Wächter and L. T. Biegler, "Line search filter methods for nonlinear programming: local convergence," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 32–48, 2005.

- [11] N. I. M. Gould, C. Sainvitu, and P. L. Toint, "A filter-trust-region method for unconstrained optimization," *SIAM Journal on Optimization*, vol. 16, no. 2, pp. 341–357, 2006.
- [12] C. G. Shen, W. J. Xue, and D. G. Pu, "Global convergence of a tri-dimensional filter SQP algorithm based on the line search method," *Applied Numerical Mathematics*, vol. 59, no. 2, pp. 235–250, 2009.
- [13] K. Su and D. G. Pu, "A nonmonotone filter trust region method for nonlinear constrained optimization," *Journal of Computational and Applied Mathematics*, vol. 223, no. 1, pp. 230–239, 2009.
- [14] P. Y. Nie, "A filter method for solving nonlinear complementarity problems," *Applied Mathematics and Computation*, vol. 167, no. 1, pp. 677–694, 2005.
- [15] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer Series in Operations Research, Springer, New York, NY, USA, 1999.
- [16] D. G. Pu, S. H. Gui, and W. W. Tian, "A class of revised Broyden algorithms without exact line search," *Journal of Computational Mathematics*, vol. 22, no. 1, pp. 11–20, 2004.
- [17] I. Bongartz, A. R. Conn, N. Gould, and P. L. Toint, "CUTE: constrained and unconstrained testing environment," *ACM Transactions on Mathematical Software*, vol. 21, no. 1, pp. 123–160, 1995.
- [18] P. E. Gill, W. Murray, and M. A. Saunders, "SNOPT: an SQP algorithm for large-scale constrained optimization," *SIAM Review*, vol. 47, no. 1, pp. 99–131, 2005.

*Research Article*

## **Optimal Incentive Pricing on Relaying Services for Maximizing Connection Availability in Multihop Cellular Networks**

**Ming-Hua Lin and Hao-Jan Hsu**

*Department of Information Technology and Management, Shih Chien University, No. 70, Ta-Chih Street, Taipei 10462, Taiwan*

Correspondence should be addressed to Ming-Hua Lin, mhlin@mail.usc.edu.tw

Received 16 September 2011; Accepted 12 October 2011

Academic Editor: Yi-Chung Hu

Copyright © 2012 M.-H. Lin and H.-J. Hsu. 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 investigates an incentive pricing problem for relaying services in multihop cellular networks. Providing incentives to encourage mobile nodes to relay data is a critical factor in building successful multihop cellular networks. Most existing approaches adopt fixed-rate or location-based pricing on rewarding packets forwarding. This study applies a mathematical programming model to determine an optimal incentive price for each intermediate node that provides relaying services. Under the obtained incentive price, the connection availability of the networks is maximized by using the same relaying costs as other pricing schemes. A signomial geometric programming problem is constructed, and a deterministic optimization approach is employed to solve the problem. Besides, quality-of-service constraints are added in the proposed model to mitigate the unfairness between connection availabilities of individual nodes. Computational results demonstrate that the proposed model obtains the optimal incentive price on relaying services to maximize connection availability of the networks.

### **1. Introduction**

Over the past few years, wireless networks and wireless devices have rapidly developed and undergone significant advances. More and more services that dramatically affect personal and business communications are provided by wireless access networks. How to build a seamless wireless network has received increasing attention from the practitioners and the researchers. Most wireless networks are based on cellular architecture, which means that a mobile host is handled by a central base station in a limited range. Cellular networks have inherent limitations on cell coverage and the dead spot problem. Traditionally, the network providers utilize more infrastructure equipments such as base stations, to solve

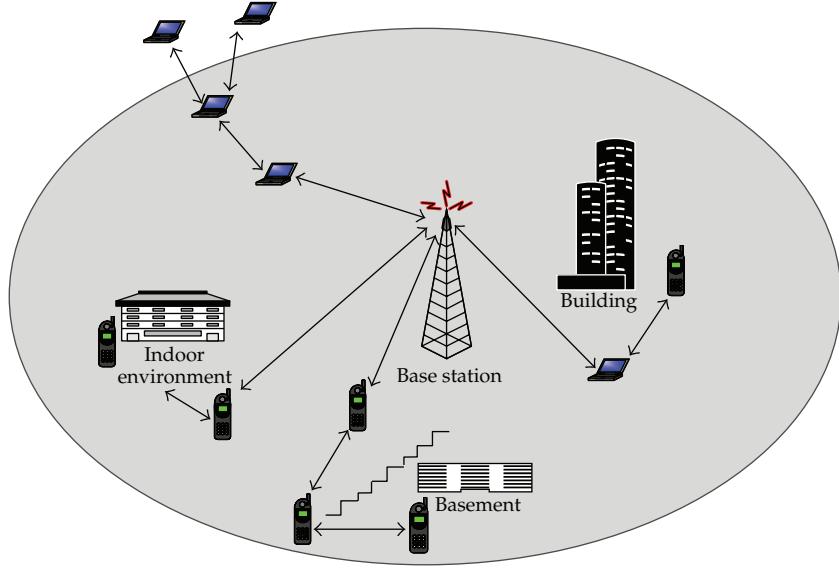
these problems. However, this method is expensive. Therefore relaying technology has been developed to solve this problem. In the last decade, multihop cellular networks have been proposed to harness the benefits of conventional cellular networks and emerging multihop ad hoc networks. In cellular networks, a mobile device directly connects with the base station; in multihop networks, a mobile device communicates with others over peer-to-peer connections. Figure 1 indicates the scenario of general multihop cellular networks. Adopting hop-by-hop connections can extend the service area at the boundaries of the network and eliminate dead spots, including indoor environments and basements. Much research has evaluated and summarized the advantages of multihop cellular networks over existing single-hop cellular networks as follows [1–5].

- (i) Increases the speed of data transmission.
- (ii) Reduces total transmission power.
- (iii) Extends the service area.
- (iv) Increases system capacity.
- (v) Balances traffic load.
- (vi) Reduces the interference with other nodes.
- (vii) Reduces the number of base station sites.

Cooperation among nodes is a critical factor for ensuring the success of the relaying ad hoc networks [2, 6]. In recent years, a number of approaches have been proposed to encourage mobile nodes to relay data for others in ad hoc networks. Most of existing motivation-based approaches focus on a charging protocol and use fixed-rate pricing that gives identical reward level on per unit of packet forwarded. Although the major advantage of fixed-rate pricing is that billing and accounting processes are simple, providing identical reward level to all mobile nodes neglects the distinct importance of each mobile node in the networks. Lo and Lin [2] developed a location-based incentive pricing scheme rewarding each mobile node based on its degree of contribution to successful hop-by-hop connections. Simulation results indicate that their method provides higher connection availability compared to the fixed-rate pricing scheme.

