

Jianyong Qiao · Xinchao Zhao
Linjiang Pan · Xingquan Zuo
Xingyi Zhang · Qingfu Zhang
Shanguo Huang (Eds.)

Communications in Computer and Information Science

952

Bio-inspired Computing: Theories and Applications

13th International Conference, BIC-TA 2018
Beijing, China, November 2–4, 2018
Proceedings, Part II

Part 2

 Springer

Communications in Computer and Information Science

952

Commenced Publication in 2007

Founding and Former Series Editors:

Phoebe Chen, Alfredo Cuzzocrea, Xiaoyong Du, Orhun Kara, Ting Liu,
Dominik Ślęzak, and Xiaokang Yang

Editorial Board

Simone Diniz Junqueira Barbosa

*Pontifical Catholic University of Rio de Janeiro (PUC-Rio),
Rio de Janeiro, Brazil*

Joaquim Filipe

Polytechnic Institute of Setúbal, Setúbal, Portugal

Igor Kotenko

*St. Petersburg Institute for Informatics and Automation of the Russian
Academy of Sciences, St. Petersburg, Russia*

Krishna M. Sivalingam

Indian Institute of Technology Madras, Chennai, India

Takashi Washio

Osaka University, Osaka, Japan

Junsong Yuan

University at Buffalo, The State University of New York, Buffalo, USA

Lizhu Zhou

Tsinghua University, Beijing, China

More information about this series at <http://www.springer.com/series/7899>

Jianyong Qiao · Xinchao Zhao
Linqiang Pan · Xingquan Zuo
Xingyi Zhang · Qingfu Zhang
Shanguo Huang (Eds.)


Bio-inspired Computing: Theories and Applications


13th International Conference, BIC-TA 2018
Beijing, China, November 2–4, 2018
Proceedings, Part II

Editors

Jianyong Qiao
Beijing University of Posts
and Telecommunications
Beijing
China

Xinchao Zhao 
Beijing University of Posts
and Telecommunications
Beijing
China

Linqiang Pan 
Huazhong University of Science
and Technology
Wuhan
China

Xingquan Zuo 
Beijing University of Posts
and Telecommunications
Beijing
China

Xingyi Zhang
Anhui University
Hefei
China

Qingfu Zhang
City University of Hong Kong
Kowloon
Hong Kong

Shanguo Huang
Beijing University of Posts
and Telecommunications
Beijing
China

ISSN 1865-0929 ISSN 1865-0937 (electronic)
Communications in Computer and Information Science
ISBN 978-981-13-2828-2 ISBN 978-981-13-2829-9 (eBook)
<https://doi.org/10.1007/978-981-13-2829-9>

Library of Congress Control Number: 2018957098

© Springer Nature Singapore Pte Ltd. 2018

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd. The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

Bio-inspired computing is a field of study that abstracts computing ideas (data structures, operations with data, ways to control operations, computing models, etc.) from living phenomena or biological systems such as evolution, cells, tissues, neural networks, immune system, and ant colonies. Bio-Inspired Computing: Theories and Applications (BIC-TA) is a series of conferences that aims to bring together researchers working in the main areas of natural computing inspired from biology, for presenting their recent results, exchanging ideas, and cooperating in a friendly framework.

Since 2006, the conference has taken place at Wuhan (2006), Zhengzhou (2007), Adelaide (2008), Beijing (2009), Liverpool and Changsha (2010), Penang (2011), Gwalior (2012), Huangshan (2013), Wuhan (2014), Hefei (2015), Xi'an (2016), and Harbin (2017). Following the success of previous editions, the 13th International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2018) was organized by Beijing University of Posts and Telecommunications, during November 2–4, 2018.

BIC-TA 2018 attracted a wide spectrum of interesting research papers on various aspects of bio-inspired computing with a diverse range of theories and applications. In all, 89 papers were selected for this volume of *Communications in Computer and Information Science*.

We gratefully thank Beijing University of Posts and Telecommunications, Huazhong University of Science and Technology, the Operation Research Society of China, and the Chinese Society of Optimization and Overall Planning and Economic Mathematics for extensive assistance in organizing the conference. We thank Tingfang Wu, Lianghao Li, Di Zhang, Taosheng Zhang, and Wenting Xu for their help in collecting the final files of the papers and editing the volume. We thank Xing Wan for his contribution in maintaining the website of BIC-TA 2018 (<http://2018.bicta.org/>). Many thanks are given to Hui Tong, Guangzhi Xu, Rui Li, Sai Guo, Min Chen, Jia Liu, Jiaqi Chen, Shuai Feng, and Qing Xiong for their work in organizing the conference. We also thank all the other volunteers, whose efforts ensured the smooth running of the conference.

The editors warmly thank the Program Committee members for their prompt and efficient support in reviewing and handling the papers. The warmest thanks should be given to all the authors for submitting their interesting research work.

Special thanks are due to Springer for their skilled cooperation in the timely production of these volumes.