This paper constructs a mathematical programming model to the problem of optimal pricing on relaying services provided by the mobile nodes in the multihop cellular networks. The formulated model that maximizes connection availability of the networks under identical relaying costs used by the fixed-rate pricing scheme and the location-based pricing scheme [2] is a signomial geometric programming (SGP) problem. Convexification strategies and piecewise linearization techniques are employed to reformulate the problem into a convex mixed-integer nonlinear programming (MINLP) problem that can be solved by conventional MINLP methods to reach a global solution. We also add quality-of-service (QoS) constraints in the constructed model to guarantee each mobile node with a minimum successful connection probability, therefore mitigating the unfairness between connection availabilities of individual nodes. Computational experiments are conducted to compare the proposed method with existing pricing schemes. Simulation results indicate that the proposed method obtains higher connection availability of the networks than the existing pricing methods without additional relaying costs.

The rest of the paper is organized as follows. Section 2 reviews existing multihop cellular networking models and incentive pricing models. In Section 3, an incentive pricing model for maximizing connection availability is proposed to determine the optimal price on



**Figure 1:** Scenario of general multihop cellular networks.

relaying services provided by mobile nodes. Section 4 provides a solution approach based on variable transformations and piecewise linearization techniques. In Section 5, we present the computational experiments, and finally concluding remarks are made in Section 6.

## 2. Literature Review

Opportunity-driven multiple access (ODMA) is an ad hoc multihop protocol where the transmissions from the mobile hosts to the base station are broken into multiple wireless hops, thereby reducing transmission power [4]. The high-data-rate coverage of the cell can be increased by adopting relaying technologies for the mobile nodes outside the high-data-rate coverage area. The Ad Hoc GSM (A-GSM) system is a network protocol platform that accommodates relaying capability in GSM cellular networks. Although the GSM system aims to provide global roaming, the dead spot problem still exists, for example, in subway stations and basements. Since installing additional base stations at each dead spot location is not economical, the A-GSM system extends the data communication through the mobile nodes [7]. Qiao and Wu [3] presented an integrated cellular and ad hoc relay (iCAR) system to combine the cellular infrastructure with ad hoc relaying technologies. All cellular networks have problems of limited capacity and unbalanced traffic. Cellular networks probably cannot provide the connection service because some of the cells are heavily congested, but at the same time other cells still have available channels. This kind of centralized obstruction makes the system unable to establish successful communication, even though the number of required channels does not reach the maximum capacity of the entire system. Utilizing relaying technologies a mobile host in a congested cell obtains a free channel in another cell and establishes a new call successfully. Wu et al. [8] proposed a scheme called mobile-assisted data forwarding (MADF) to add an ad hoc overlay to the fixed cellular infrastructure, and special channels are assigned to connect users in a hot cell to its neighboring cold cells without going through the base station in the hot cell. An intermediate forwarding

agent, such as a repeater or another mobile terminal, in the cold cell is required to relay the data to that cell. Wu et al. [8] observed that under a certain delay requirement, the throughput can be greatly improved. Luo et al. [9] proposed a unified cellular and ad hoc network (UCAN) architecture that considers the balanced traffic and network throughput. The UCAN architecture uses relaying technologies to send information to the mobile device if the transmission quality of the channel is poor. Each mobile device in the UCAN model has both a 3G cellular link and IEEE 802.11-based peer-to-peer links. The 3G base station forwards packets for destination clients with poor channel quality to proxy clients with better channel quality. Multihop cellular network is considered as a promising candidate of 4G wireless network for future mobile communications. The complete surveys of technologies advances and economic perspective on the deployment of multihop cellular networks are provided by Li et al. [1] and Manoj et al. [10].

Since forwarding data for others consumes battery energy and delays its own data, providing incentives for mobile nodes to cooperate as relaying entries is necessary. The existing incentive schemes can be classified into detection-based and motivation-based approaches. The detection-based approach finds out the misbehaving nodes and reduces their impact in the networks. Marti et al. [11] developed two methods that find the misbehaving nodes and avoid routing packets through these nodes. Michiardi and Molva [12] proposed a mechanism to enforce cooperation among nodes based on reputation and to prevent denial of service attacks because of selfishness. Buchegger and Le Boudec [13] developed a protocol to detect and isolate misconduct nodes, therefore making it unattractive to deny cooperation.

Instead of discouraging misbehavior by punishing misbehavior node, the motivation-based approach encourages positive cooperation by rewarding incentives for relaying packets. Buttyán and Hubaux [6, 14] developed different approaches to provide incentives to cooperative nodes, therefore simulating packet forwarding. Buttyán and Hubaux [14] did not discuss the reward level, and Buttyán and Hubaux [6] suggested to reward the relaying service based on the number of forwarding packets. Jakobsson et al. [15] developed a micro-payment scheme to encourage collaboration and discourage dishonest behavior in multihop cellular networks. A subject reward level is determined according to the importance of the packet. Lamparter et al. [16] proposed a charging scheme using volume-based pricing. A fixed price per unit of data is rewarded for forwarding traffic in ad hoc stub networks. The rewarding mechanisms mentioned above adopt fixed-rate pricing and do not consider the importance of each mobile node in the routing topology.

Lo and Lin [2] proposed a location-based incentive pricing scheme that adjusts the price of incentives for packet forwarding based on the degree of each mobile nodes contributing to successful hop-by-hop connections. Since the willingness of the mobile node to relay packets has a significant impact on the success of the multihop connections from all nodes in its subtree to the base station, the importance of a mobile node depends on the number of mobile nodes in its subtree. They defined the location index  $LI_v$  of a mobile node  $v$  as the number of nodes residing in the tree rooted at node  $v$ . Let  $N$  be the set of intermediate nodes providing relaying services for the mobile nodes that require hop-by-hop connections to the base station, and ALI be the average location index of all nodes in  $N$ ; the price of the feedback incentives for node  $v$ ,  $p_v$ , is defined as follows [2]:

$$p_v = p_0 + (LI_v - ALI) \cdot \frac{R_p}{R_{LI}} \cdot \frac{1}{LI_v}, \quad (2.1)$$

where  $R_p = \min\{p_0, P_{\max} - p_0\}$ ,  $R_{LI} = \max\{\text{ALI} - \min_{v \in N}\{LI_v\}, \max_{v \in N}\{LI_v\} - \text{ALI}\}$ ,  $p_0$  is the price used in the fixed-rate pricing method, and  $P_{\max}$  is the maximum price the network provider can reward to an intermediate mobile node. Equation (2.1) employs  $p_0$  as a basic price and gives a higher incentive price on relaying services for the node with a higher location index. Because some incentive rewards are shifted from the nodes of low importance to the node of high importance, the Lo and Lin [2] pricing scheme results in higher connection availability but does not generate higher relaying costs compared to the fixed-rate pricing scheme. However, their method does not provide an optimal incentive pricing solution that maximizes connection availability of the networks.

### 3. Proposed Incentive Pricing Model

#### 3.1. Connection Availability Maximization Problem

Pricing is an inducer for suppliers to provide services. Monetary incentives can affect the motivation of mobile nodes providing services and are usually characterized by a supply function that represents the reaction of mobile nodes to the change of the price [17]. In this paper, we assume a linear relationship between the price of incentives and the willingness of forwarding packets [18], that is,

$$S(p_v) = \frac{p_v}{P_{\max}}, \quad 0 \leq p_v \leq P_{\max}, \quad (3.1)$$

where  $p_v$  is the incentive price on per unit of relayed data,  $P_{\max}$  is the maximum price the network provider can reward to an intermediate mobile node per unit of relayed data, and  $S(p_v)$  is the willingness of forwarding packets under the incentive price  $p_v$ .  $S(p_v)$  is the supply function representing the reaction of mobile nodes to the change in the price of the incentives.  $S(p_v = 0) = 0$  means that node  $v$  will not relay traffic for others if no feedback is provided for relaying services. The willingness of forwarding packets linearly increases as the incentive price on relaying services increases.  $S(p_v = P_{\max}) = 1$  means the maximum price is acceptable for all mobile nodes to provide relaying services.

In multihop cellular networks, data packets must be relayed hop by hop from a given mobile node to a base station; thus the connection availability of node  $i$  depends on the willingness of all intermediate mobile nodes on the routing path to forward packets. Let  $\text{CA}_i$  be the connection availability of node  $i$ , that is, the successful connection probability from node  $i$  to the base station.  $\text{CA}_i$  can be expressed as [2]

$$\text{CA}_i = \prod_{v \in M_i} S(p_v), \quad (3.2)$$

where  $M_i$  is the set of intermediate nodes in the path from node  $i$  to the base station, and all the other variables are the same as defined before.

The connection availability maximization problem in the multihop cellular networks considered in this paper can be formulated as follows:

$$\text{Maximize} \quad \frac{(\sum_{i=1}^w \text{CA}_i)}{w} \quad (3.3)$$

$$\text{subject to} \quad \text{CA}_i = \prod_{v \in M_i} \left( \frac{p_v}{P_{\max}} \right), \quad i = 1, \dots, w, \quad (3.4)$$

$$\sum_{i=1}^w \left( T_i \cdot \sum_{v \in M_i} p_v \right) \leq \sum_{i=1}^w \left( T_i \cdot \sum_{v \in M_i} P_{\text{fixed}} \right), \quad (3.5)$$

$$0 \leq p_v \leq P_{\max}, \quad (3.6)$$

where  $w$  is the number of nodes requiring hop-by-hop connections to the base station in the networks,  $T_i$  is the units of traffic sent by node  $i$ , and  $P_{\text{fixed}}$  is the fixed incentive price on relaying services used by the fixed-rate pricing scheme. The objective function aims to maximize the connection availability of the networks, that is, the average connection availability of all mobile nodes using hop-by-hop connections to the base station. Lo and Lin [2] refers the objective function as service availability. Constraint (3.5) indicates the total relaying costs of the proposed method are not greater than the total relaying costs of the fixed-rate pricing scheme.

### 3.2. Connection Availability Maximization Problem with QoS Requirements

In the numerical examples, we find the connection availabilities of some mobile nodes are zero by using the proposed model described previously. In order to alleviate the unfairness situation between connection availabilities of individual nodes, this study employs QoS constraints in the original model to guarantee each mobile node with a minimum successful connection probability. The connection availability maximization problem with QoS requirements considered in this study can be formulated as follows:

$$\begin{aligned} & \text{Maximize} && (3.3) \\ & \text{subject to} && (3.4) \sim (3.6), \end{aligned} \quad (3.7)$$

$$\text{CA}_i \geq \text{QoS}_{\text{CA}}, \quad i = 1, \dots, w, \quad (3.8)$$

$$\text{QoS}_{\text{CA}} = \begin{cases} \text{Min CA}_{\text{LB}}, & \text{if } \text{Min CA}_{\text{LB}} > 0, \\ 0.01, & \text{if } \text{Min CA}_{\text{LB}} = 0, \end{cases} \quad (3.9)$$

where  $\text{Min CA}_{\text{LB}}$  represents the minimal connection availability of all mobile nodes requiring hop-by-hop connections to the base station in the location-based pricing scheme. Constraint (3.8) indicates that the connection availability of each mobile node satisfies the required QoS level ( $\text{QoS}_{\text{CA}}$ ). If the minimal connection availability of all mobile nodes in the location-based pricing scheme is greater than zero, then the required QoS level is set as  $\text{Min CA}_{\text{LB}}$ . Otherwise, the required QoS level is set as 0.01.

#### 4. Problem-Solving Approach

Since the problem described in the previous section is an SGP problem, that is, a class of nonconvex programming problems. SGP problems generally possess multiple local optima and experience much more theoretical and computational difficulties. This study uses variable transformations and piecewise linearization techniques to reformulate the problem into a convex MINLP problem that can be globally solved by conventional MINLP methods. Much research has proposed variable transformation techniques to solve optimization problems including signomial functions to global optimality [19–22]. For convexifying positive signomial terms, Lundell and Westerlund [23] proved that the exponential transformation always results in a tighter underestimator than the negative power transformation. This study applies the exponential transformation to convexify a positive signomial function  $\prod_{i=1}^n x_i^{\alpha_i}$  by the following remark [19, 24].

*Remark 4.1.* If  $\alpha_j > 0$  for some  $j, j \notin I, I = \{k \mid \alpha_k < 0, k = 1, 2, \dots, n\}$ , then we convert  $\prod_{i=1}^n x_i^{\alpha_i}$  into another function  $(\prod_{i \in I} x_i^{\alpha_i}) e^{\sum_{j \notin I} \alpha_j y_j}$ , where  $y_j = L(\ln x_j)$  and  $L(\ln x_j)$ , is a piecewise linear function of  $\ln x_j$ . Then  $(\prod_{i \in I} x_i^{\alpha_i}) e^{\sum_{j \notin I} \alpha_j y_j}$  where  $x_i > 0, i \in I, y_j \in \mathbb{R}, j \notin I$  is a convex function.

For convexifying negative signomial terms, we apply the power transformation to reformulate a negative signomial function  $-\prod_{i=1}^n x_i^{\alpha_i}$  by the following remark [19–22, 24].