August 2018

Jianyong Qiao
Xinchao Zhao
Linqiang Pan
Xingquan Zuo
Xingyi Zhang
Qingfu Zhang
Shanguo Huang

Organization

Steering Committee

Guangzhao Cui	Zhengzhou University of Light Industry, China
Kalyanmoy Deb	Indian Institute of Technology Kanpur, India
Miki Hirabayashi	National Institute of Information and Communications Technology (NICT), Japan
Joshua Knowles	University of Manchester, UK
Thom LaBean	North Carolina State University, USA
Jiuyong Li	University of South Australia, Australia
Kenli Li	University of Hunan, China
Giancarlo Mauri	Università di Milano-Bicocca, Italy
Yongli Mi	Hong Kong University of Science and Technology, SAR China
Atulya K. Nagar	Liverpool Hope University, UK
Linqiang Pan	Huazhong University of Science and Technology, China
Gheorghe Păun	Romanian Academy, Bucharest, Romania
Mario J. Pérez-Jiménez	University of Seville, Spain
K. G. Subramanian	Universiti Sains Malaysia, Malaysia
Robinson Thamburaj	Madras Christian College, India
Jin Xu	Peking University, China
Hao Yan	Arizona State University, USA

Program Committee

Muhammad Abulaish	South Asian University, India
Chang Wook Ahn	Gwangju Institute of Science and Technology, Republic of Korea
Adel Al-Jumaily	University of Technology Sydney, Australia
Junfeng Chen	Hohai University, China
Wei-Neng Chen	Sun Yat-Sen University, China
Tsung-Che Chiang	National Taiwan Normal University, China
Shi Cheng	Shaanxi Normal University, China
Bei Dong	Shaanxi Normal University, China
Xin Du	Fujian Normal University, China
Carlos Fernandez-Llatas	Universitat Politècnica de Valencia, Spain
Shangce Gao	University of Toyama, Japan
Wenyin Gong	China University of Geosciences, China
Shivaprasad Gundibail	Manipal Academy of Higher Education, India
Ping Guo	Beijing Normal University, China
Yi-Nan Guo	China University of Mining and Technology, China

Shan He	University of Birmingham, UK
Tzung-Pei Hong	National University of Kaohsiung, China
Florentin Ipate	University of Bucharest, Romania
Sunil Jha	Banaras Hindu University, India
He Jiang	Dalian University of Technology, China
Liangjun Ke	Xi'an Jiaotong University, China
Ashwani Kush	Kurukshetra University, India
Hui Li	Xi'an Jiaotong University, China
Kenli Li	Hunan University, China
Li Li	Guilin University of Electronic Technology, China
Xingmei Li	North China Electric Power University, China
Yangyang Li	Xidian University, China
Qunfeng Liu	Dongguan University of Technology, China
Xiaobo Liu	China University of Geosciences (Wuhan), China
Wenjian Luo	University of Science and Technology of China, China
Lianbo Ma	Northeastern University, China
Wanli Ma	University of Canberra, Australia
Holger Morgenstern	Albstadt-Sigmaringen University, Germany
G. R. S. Murthy	Lendi Institute of Engineering and Technology, India
Akila Muthuramalingam	KPR Institute of Engineering and Technology, India
Yusuke Nojima	Osaka Prefecture University, Japan
Linqiang Pan	Huazhong University of Science and Technology, China
Andrei Paun	University of Bucharest, Romania
Xingguang Peng	Northwestern Polytechnical University, China
Chao Qian	University of Science and Technology of China, China
Rawya Rizk	Port Said University, Egypt
Rajesh Sanghvi	G. H. Patel College of Engineering and Technology, India
Ronghua Shang	Xidian University, China
Ravi Shankar	Florida Atlantic University, USA
Yindong Shen	Huazhong University of Science and Technology, China
Chuan Shi	Beijing University of Posts and Telecommunications, China
Chengyong Si	University of Shanghai for Science and Technology, China
Bosheng Song	Huazhong University of Science and Technology, China
Tao Song	China University of Petroleum, China
Jianyong Sun	University of Nottingham, UK
Shiwei Sun	Chinese Academy of Sciences, China
Yifei Sun	Shaanxi Normal University, China
Gaige Wang	Ocean University of China, China
Feng Wang	Wuhan University, China

Hui Wang	South China Agricultural University, China
Hui Wang	Nanchang Institute of Technology, China
Yong Wang	Central South University, China
Sudhir Warier	IIT Bombay, India
Slawomir T. Wierzchon	Polish Academy of Sciences, Poland
Xiuli Wu	University of Science and Technology Beijing, China
Zhou Wu	Chongqing University, China
Bin Xin	Beijing Institute of Technology, China
Gang Xu	Nanchang University, China
Yingjie Yang	De Montfort University, UK
Zhile Yang	Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, China
Kunjie Yu	Zhengzhou University, China
Defu Zhang	Xiamen University, China
Jie Zhang	Newcastle University, UK
Gexiang Zhang	Southwest Jiaotong University, China
Peng Zhang	Beijing University of Posts and Telecommunications, China
Xingyi Zhang	Anhui University, China
Yong Zhang	China University of Mining and Technology, China
Xinchao Zhao	Beijing University of Posts and Telecommunications, China
Yujun Zheng	Hangzhou Normal University, China
Aimin Zhou	East China Normal University, China
Shang-Ming Zhou	Swansea University, UK
Xinjian Zhuo	Beijing University of Posts and Telecommunications, China
Dexuan Zou	Jiangsu Normal University, China
Xingquan Zuo	Beijing University of Posts and Telecommunications, China

Contents – Part II

Application of Artificial Fish Swarm Algorithm in Vehicle Routing Problem.	1
<i>Shiyu Jia, Kang Zhou, Yu Yang, Huaqing Qi, Yiting Zhen, Long Hu, Zhou Zhang, and Heping Zhang</i>	
Three-Input and Nine-Output Cubic Logical Circuit Based on DNA Strand Displacement	13
<i>Yanfeng Wang, Meng Li, Junwei Sun, and Chun Huang</i>	
A Simulated Annealing for Multi-modal Team Orienteering Problem with Time Windows	23
<i>Yalan Zhou, Chen Li, and Yanyue Li</i>	
Discrete Harmony Search Algorithm for Flexible Job-Shop Scheduling Problems	31
<i>Xiuli Wu and Jing Li</i>	
Barebones Particle Swarm Optimization with a Neighborhood Search Strategy for Feature Selection	42
<i>Chenye Qiu and Xingquan Zuo</i>	
The Chinese Postman Problem Based on the Probe Machine Model	55
<i>Jing Yang, Zhixiang Yin, Jianzhong Cui, Qiang Zhang, and Zhen Tang</i>	
Research on Pulse Classification Based on Multiple Factors	63
<i>Zhihua Chen, An Huang, and Xiaoli Qiang</i>	
Hybrid Invasive Weed Optimization and GA for Multiple Sequence Alignment	72
<i>Chong Gao, Bin Wang, Changjun Zhou, Qiang Zhang, Zhixiang Yin, and Xianwen Fang</i>	
RNA Sequences Similarities Analysis by Cross-Correlation Function.	83
<i>Shanshan Xing, Bin Wang, Xiaopeng Wei, Changjun Zhou, Qiang Zhang, and Zhonglong Zheng</i>	
Refrigerant Capacity Detection of Dehumidifier Based on Time Series and Neural Networks.	95
<i>Gang Peng, Zuhuang Yang, and Min Wang</i>	
An Improved Artificial Bee Colony Algorithm and Its Taguchi Analysis	104
<i>Yudong Ni, Yuanyuan Li, and Yindong Shen</i>	

PLS-Based RBF Network Interpolation for Nonlinear FEM Analysis of Dropped Drum in Offshore Platform Operations	118
<i>Hongwei Liu, Wenjun Zhang, Shuaichen Liu, and Yan Li</i>	
Logic Circuit Design of Sixteen-Input Encoder by DNA Strand Displacement	129
<i>Yanfeng Wang, Aolong Lv, Chun Huang, and Junwei Sun</i>	
PLS-RBF Neural Network for Nonlinear FEM Analysis of Dropped Container in Offshore Platform Operations.	138
<i>Zehua Li, Wenjun Zhang, and Haibo Xie</i>	
A Multiobjective Genetic Algorithm Based Dynamic Bus Vehicle Scheduling Approach.	152
<i>Hongyi Shi, Chunlu Wang, Xingquan Zuo, and Xinchao Zhao</i>	
Research on the Addition, Subtraction, Multiplication and Division Complex Logical Operations Based on the DNA Strand Displacement.	162
<i>Chun Huang, Yanfeng Wang, and Qinglei Zhou</i>	
An Improved GMM-Based Moving Object Detection Method Under Sudden Illumination Change.	178
<i>Jian Cheng, Yusen Gang, Shuai Bai, Yi-nan Guo, and Dongwei Wang</i>	
A Method of Accurately Accepting Tasks for New Workers Incorporating with Capacities and Competition Intensities	188
<i>Dunwei Gong, Chao Peng, Xinchao Zhao, and Qiuzhen Lin</i>	
Iteration-Related Various Learning Particle Swarm Optimization for Quay Crane Scheduling Problem	201
<i>Mingzhu Yu, Xuwen Cong, Ben Niu, and Rong Qu</i>	
An Image Encryption Algorithm Based on Chaotic System Using DNA Sequence Operations	213
<i>Xuncaizhang, Zheng Zhou, Ying Niu, Yanfeng Wang, and Lingfei Wang</i>	
An Image Encryption Algorithm Based on Dynamic DNA Coding and Hyper-chaotic Lorenz System.	226
<i>Guangzhao Cui, Lingfei Wang, Xuncaizhang, and Zheng Zhou</i>	
Application of BFO Based on Path Interaction in Yard Truck Scheduling and Storage Allocation Problem	239
<i>Lei Liu, Lu Xiao, Lulu Zuo, Jia Liu, and Chen Yang</i>	
Research on Optimization of Warehouse Allocation Problem Based on Improved Genetic Algorithm	252
<i>Ding Ning, Wang Li, Teng Wei, and Zhao Yue</i>	

An Expert System for Diagnosis and Treatment of Hypertension Based on Ontology 264
Wang Jie, Peng Yan, Ren Xiaoxiao, and Qiao Yixuan

A Three Input Look-Up-Table Design Based on Memristor-CMOS 275
Junwei Sun, Xingtong Zhao, and Yanfeng Wang

Complex Logic Circuit of Three-Input and Nine-Output by DNA Strand Displacement 287
Yanfeng Wang, Guodong Yuan, Chun Huang, and Junwei Sun