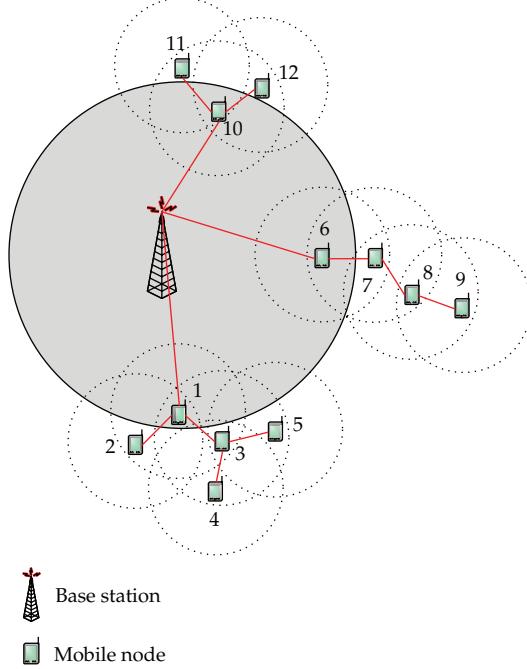
*Remark 4.2.* If  $0 \leq \alpha_1 \leq \alpha_2 \leq \dots \leq \alpha_p, 0 \geq \alpha_{p+1} \geq \alpha_{p+2} \geq \dots \geq \alpha_n$ , and  $\sum_{i=1}^r \alpha_i < 1$  for some largest integer  $r$ , such that  $r \leq p, I = \{k \mid k = 1, 2, \dots, p\}$ , then we convert  $-\prod_{i=1}^n x_i^{\alpha_i}$  into another function  $-\prod_{i \in I} x_i^{\alpha_i} \prod_{j \notin I} y_j^\beta, \beta = (1 - \sum_{i=1}^r \alpha_i) / (n - r)$ , where  $y_j = L(x_j^{\alpha_j/\beta})$  and  $L(x_j^{\alpha_j/\beta})$  is a piecewise linear function of  $x_j^{\alpha_j/\beta}$ . Then  $-\prod_{i \in I} x_i^{\alpha_i} \prod_{j \notin I} y_j^\beta$ , where  $x_i > 0, i \in I, y_j \in \mathbb{R}, j \notin I$ , is a convex function.

Herein the concept of special ordered set of type 2 (SOS-2) constraints can be utilized to formulate the piecewise linear function [25, 26]. This study adopts the piecewise linearization technique introduced by Vielma and Nemhauser [27] that uses a logarithmic number of binary variables and extra constraints. The computational results in [27] show that their piecewise linearization technique outperforms other piecewise linearization formulations. Tsai and Lin [28] applied the piecewise linearization technique developed by Vielma and Nemhauser [27] to efficiently solve posynomial geometric programming problems.

The original model has one nonconvex objective function, one constraint, and  $n$  variables, where  $n$  is the number of intermediate nodes providing relaying services. The reformulated model has one convex objective function, one constraint,  $n$  variables, and several piecewise linear functions. The number of piecewise linear functions depends on convexification process in reformulating the problem. For each piecewise linear function,  $\lceil \log_2 m \rceil$  binary variables,  $m+1$  continuous variables, and  $3+2\lceil \log_2 m \rceil$  constraints are required to express a piecewise linear function with  $m$  line segments.

The following example is used to illustrate how the proposed method discussed previously determines the incentive price on relaying services provided by each mobile node.

*Example 4.3.* Consider an example taken from Lo and Lin [2] with twelve mobile nodes distributed in the multihop cellular networks indicated in Figure 2. Nine nodes (nodes 2, 3, 4, 5, 7, 8, 9, 11, 12) require hop-by-hop connections to reach the central base station.



**Figure 2:** Relaying topology of Example 4.3 [2].

Assume each mobile node has identical traffic load  $u$  units, and the fixed-rate pricing scheme uses  $0.5P_{\max}$  as the incentive price for relaying per unit of data, that is,  $P_{\text{fixed}} = 0.5P_{\max}$ . A mathematical programming model can be constructed for the connection availability maximization problem as follows:

$$\begin{aligned}
 \text{Maximize} \quad & \frac{(CA_2 + CA_3 + CA_4 + CA_5 + CA_7 + CA_8 + CA_9 + CA_{11} + CA_{12})}{9} \\
 \text{subject to} \quad & CA_2 = CA_3 = \frac{p_1}{P_{\max}}, \\
 & CA_4 = CA_5 = \frac{p_1}{P_{\max}} \cdot \frac{p_3}{P_{\max}}, \\
 & CA_7 = \frac{p_6}{P_{\max}}, \\
 & CA_8 = \frac{p_6}{P_{\max}} \cdot \frac{p_7}{P_{\max}}, \\
 & CA_9 = \frac{p_6}{P_{\max}} \cdot \frac{p_7}{P_{\max}} \cdot \frac{p_8}{P_{\max}}, \\
 & CA_{11} = CA_{12} = \frac{p_{10}}{P_{\max}},
 \end{aligned}$$

$$\begin{aligned}
& 2u \cdot p_1 + 2u \cdot (p_1 + p_3) + u \cdot (p_6) + u \cdot (p_6 + p_7) + u \cdot (p_6 + p_7 + p_8) + 2u \cdot (p_{10}) \\
& \leq 2u \cdot (0.5P_{\max}) + 2u \cdot (0.5P_{\max} + 0.5P_{\max}) + u \cdot (0.5P_{\max}) \\
& \quad + u \cdot (0.5P_{\max} + 0.5P_{\max}) + u \cdot (0.5P_{\max} + 0.5P_{\max} + 0.5P_{\max}) + 2u \cdot (0.5P_{\max}), \\
& 0 \leq p_i \leq P_{\max}, \quad i = 1, 3, 6, 7, 8, 10,
\end{aligned} \tag{4.1}$$

where  $CA_i$ ,  $i = 2, 3, 4, 5, 7, 8, 9, 11, 12$ , represents the connection availability of mobile node  $i$ , and  $P_{\max}$  and  $u$  are constants. This program is a nonconvex SGP problem. Applying the method mentioned previously, the connection maximization problem of Example 4.3 can be converted into a convex MINLP problem as follows:

$$\text{Minimize} \quad \left( -2 \frac{p_1}{P_{\max}} - 2 \frac{y_1^{0.5}}{P_{\max}} \cdot \frac{y_3^{0.5}}{P_{\max}} - \frac{p_6}{P_{\max}} - \frac{y_6^{1/3}}{P_{\max}} \cdot \frac{y_7^{1/3}}{P_{\max}} - \frac{y_6^{1/3}}{P_{\max}} \cdot \frac{y_7^{1/3}}{P_{\max}} \cdot \frac{y_8^{1/3}}{P_{\max}} - 2 \frac{p_{10}}{P_{\max}} \right) / 9 \tag{4.2}$$

$$\begin{aligned}
\text{subject to} \quad & 2u \cdot p_1 + 2u \cdot (p_1 + p_3) + u \cdot (p_6) + u \cdot (p_6 + p_7) + u \cdot (p_6 + p_7 + p_8) + 2u \cdot (p_{10}) \\
& \leq 7uP_{\max}, \\
& y_1 = L(p_1^2), \quad y_3 = L(p_3^2), \quad y_6 = L(p_6^3), \quad y_7 = L(p_7^3), \quad y_8 = L(p_8^3), \\
& 0 \leq p_i \leq P_{\max}, \quad i = 1, 3, 6, 7, 8, 10,
\end{aligned} \tag{4.3}$$