Modified Mixed-Dimension Chaotic Particle Swarm Optimization for Liner Route Planning with Empty Container Repositioning 296
Mingzhu Yu, Zhichuan Chen, Li Chen, Rong Qu, and Ben Niu

A Wrapper Feature Selection Algorithm Based on Brain Storm Optimization 308
Xu-tao Zhang, Yong Zhang, Hai-rong Gao, and Chun-lin He

A Hybrid Model Based on K-EPF and DPIO for UAVs Target Detection . . . 316
Jinsong Chen, Lu Xiao, Jun Wang, Huan Liu, and Qianying Liu

A Hybrid Data Clustering Approach Based on Hydrologic Cycle Optimization and K-means 328
Ben Niu, Huan Liu, Lei Liu, and Hong Wang

A Decomposition Based Multiobjective Evolutionary Algorithm for Dynamic Overlapping Community Detection 338
Xing Wan, Xingquan Zuo, and Feng Song

Research on Public Opinion Communication Mechanism Based on Individual Behavior Model 351
Weidong Huang and Yang Cui

A Comprehensive Evaluation: Water Cycle Algorithm and Its Applications. 360
Rana Muhammad Sohail Jafar, Shuang Geng, Wasim Ahmad, Safdar Hussain, and Hong Wang

A Bias Neural Network Based on Knowledge Distillation 377
Yulong Wang, Zhi Wu, and Yifeng Huang

LSTM Encoder-Decoder with Adversarial Network for Text Generation from Keyword 388
Dongju Park and Chang Wook Ahn

Quantum Algorithm for Crowding Method. 397
Jun Suk Kim and Chang Wook Ahn

Random Repeatable Network: Unsupervised Learning to Detect Interest Point 405
Pei Yan and Yihua Tan

An Orthogonal Genetic Algorithm with Multi-parent Multi-point Crossover for Knapsack Problem 415
Xinchao Zhao, Jiaqi Chen, Rui Li, Dunwei Gong, and Xingmei Li

Cooperative Co-evolution with Principal Component Analysis for Large Scale Optimization 426
Guangzhi Xu, Xinchao Zhao, and Rui Li

HCO-Based RFID Network Planning. 435
Jun Wang, Jinsong Chen, Qianying Liu, and Jia Liu

Cuckoo Search Algorithm Based on Individual Knowledge Learning. 446
Juan Li, Yuan-Xiang Li, and Jie Zou

An Improved DV-Hop Algorithm with Jaccard Coefficient Based on Optimization of Distance Correction 457
Wangsheng Fang, Geng Yang, and Zhongdong Hu

An Image Encryption Algorithm Based on Hyper-chaotic System and Genetic Algorithm. 466
Xuncai Zhang, Hangyu Zhou, Zheng Zhou, Lingfei Wang, and Chao Li

A Performance Comparison of Crossover Variations in Differential Evolution for Training Multi-layer Perceptron Neural Networks 477
Tae Jong Choi, Yun-Gyung Cheong, and Chang Wook Ahn

Author Index 489

Contents – Part I

Research on Price Forecasting Method of China’s Carbon Trading Market Based on PSO-RBF Algorithm	1
<i>Yuansheng Huang and Hui Liu</i>	
An Efficient Restart-Enhanced Genetic Algorithm for the Coalition Formation Problem	12
<i>Miao Guo, Bin Xin, Jie Chen, and Yipeng Wang</i>	
U-NSGA-III: An Improved Evolutionary Many-Objective Optimization Algorithm	24
<i>Rui Ding, Hongbin Dong, Jun He, Xianbin Feng, Xiaodong Yu, and Lijie Li</i>	
Elman Neural Network Optimized by Firefly Algorithm for Forecasting China’s Carbon Dioxide Emissions	36
<i>Yuansheng Huang and Lei Shen</i>	
Research on “Near-Zero Emission” Technological Innovation Diffusion Based on Co-evolutionary Game Approach	48
<i>Yuansheng Huang, Hongwei Wang, and Shijian Liu</i>	
Improved Clonal Selection Algorithm for Solving AVO Elastic Parameter Inversion Problem	60
<i>Zheng Li, Xuesong Yan, Yuanyuan Fan, and Ke Tang</i>	
A Pests Image Classification Method Based on Improved Wolf Pack Algorithm to Optimize Bayesian Network Structure Learning	70
<i>Lin Mei, Shengsheng Wang, and Jie Liu</i>	
Differential Grouping in Cooperative Co-evolution for Large-Scale Global Optimization: The Experimental Study	82
<i>Heng Lei, Ming Yang, and Jing Guan</i>	
Spiking Neural P Systems with Anti-spikes Based on the Min-Sequentiality Strategy	94
<i>Li Li and Keqin Jiang</i>	
Solving NP Hard Problems in the Framework of Gene Assembly in Ciliates	107
<i>Ganbat Ganbaatar, Khuder Altangerel, and Tseren-Onolt Ishdorj</i>	

A Study of Industrial Structure Optimization Under Economy, Employment and Environment Constraints Based on MOEA. 120
Ruozhu Zhang