where  $L(p_1^2)$ ,  $L(p_3^2)$ ,  $L(p_6^3)$ ,  $L(p_7^3)$ , and  $L(p_8^3)$  are piecewise linear functions of  $p_1^2$ ,  $p_3^2$ ,  $p_6^3$ ,  $p_7^3$ , and  $p_8^3$ , respectively. By using the efficient piecewise linearization technique introduced by Vielma and Nemhauser [27], this program is reformulated as a convex MINLP problem that can be solved on LINGO [29] to obtain a global solution  $(p_1, p_3, p_6, p_7, p_8, p_{10}) = (0.8763P_{\max}, 0.7474P_{\max}, 0, 0, 0, P_{\max})$ . Table 1 compares the incentive price on relaying services and connection availability of the networks under different pricing schemes. Herein the fixed-rate pricing scheme rewards each mobile node  $0.5P_{\max}$  for relaying per unit of data. From the data listed in Table 1, we find the proposed pricing scheme provides higher connection availability of the networks than the fixed-rate pricing scheme and the location-based pricing scheme do. We also observe that the three methods use approximately the same relaying costs.

From Table 1, we find out that although the proposed method has better connection availability of the networks than the other two methods, the connection availabilities of some mobile nodes ( $CA_7$ ,  $CA_8$ ,  $CA_9$ ) are zero; that is, these nodes cannot connect to the base station. This study adds the QoS constraints  $CA_i \geq 0.026$ ,  $i = 2, 3, 4, 5, 7, 8, 9, 11, 12$ , to guarantee each mobile node with a minimum successful connection probability. The required QoS level 0.026 is the minimal individual connection availability obtained from

**Table 1:** Comparison between the fixed-rate pricing scheme, the location-based pricing scheme, and the proposed pricing scheme of Example 4.3.

	Fixed-rate pricing scheme	Location-based pricing scheme	Proposed pricing scheme
Incentive price on relaying services	$p_1 = p_3 = p_6 = p_7 = p_8 = p_{10} = 0.5P_{\max}$	$p_1 = 0.625P_{\max}$ $p_6 = 0.567P_{\max}$ $p_3 = p_7 = p_{10} = 0.45P_{\max}$ $p_8 = 0.1P_{\max}$	$p_1 = 0.8763P_{\max}$ $p_3 = 0.7474P_{\max}$ $p_6 = p_7 = p_8 = 0$ $p_{10} = P_{\max}$
Connection availability of each node	$CA_2 = CA_3 = S(p_1) = 0.5$ $CA_4 = CA_5 = S(p_1)S(p_3) = 0.25$ $CA_7 = S(p_6) = 0.5$ $CA_8 = S(p_6)S(p_7) = 0.25$ $CA_9 = S(p_6)S(p_7)S(p_8) = 0.125$ $CA_{11} = CA_{12} = S(p_{10}) = 0.5$	$CA_2 = CA_3 = S(p_1) = 0.625$ $CA_4 = CA_5 = S(p_1)S(p_3) = 0.281$ $CA_7 = S(p_6) = 0.567$ $CA_8 = S(p_6)S(p_7) = 0.255$ $CA_9 = S(p_6)S(p_7)S(p_8) = 0.026$ $CA_{11} = CA_{12} = S(p_{10}) = 0.45$	$CA_2 = CA_3 = S(p_1) = 0.8763$ $CA_4 = CA_5 = S(p_1)S(p_3) = 0.6549$ $CA_7 = S(p_6) = 0$ $CA_8 = S(p_6)S(p_7) = 0$ $CA_9 = S(p_6)S(p_7)S(p_8) = 0$ $CA_{11} = CA_{12} = S(p_{10}) = 1$
Connection availability of the networks	$(0.5 + 0.5 + 0.25 + 0.25 + 0.5 + 0.25 + 0.125 + 0.5 + 0.5)/9 = 0.375$	$(0.625 + 0.625 + 0.281 + 0.281 + 0.567 + 0.255 + 0.026 + 0.45 + 0.45)/9 = 0.3956$	$(0.8763 + 0.8763 + 0.6549 + 0.6549 + 0 + 0 + 0 + 1 + 1)/9 = 0.5625$
Relaying costs	$2u(0.5P_{\max}) + 2u(0.5P_{\max}) + 0.5P_{\max} + u(0.5P_{\max}) + u(0.5P_{\max} + 0.5P_{\max}) + u(0.5P_{\max} + 0.5P_{\max} + 0.5P_{\max}) + 2u(0.5P_{\max}) = 7uP_{\max}$	$2u(0.625P_{\max}) + 2u(0.625P_{\max}) + 0.45P_{\max} + u(0.567P_{\max}) + u(0.45P_{\max} + 0.567P_{\max}) + u(0.1P_{\max} + 0.45P_{\max}) + 0.567P_{\max} + 2u(0.45P_{\max}) = 7.001uP_{\max}$	$2u(0.8763P_{\max}) + 2u(0.8763P_{\max}) + 0.7474P_{\max} + 2u(P_{\max}) = 7uP_{\max}$

the location-based pricing scheme. The proposed model with QoS requirements becomes as follows:

$$\text{Minimize} \quad (4.2)$$

$$\text{subject to} \quad (4.3)$$

$$\begin{aligned} -\frac{p_1}{P_{\max}} &< -0.026, & -\frac{y_1^{0.5}}{P_{\max}} \cdot \frac{y_3^{0.5}}{P_{\max}} &< -0.026, & -\frac{p_6}{P_{\max}} &< -0.026, \\ -\frac{y_6^{1/3}}{P_{\max}} \cdot \frac{y_7^{1/3}}{P_{\max}} &< -0.026, & -\frac{y_6^{1/3}}{P_{\max}} \cdot \frac{y_7^{1/3}}{P_{\max}} \cdot \frac{y_8^{1/3}}{P_{\max}} &< -0.026, & \frac{p_{10}}{P_{\max}} &< -0.026. \end{aligned} \quad (4.4)$$

Solving this problem can obtain a globally optimal solution  $(p_1, p_3, p_6, p_7, p_8, p_{10}) = (0.6539P_{\max}, 0.3085P_{\max}, 0.3165P_{\max}, 0.2313P_{\max}, 0.3550P_{\max}, P_{\max})$ . Table 2 shows comparison between the location-based pricing scheme and the proposed pricing scheme with QoS requirements of Example 4.3. We find the minimal individual connection availability from the proposed pricing scheme is equal to that from the location-based pricing scheme, and the

**Table 2:** Comparison between the location-based pricing scheme and the proposed pricing scheme with QoS requirements of Example 4.3.

	Location-based pricing scheme	Proposed pricing scheme with QoS requirements
Incentive price on relaying services	$p_1 = 0.625P_{\max}$	$p_1 = 0.6539P_{\max}$
	$p_6 = 0.567P_{\max}$	$p_3 = 0.3085P_{\max}$
	$p_3 = p_7 = p_{10} = 0.45P_{\max}$	$p_6 = 0.3165P_{\max}$
	$p_8 = 0.1P_{\max}$	$p_7 = 0.2313P_{\max}$
		$p_8 = 0.3550P_{\max}$
		$p_{10} = P_{\max}$
Connection availability of each node	$CA_2 = CA_3 = S(p_1) = 0.625$	$CA_2 = CA_3 = S(p_1) = 0.6539$
	$CA_4 = CA_5 = S(p_1)S(p_3) = 0.281$	$CA_4 = CA_5 = S(p_1)S(p_3) = 0.2017$
	$CA_7 = S(p_6) = 0.567$	$CA_7 = S(p_6) = 0.3165$
	$CA_8 = S(p_6)S(p_7) = 0.255$	$CA_8 = S(p_6)S(p_7) = 0.0732$
	$CA_9 = S(p_6)S(p_7)S(p_8) = 0.026$	$CA_9 = S(p_6)S(p_7)S(p_8) = 0.0260$
	$CA_{11} = CA_{12} = S(p_{10}) = 0.45$	$CA_{11} = CA_{12} = S(p_{10}) = 1$
Connection availability of the networks	$(0.625 + 0.625 + 0.281 + 0.281 + 0.567 + 0.255 + 0.026 + 0.45 + 0.45)/9 = 0.3956$	$(0.6539 + 0.6539 + 0.2017 + 0.2017 + 0.3165 + 0.0732 + 0.0260 + 1 + 1)/9 = 0.4585$
Relaying costs	$2u(0.625P_{\max}) + 2u(0.625P_{\max} + 0.45P_{\max}) + u(0.567P_{\max}) + u(0.45P_{\max} + 0.567P_{\max}) + u(0.1P_{\max} + 0.45P_{\max} + 0.567P_{\max}) + 2u(0.45P_{\max}) = 7.001P_{\max}$	$2u(0.6539P_{\max}) + 2u(0.6539P_{\max} + 0.3085P_{\max}) + u(0.3165P_{\max}) + u(0.3165P_{\max} + 0.2313P_{\max}) + P_{\max} + u(0.3165P_{\max} + 0.2313P_{\max} + 0.3550P_{\max}) + 2u(P_{\max}) = 6.9997u$

connection availability of the networks from the proposed pricing scheme is still higher than that from the location-based pricing scheme.

## 5. Numerical Experiments

### 5.1. Fixed-Rate Method versus Location-Based Method versus Proposed Method

This section describes the simulation results for verifying the advantages of the proposed pricing scheme. We design our simulation tests by C++ language. All simulations are run on a Notebook with an Intel CPU P8700 and 4 GB RAM. The simulation environment is a rectangular region of 100 units width and 100 units height with a single base station of 30 units radius located in the central point. The radius of each mobile node is 20 units. In this study, a shortest path tree is built such that each mobile node connects to the base station with a minimum number of hops.

In the experiments 32, 64, and 128, mobile nodes, respectively, are randomly distributed in the rectangular region. 10 simulations are run for each set of parameter settings. Table 3 compares average connection availability of the networks of 10 simulations by different incentive pricing schemes. Figure 3 indicates that the proposed pricing scheme obtains higher average connection availability than the fixed-rate pricing scheme and the

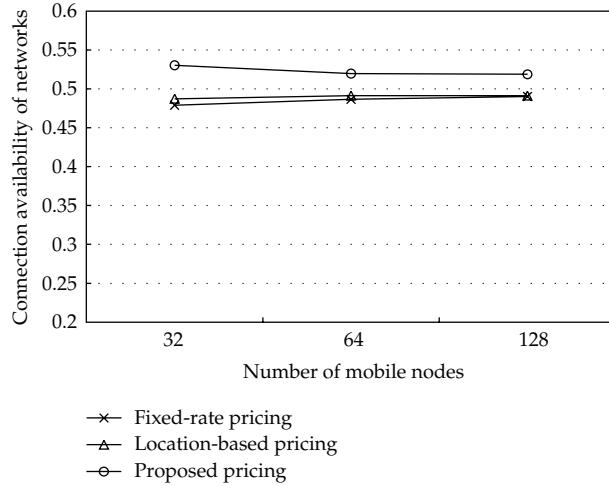
**Table 3:** Comparison of connection availability of the networks by three methods in the simulation space of (width, height) = (100 units, 100 units).

Number of mobile nodes	CA <sub>FR</sub>	CA <sub>LB</sub>	CA <sub>P</sub>	(CA <sub>P</sub> – CA <sub>FR</sub> )/CA <sub>FR</sub>	(CA <sub>P</sub> – CA <sub>LB</sub> )/CA <sub>LB</sub>	Average path length
32	0.47898643	0.48730129	0.53038439	10.82%	8.88%	1.0840542
64	0.48652063	0.49130662	0.51966696	6.87%	5.81%	1.0539176
128	0.49012412	0.49131695	0.51875024	5.88%	5.62%	1.0395036

CA<sub>FR</sub>: connection availability of the networks from the fixed-rate pricing scheme.

CA<sub>LB</sub>: connection availability of the networks from the location-based pricing scheme.

CA<sub>P</sub>: connection availability of the networks from the proposed pricing scheme.



**Figure 3:** Comparison of connection availability of the networks by three methods in the simulation space of (width, height) = (100 units, 100 units).

location-based pricing scheme under different number of mobile nodes in the simulation environment. As the number of mobile nodes in the networks increases, the mobile nodes are easier to find a shorter hop-by-hop path for connecting to the base station. Therefore the average path length decreases when the number of mobile nodes increases. The effect on improving connection availability of the networks by the proposed method is more significant when the average path is longer.

## 5.2. Larger Simulation Space

To investigate the advantages of the proposed pricing scheme under a longer path, in this section we change the simulation space to a rectangular region of 200 units width and 200 units height. If the simulation area becomes larger, the path of the hop-by-hop connection to the base station required by a mobile node will be longer. Then the impact of the path length on the performance of the proposed pricing method can be observed. Table 4 shows the average connection availability of the networks of 10 simulations in the rectangular region of 200 units width and 200 units height. The average connection availability of the networks obtained by the proposed pricing scheme is higher than the other two pricing

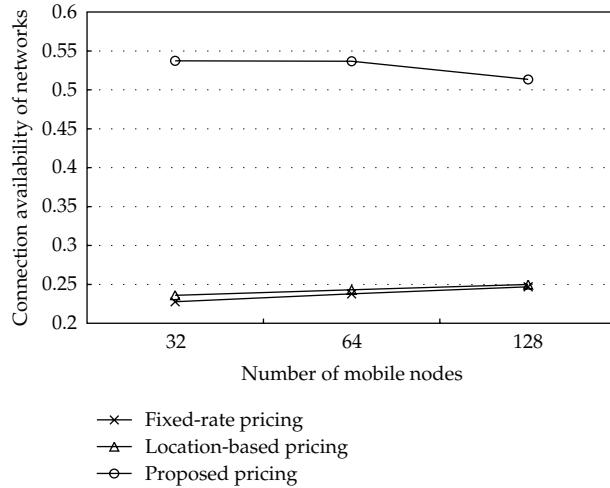
**Table 4:** Comparison of connection availability of the networks by three methods in the simulation space of (width, height) = (200 units, 200 units).