DNA Strand Displacement Based on Nicking Enzyme for DNA Logic Circuits. 133
Gaiying Wang, Zhiyu Wang, Xiaoshan Yan, and Xiangrong Liu

Motor Imaginary EEG Signals Classification Based on Deep Learning 142
Haoran Wang and Wanying Mo

DNA Origami Based Computing Model for the Satisfiability Problem. 151
Zhenqin Yang, Zhixiang Yin, Jianzhong Cui, and Jing Yang

DNA 3D Self-assembly Algorithmic Model to Solve Maximum Clique Problem. 161
Jingjing Ma and Wenbin Gao

Industrial Air Pollution Prediction Using Deep Neural Network 173
Yu Pengfei, He Juanjuan, Liu Xiaoming, and Zhang Kai

An Efficient Genetic Algorithm for Solving Constraint Shortest Path Problem Through Specified Vertices. 186
Zhang Kai, Shao Yunfeng, Zhang Zhaozong, and Hu Wei

An Attribute Reduction P System Based on Rough Set Theory. 198
Ping Guo and Junqi Xiang

Spatial-Temporal Analysis of Traffic Load Based on User Activity Characteristics in Mobile Cellular Network. 213
Moqin Zhou, Xueli Wang, Xing Zhang, and Wenbo Wang

A Simulator for Cell-Like P System 223
Ping Guo, Changsheng Quan, and Lian Ye

Dynamic Multimodal Optimization Using Brain Storm Optimization Algorithms 236
Shi Cheng, Hui Lu, Wu Song, Junfeng Chen, and Yuhui Shi

A Hybrid Replacement Strategy for MOEA/D 246
Xiaoji Chen, Chuan Shi, Aimin Zhou, Siyong Xu, and Bin Wu

A Flexible Memristor-Based Neural Network 263
Junwei Sun, Gaoyong Han, and Yanfeng Wang

A Biogeography-Based Memetic Algorithm for Job-Shop Scheduling 273
Xue-Qin Lu, Yi-Chen Du, Xu-Hua Yang, and Yu-Jun Zheng

Analysing Parameters Leading to Chaotic Dynamics in a Novel Chaotic System.	285
<i>Junwei Sun, Nan Li, and Yanfeng Wang</i>	
Enhanced Biogeography-Based Optimization for Flow-Shop Scheduling	295
<i>Yi-Chen Du, Min-Xia Zhang, Ci-Yun Cai, and Yu-Jun Zheng</i>	
A Weighted Bagging LightGBM Model for Potential lncRNA-Disease Association Identification	307
<i>Xin Chen and Xiangrong Liu</i>	
DroidGene: Detecting Android Malware Using Its Malicious Gene	315
<i>Yulong Wang and Hua Zong</i>	
Visualize and Compress Single Logo Recognition Neural Network	331
<i>Yulong Wang and Haoxin Zhang</i>	
Water Wave Optimization for Artificial Neural Network Parameter and Structure Optimization	343
<i>Xiao-Han Zhou, Zhi-Ge Xu, Min-Xia Zhang, and Yu-Jun Zheng</i>	
Adaptive Recombination Operator Selection in Push and Pull Search for Solving Constrained Single-Objective Optimization Problems	355
<i>Zhun Fan, Zhaojun Wang, Yi Fang, Wenji Li, Yutong Yuan, and Xinchao Bian</i>	
DeepPort: Detect Low Speed Port Scan Using Convolutional Neural Network	368
<i>Yulong Wang and Jiuchao Zhang</i>	
A Dual-Population-Based Local Search for Solving Multiobjective Traveling Salesman Problem	380
<i>Mi Hu, Xinye Cai, and Zhun Fan</i>	
A Cone Decomposition Many-Objective Evolutionary Algorithm with Adaptive Direction Penalized Distance	389
<i>Weiqin Ying, Yali Deng, Yu Wu, Yuehong Xie, Zhenyu Wang, and Zhiyi Lin</i>	
Origin Illusion, Elitist Selection and Contraction Guidance.	401
<i>Rui Li, Guangzhi Xu, Xinchao Zhao, and Dunwei Gong</i>	
A Multi Ant System Based Hybrid Heuristic Algorithm for Vehicle Routing Problem with Service Time Customization	411
<i>Yuan Wang and Lining Xing</i>	
Model Predictive Control of Data Center Temperature Based on CFD	423
<i>Gang Peng, Chenyang Zhou, and Siming Wang</i>	

Computer System for Designing Musical Expressiveness in an Automatic Music Composition Process	434
<i>Michele Della Ventura</i>	
A Hybrid Dynamic Population Genetic Algorithm for Multi-satellite and Multi-station Mission Planning System	444
<i>Yan-Jie Song, Xin Ma, Zhong-Shan Zhang, Li-Ning Xing, and Ying-Wu Chen</i>	
An 8 to 3 Priority Encoder Based on DNA Strand Displacement	454
<i>Mingliang Wang and Bo Bi</i>	
Multifunctional Biosensor Logic Gates Based on Graphene Oxide.	473
<i>Luhui Wang, Yingying Zhang, Yani Wei, and Yafei Dong</i>	
Medium and Long-Term Forecasting Method of China’s Power Load Based on SaDE-SVM Algorithm.	484
<i>Yuansheng Huang, Lijun Zhang, Mengshu Shi, Shijian Liu, and Siyuan Xu</i>	
Coupling PSO-GPR Based Medium and Long Term Load Forecasting in Beijing.	496
<i>Yuansheng Huang, Jianjun Hu, Yaqian Cai, and Lei Yang</i>	
Nonlinear Finite-Element Analysis of Offshore Platform Impact Load Based on Two-Stage PLS-RBF Neural Network	508
<i>Shibo Zhou and Wenjun Zhang</i>	
Author Index	519