Number of mobile nodes	CA <sub>FR</sub>	CA <sub>LB</sub>	CA <sub>P</sub>	(CA <sub>P</sub> - CA <sub>FR</sub> ) / CA <sub>FR</sub>	(CA <sub>P</sub> - CA <sub>LB</sub> ) / CA <sub>LB</sub>	Average path length
32	0.22764841	0.23610142	0.53722796	138.12%	130.04%	2.6502985
64	0.23798218	0.24306587	0.53669912	125.83%	121.06%	2.5031434
128	0.24686002	0.24986204	0.51335212	108.03%	105.54%	2.3962481

CA<sub>FR</sub>: connection availability of the networks from the fixed-rate pricing scheme.

CA<sub>LB</sub>: connection availability of the networks from the location-based pricing scheme.

CA<sub>P</sub>: connection availability of the networks from the proposed pricing scheme.



**Figure 4:** Comparison of connection availability of the networks by three methods in the simulation space of (width, height) = (200 units, 200 units).

models. Figure 4 indicates that in the simulation space of 200 units width and 200 units height, the difference in the connection availability of the networks obtained by the fixed-rate pricing scheme and the location-based pricing scheme is not obvious. However, the connection availability of the networks by the proposed method is much higher than that by the other two methods if a longer path is required to reach the base station.

### 5.3. Location-Based Method versus Proposed Method with QoS Requirements

Section 4 gives an example that indicates the proposed pricing scheme results in some unfairness, and some of the node's connection availabilities are zero. This section performs several simulations to verify that adding QoS constraints on the connection availability of each mobile node can mitigate the unfairness situation. Each mobile node is guaranteed to obtain the minimum connection availability by taking the minimum connection availability from the location-based pricing scheme. If the minimum connection availability from the location-based pricing scheme is zero, then we move the required level of individual connection availability to 0.01.

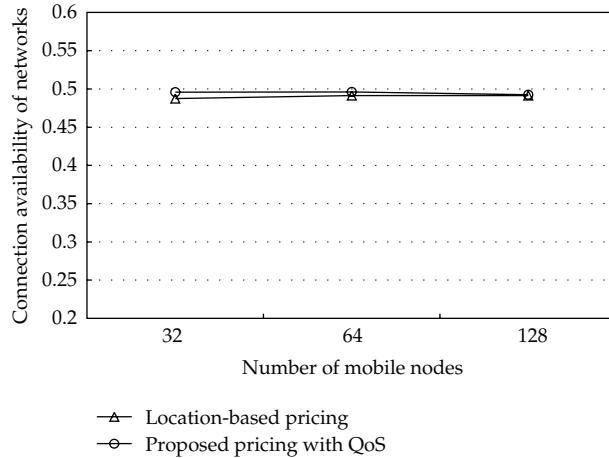
Tables 5 and 6 show the average connection availability of the networks of 10 simulations in the simulation space of (width, height) = (100 units, 100 units) and (width,

**Table 5:** Comparison of connection availability by two methods in the simulation space of (width, height) = (100 units, 100 units).

Number of mobile nodes	CA <sub>LB</sub>	CA <sub>P,QoS</sub>	(CA <sub>P,QoS</sub> - CA <sub>LB</sub> ) / CA <sub>LB</sub>
32	0.48730129	0.49575548	1.74%
64	0.49130662	0.49617013	1.00%
128	0.49131695	0.49246111	0.24%

CA<sub>LB</sub>: connection availability of the networks from the location-based pricing scheme.

CA<sub>P,QoS</sub>: connection availability of the networks from the proposed pricing scheme with QoS requirements.



**Figure 5:** Comparison of connection availability of the networks by two methods in the simulation space of (width, height) = (100 units, 100 units).

height) = (200 units, 200 units), respectively. The comparisons are also indicated in Figures 5 and 6. Compared to the results from the proposed pricing method without QoS constraints in Tables 3 and 4, although the connection availability of the networks decreases, the proposed pricing method with QoS requirements still performs better than the location-based pricing method. Since the minimum individual node's connection availability in the proposed pricing method is greater than or equal to that in the location-based pricing method, adding QoS constraints makes the proposed pricing method consider both connection availability of the networks and fairness between individual node's connection availabilities.

## 6. Conclusions

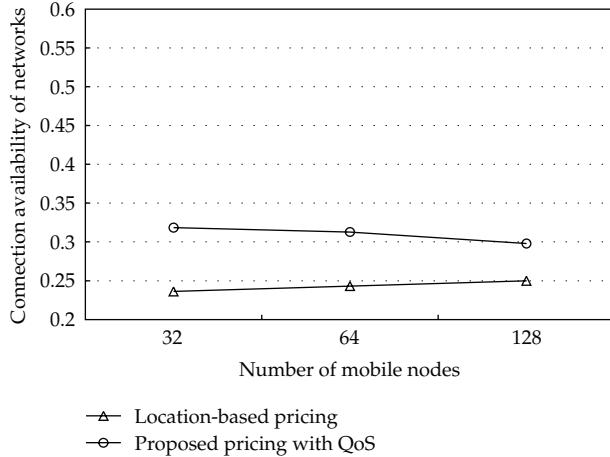
Cost savings and connection availability are two crucial issues of a network provider adopting multihop cellular networking technology. This paper determines the optimal incentive price on relaying services for each mobile node by constructing a mathematical programming model that maximizes connection availability without extra relaying costs. A deterministic optimization approach based on variable transformations and piecewise linearization techniques is utilized to solve the formulated problem. Simulation results demonstrate that the proposed pricing model results in higher connection availability than the fixed-rate pricing scheme and the location-based pricing scheme. In addition, a mathematical programming model involving QoS requirements in connection availability of

**Table 6:** Comparison of connection availability of the networks by two methods in the simulation space of (width, height) = (200 units, 200 units).

Number of mobile nodes	CA <sub>LB</sub>	CA <sub>P,QoS</sub>	(CA <sub>P,QoS</sub> - CA <sub>LB</sub> ) / CA <sub>LB</sub>
32	0.23610142	0.3183623	34.84%
64	0.24306587	0.31280795	28.69%
128	0.24986204	0.29797671	19.26%

CA<sub>LB</sub>: connection availability of the networks from the location-based pricing scheme.

CA<sub>P,QoS</sub>: connection availability of the networks from the proposed pricing scheme with QoS requirements.