Application of Artificial Fish Swarm Algorithm in Vehicle Routing Problem

Shiyu Jia¹, Kang Zhou¹(✉), Yu Yang¹, Huaqing Qi², Yiting Zhen¹,
Long Hu¹, Zhou Zhang¹, and Heping Zhang¹

¹ School of Math and Computer, Wuhan Polytechnic University,
Wuhan 430023, Hubei, China
zhoukang_wh@163.com

² Department of Economics and Management, Wuhan Polytechnic University,
Wuhan 430023, Hubei, China

Abstract. Artificial fish swarm algorithm (AFSA) has important theoretical research value and practical significance in solving VRP. The traditional AFSA which does not consider the structural features of VRP will lead to too complex for the process to solve problems, too much time to search optimal solution and too low computational accuracy. In this paper, the traditional method is improved that neighborhood search which are more efficient for VRP are used in the three behaviors of AF swarm, and discretize the three behaviors. The improvement optimizes the behavior of finding optimal solution of AFSA in VRP, and avoids the convergence rate becoming too fast in later stage and falling into the local optimal while expanding the search. Through the experimental comparative analysis, the improved method is more effective and feasible than traditional method.

Keywords: VRP · Artificial fish swarm algorithm (AFSA) · Improved AFSA

1 Introduction

VRP is an NP-hard problem, which is also a hot issue in the field of logistics [1]. With the continuous development of the Internet, people change their way of consumption, logistics and distribution activities become more and more frequent. Considering the different needs of the logistics distribution process, the purpose of reducing transportation costs and transportation time can be achieved by planning the route of the vehicle and the number of vehicles, and making full use of distribution resources to meet the different result of objectives. Now, in the problem of post office delivery arrangements, bus route arrangements, power dispatch issues, express mail delivery, setting of Aviation and railway schedules, collection of waste and many other problems in real life can be abstractly mapped to VRP [2]. Obviously, the application of VRP in our lives is important.

With the deep exploration of VRP, some new intelligent algorithms have been put forward, including artificial fish swarm algorithm (AFSA), genetic algorithm, particle swarm algorithm, ant colony algorithm. The latter will take long time in searching and have low accuracy [3]. AFSA is better in global convergence. The sensitivity of the

selection of parameters is lesser, and it is easier to realize. Therefore, the AFSA is used to find the optimal solution for VRP in this paper.

And now AFSA has been applied in different fields. In the online identification of time-varying systems, parameter optimization of robust PDI and optimization of forward neural networks, which is shown that AFSA is robust and simple and easy to achieve [4], in the forex forecast and portfolio, AFSA can get the higher accuracy of the forecasting exchange rate and the expected rate of return [5]. In the swarming diagnosis of mechanical fault, the fault diagnosis model is established by establishing the fault swarming diagnosis [6]. In the problem of vehicle congestion dispatch, it can overcome the shortcomings of convergence speed and improve the accurate selection of optimal path [7]. The AFSA has a short convergence time and can approach the extreme point quickly. However, at later time of the algorithm, the diversity of the fish population is easy to fall into the local extreme point [8]. Traditional method is improved in this paper, by using a unique artificial fish field of view and artificial fish preying, swarming, following behavior and multiple goals to quickly close to the optimal solution.

2 VRP and AFSA

2.1 VRP and Its Mathematical Model

Logistics distribution is a distribution activity which includes the material storage, sorting, assembly, transportation, and the needs of customers will be satisfied by sending the goods to the designated location. The research on VRP can effectively reduce transportation costs and achieve rational allocation of effective resources. The main research methods of VRP include precise optimization algorithm, heuristic optimization algorithm and bionic optimization algorithm [9]. The VRP in logistics delivery can be described as follows: According to the different needs of customers, distribution center will design a reasonable route and return to the starting point of the process (under some certain restrictions, such as the length of the path, the number of vehicles, transport time, etc.).

In order to describe the mathematical model of VRP, this paper uses d_{ij} to denote the distance from customer point i to customer point j . Q represents the maximum load of the car. q_i represents the demand of customer i . x_{ijk} is used to insure whether vehicle k has driven from point i to point j or not, if it is yes, set x_{ijk} to be 1, else to be 0. The mathematical model can be defined as follows:

$$\min f_1 = \sum_{k=1}^K \sum_{i=0}^n \sum_{j=0}^n d_{ij} x_{ijk} \quad (1)$$

$$\sum_{i=1}^n q_i y_{ik} \leq Q; \forall k \in I \quad (2)$$

$$\sum_{i=0}^n x_{ijk} = y_{jk}; j = 1, 2, \dots, n; \forall k \in I \quad (3)$$

$$\sum_{i=0}^n x_{ijk} = y_{ijk}; i = 1, 2, \dots, n; \forall k \in I \quad (4)$$

$$\sum_{k=1}^K y_{ik} = \begin{cases} 1 (i = 1, 2, \dots, n; \forall k \in I) \\ k (i = 0) \end{cases} \quad (5)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} \leq |S| - 1; \forall S (2 \leq |S| \leq n - 1) \subseteq V/0 \quad (6)$$

Equation (1) is the objective function of mathematical model, formula (2) means that the carrying capacity can not exceed the maximum load, formula (3) and (4) means each customer point can only be served by one car, formula (5) and (6) means the route of each car starts from warehouse 0 and goes back to warehouse 0 when the service is completed.