**Figure 6:** Comparison of connection availability of the networks by two methods in simulation space of (width, height) = (200 units, 200 units).

each individual mobile node is developed to eliminate the unfairness situation in the original model.

## Acknowledgments

The authors thank the anonymous referees for contributing their valuable comments regarding this paper and thus significantly improving the quality of this paper. The research is supported by Taiwan NSC Grant NSC 99-2410-H-158-010-MY2.

## References

- [1] X. J. Li, B. C. Seet, and P. H. J. Chong, "Multihop cellular networks: technology and economics," *Computer Networks*, vol. 52, no. 9, pp. 1825–1837, 2008.
- [2] C. C. Lo and M. H. Lin, "Location-based incentive pricing for tree-based relaying in multi-hop cellular networks," *Computer Communications*, vol. 29, no. 16, pp. 3148–3157, 2006.
- [3] C. Qiao and H. Wu, "ICAR: an intelligent cellular and ad-hoc relay system," in *Proceedings of the IEEE International Conference on Computer Communications and Networks (IC3N '00)*, pp. 154–161, Las Vegas, Nev, USA, October 2000.
- [4] T. Rouse, I. Band, and S. McLaughlin, "Capacity and power investigation of Opportunity Driven Multiple Access (ODMA) networks in TDD-CDMA based systems," in *Proceedings of the International Conference on Communications (ICC 2002)*, pp. 3202–3206, New York, NY, USA, May 2002.

- [5] N. Ben Salem, L. Buttyán, J. P. Hubaux, and M. Jakobsson, "A charging and rewarding scheme for packet forwarding in multi-hop cellular networks," in *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking and Computing (MobiHoc '03)*, pp. 13–24, Annapolis, Md, USA, June 2003.
- [6] L. Buttyán and J. P. Hubaux, "Stimulating cooperation in self-organizing mobile ad hoc networks," *Mobile Networks and Applications*, vol. 8, no. 5, pp. 579–592, 2003.
- [7] G. N. Aggélou and R. Tafazolli, "On the relaying capability of next-generation GSM cellular networks," *IEEE Personal Communications*, vol. 8, no. 1, pp. 40–47, 2001.
- [8] X. Wu, S. -H.G. Chan, and B. Mukherjee, "MADF: a novel approach to add an ad-hoc overlay on a fixed cellular infrastructure," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '02)*, vol. 2, pp. 549–554, Chicago, Ill, USA, September 2002.
- [9] H. Luo, R. Ramjee, P. Sinha, L. Li, and S. Lu, "UCAN: a unified cellular and ad-hoc network architecture," in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MobiCom '03)*, pp. 353–367, San Diego, Calif, USA, September 2003.
- [10] B. S. Manoj, R. Ananthapadmanabha, and C. S. R. Murthy, "Multi-hop cellular networks: architecture and protocols for best-effort and real-time communication," *Journal of Parallel and Distributed Computing*, vol. 65, no. 6, pp. 767–791, 2005.
- [11] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom '00)*, pp. 255–265, Boston, Mass, USA, August 2000.
- [12] P. Michiardi and R. Molva, "Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Proceedings of the IFIP TC6/TC11 6th Joint Working Conference on Communications and Multimedia Security*, pp. 107–121, Portorož, Slovenia, September 2002.
- [13] S. Buchegger and J. Y. Le Boudec, "Performance analysis of the CONFIDANT protocol (Cooperation of nodes: fairness in dynamic ad-hoc networks)," in *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking and Computing (MobiHoc '02)*, pp. 226–236, Lausanne, Switzerland, June 2002.
- [14] L. Buttyán and J.-P. Hubaux, "Enforcing service availability in mobile ad-hoc WANs," in *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking and Computing (MobiHoc '00)*, pp. 87–96, Boston, Mass, USA, August 2000.
- [15] M. Jakobsson, J. P. Hubaux, and L. Buttyán, "A micro-payment scheme encouraging collaboration in multi-hop cellular networks," *Lecture Notes in Computer Science*, vol. 2742, pp. 15–33, 2003.
- [16] B. Lamparter, K. Paul, and D. Westhoff, "Charging support for ad hoc stub networks," *Computer Communications*, vol. 26, no. 13, pp. 1504–1514, 2003.
- [17] J. Hou, J. Yang, and S. Papavassiliou, "Integration of pricing with call admission control to meet QoS requirements in cellular networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 9, pp. 898–910, 2002.
- [18] P. C. Fishburn and A. M. Odlyzko, "Dynamic behavior of differential pricing and quality of service options for the internet," in *Proceedings of the 1st International Conference on Information and Computation Economies*, pp. 128–139, Charleston, SC, USA, October 1998.
- [19] A. Lundell and T. Westerlund, "Convex underestimation strategies for signomial functions," *Optimization Methods and Software*, vol. 24, no. 4-5, pp. 505–522, 2009.
- [20] A. Lundell, J. Westerlund, and T. Westerlund, "Some transformation techniques with applications in global optimization," *Journal of Global Optimization*, vol. 43, no. 2-3, pp. 391–405, 2009.
- [21] J.-F. Tsai and M.-H. Lin, "Global optimization of signomial mixed-integer nonlinear programming problems with free variables," *Journal of Global Optimization*, vol. 42, no. 1, pp. 39–49, 2008.
- [22] T. Westerlund, "Some transformation techniques in global optimization," in *Global Optimization, from Theory to Implementations*, L. Liberti and N. Maculan, Eds., vol. 84, pp. 45–74, Springer, New York, NY, USA, 2007.
- [23] A. Lundell and T. Westerlund, "On the relationship between power and exponential transformations for positive signomial functions," *Chemical Engineering Transactions*, vol. 17, pp. 1287–1292, 2009.
- [24] R. Pörn, K.-M. Björk, and T. Westerlund, "Global solution of optimization problems with signomial parts," *Discrete Optimization*, vol. 5, no. 1, pp. 108–120, 2008.
- [25] E. L. M. Beale and J. A. Tomlin, "Special facilities in a general mathematical programming system for nonconvex problems using ordered sets of variables," in *Proceedings of 5th International Conference on OR*, pp. 447–454, Venice, Italy, June 1969.
- [26] A. Martin, M. Möller, and S. Moritz, "Mixed integer models for the stationary case of gas network optimization," *Mathematical Programming*, vol. 105, no. 2-3, pp. 563–582, 2006.

- [27] J. P. Vielma and G. L. Nemhauser, "Modeling disjunctive constraints with a logarithmic number of binary variables and constraints," *Mathematical Programming*, vol. 128, no. 1-2, pp. 49–72, 2011.
- [28] J. -F. Tsai and M. -H. Lin, "An efficient global approach for posynomial geometric programming problems," *INFORMS Journal on Computing*, vol. 23, no. 3, pp. 483–492, 2011.
- [29] Lindo, *LINGO, Release 11*, Lindo System, Chicago, Ill, USA, 2004.