2.2 Three Behaviors of the Traditional Artificial Fish Algorithm

There are three steps of the calculation process for AFSA:

Step1. Initialize the population.

Step2. Find the optimal solutions by some behaviors like following, swarming and preying.

Step3. Update bulletin board.

Repeat above steps until the solution can meet the condition.

Set the current state of artificial fish as $X_i^{(t)}$, the implementations of the three behaviors are as following:

(1) Following behavior:

Calculating the artificial fish set within the vision field of the i AF (artificial fish).

$$S_i = \left\{ X_j^{(t)} \mid \left\| X_j^{(t)} - X_i^{(t)} \right\| \leq \text{Visual}, j \neq i \right\}, \text{ and } N_f = |S_i|.$$

If $N_f \neq 0$, then calculate the position $X_m^{(t)}$ with the maximum concentration of food in the perceived range.

If $Y_m^{(t)} < Y_i^{(t)}$ and $N_f \times Y_m^{(t)} < \delta \times Y_i^{(t)}$ ($\delta > 1$), then take the next movement toward the direction of this position and reach the next state $X_i^{(t+1)}$:

$$x_{ij}^{(t+1)} = x_{ij}^{(t)} + \text{rand}() \times \text{Step} \times \left(x_{cj}^{(t)} - x_{ij}^{(t)} \right) / \left\| X_c^{(t)} - X_i^{(t)} \right\| (j = 1, 2, \dots, D) \quad (7)$$

Finished, then implement preying behavior.

(2) Swarming behavior:

Calculate set S_i and $N_f = |S_i|$.

If $N_f \neq 0$, then calculate the central position $X_c^{(t)}$,

$$X_c^{(t)} = \sum (X_j^{(t)} | j = 1, 2, \dots, N_f, X_j^{(t)} \in S_i) / N_f.$$

If $Y_c^{(t)} < Y_i^{(t)}$ and $N_f \times Y_c(t) < \delta \times Y_i(t)$ ($\delta > 1$), then take the next movement toward the central position and reach the next state $X_{ij}^{(t+1)}$:

$$x_{ij}^{(t+1)} = x_{ij}^{(t)} + rand() \times Step \times (x_{cj}^{(t)} - x_{ij}^{(t)}) / \|X_c^{(t)} - X_i^{(t)}\| (j = 1, 2, \dots, D) \quad (8)$$

Finished, then implement preying behavior.

(3) Preying behavior:

AF select a state $X_v(t)$ by neighborhood search in the perceived range:

$$X_{vj} = x_{ij}^{(t)} + rand() \times Visual \quad (9)$$

If the state is better, then move to the new state $X_i(t+1)$:

$$x_{ij}^{(t+1)} = x_{ij}^{(t)} + rand() \times Step \times (x_{vj}^{(t)} - x_{ij}^{(t)}) / \|X_v^{(t)} - X_i^{(t)}\| (j = 1, 2, \dots, D) \quad (10)$$

If the optimal solution can not be found after

Iterating *Try_number* times, the stochastic searching will be implemented, the mathematical formula is as follows:

$$x_{ij}^{(t+1)} = x_{ij}^{(t)} + rand() \times Step. \quad (11)$$

3 The Design of AFSA for VRP

3.1 Encoding Selection

This paper chooses one-dimensional coding. In other words, if there are n customer points currently, then the AF's position code will be replaced by a series of ordered arrangements from 1 to n . Each arrangement represents a AF's position and the individual number in the arrangement represents a customer point.

How to decode? Since every customer point has corresponding demand, we would make judgment on the demand of all the customer points of position encodes in sequence by the vehicle's load limit. If the demand does not exceed the limit, this point will be served. Else, number 0 will be inserted in front of the customer point, which means another vehicle has been used for transportation.

As it is shown in Table 1, the position code array is 3, 2, 5, 1, 4; the corresponding demands are 1.5, 1.3, 0.9, 1.6 and 1.6; and the load capacity is 3.0.

Table 1. Position code, load capacity and corresponding demand

Position code	Corresponding demand	Load capacity
3	1.5	3.0
2	1.3	3.0
5	0.9	3.0
1	1.6	3.0
4	1.6	3.0

The first vehicle departs from the warehouse, when it has passed the point 3 as well as point 2, and the customer point 5 will be served. At the moment, the load volume has reached 3.7, exceeding the maximum capacity 3.0, so the first vehicle should return the warehouse after serving the point 2 and 3. The first decoding array is 0, 3, 2, 0. Similarly, when the second vehicle departs from the warehouse, it can not serve the point 4 after passing the point 5 and 1. Therefore, the second decoding array is 0, 5, 1, 0. Finally the third vehicle departs from warehouse, and it just serve the point 4. So the third decoding array is 0, 4, 0.

3.2 Design of Neighborhood Search

Neighborhood search will search in a neighborhood of the solution space. It has been proved that properly defining neighborhood and search action play an important part in the quality of the solution and the length of the operation time. The key to design neighborhood search is the diversity of search action which determines the diversity of the neighborhood search. So the four search methods have been selected as following.

Redesign of the Neighborhood Search

$\text{Rand}() \times \text{Step}$ in formulas from (7) to (11) denotes the displacement steps. $(x_{c_j}^{(t)} - x_{ij}^{(t)}) / \left\| X_c^{(t)} - X_i^{(t)} \right\|$ denotes a way for displacement. Since these formulas are just suitable for the continuous displacement operations, how to change the meaning of the formulas into solve discrete VRP becomes a very important issue.

From the above formulas, we can recognize their common character is that they all regard $X_i^{(T)}$ as the center. And we also find that each neighborhood search is made to approach the target point by means of displacement and the steps of displacement.

According to the current references, the most optimal way is combining the several neighborhood search and then choose one way from them. Take a random number multiplied by the corresponding variable and getting roundness as the step size. In fact, from generation t to generation $t + 1$ are gotten by taking $\text{rand}() \times \text{Step}$ times neighborhood search and iterating repeatedly.

Set X as the N-dimensional coding of VRP. The following four kinds of neighborhood search methods for encoding X are introduced:

(1) Random exchange:

Arbitrarily exchange two different numbers. The operation of the random exchange is designed as follows:

RandomSwap(X)

Step1 Produce two positive integers i, j randomly, satisfied $1 \leq i, j \leq n$ and $i < j$.

Step2 Set $Y(i) = X(j)$, $Y(j) = X(i)$, $Y(k) = X(k)$ ($k \neq i, j$).

Step3 Return(Y).

(2) Insertion:

Insert a number into the front of another number. The insertion switching operation is designed as follows:

Insert(X)

Step1 Produce two positive integers i, j , satisfied $1 \leq i, j \leq n$ and $i < j$.

Step2 Set $Y(i) = X(j)$, $Y(k) = X(k - 1)$ ($k > i$), $Y(q) = X(q)$ ($q < i$).

Step3 Return(Y).

(3) Reversal:

Reverse all the numbers in the two certain positions of the array. The flip switching operation is designed as follows:

Turnover(X)

Step1 Produce two positive integers i, j , satisfied $1 \leq i, j \leq n$ and $i < j$.

Step2 $Y(i + k) = X(j - k)$ ($k = 0, 1, \dots, [(j - i)/2]$), $Y(k) = X(k)$ ($k < i$ or $k > j$).

Step3 Return(Y).

(4) Adjacent exchange:

Exchange the two adjacent numbers. The adjacent exchange operation are designed as follows:

AdjacentExchange(X)

Step1 Produce a positive integer i , satisfied $1 \leq i \leq n - 1$.

Step2 $Y^{(i)} = X^{(i+1)}$, $Y^{(i+1)} = X^{(i)}$, $Y^{(k)} = X^{(k)}$ ($k \neq i, i + 1$).

Step3 Return(Y).

Set X as the N -dimensional encoding of VRP, the redesigned method of the neighborhood search is that select one from four operations: RandomSwap, Insert, Turnover and AdjacentExchange. Then implement neighborhood search again. A new neighborhood search method is designed as follows:

VNS(X)

Step1 Produce a positive integer i , satisfied $1 \leq i \leq 4$.

Step2 Choose one from four operations: RandomSwap, Insert, Turnover and AdjacentExchange. And then implement the neighborhood search:

Switch(i)

Case 1: $Y = \text{RandomSwap}(X)$

Case 2: $Y = \text{Insert}(X)$

Case 3: $Y = \text{Turnover}(X)$

Case 4: $Y = \text{AdjacentExchange}(X)$

Step3 Return(Y).

A Design Based on Neighborhood Search of Three Operations for AF

In the traditional AFSA, formulas from (7) to (11) are just suitable for solving continuity problems, suitable for VRP. It is necessary to discretize the three behaviors. Because the basic operation of the three behaviors is neighborhood search, a new neighborhood search is designed for VRP and discretizes the three behaviors. For the fish i of state $X_i^{(t)}$, the process of discretization for the three behaviors is designed as follows:

(1) Following behavior is designed as follows:

Step 1 Calculate the set $S_i : S_i = \{X_k^{(t)} \mid |\rho(X_k^{(t)}, X_i^{(t)})| < \text{visual}\}$; and $N_f = |S_i|$.

Step 2 If $N_f \neq 0$, then calculate the position encoding $X_m(t)$ with the biggest food concentration: $Y_m^{(t)} = \max\{Y_j^{(t)} \mid X_j^{(t)} \in S_i, i \neq j\}$.

Step 3 The following behaviors implement $[\text{rand}() \times \text{Step}]$ times:

If $Y_m^{(t)} < Y_i^{(t)}$ and $N_f \times Y_m^{(t)} < \delta \times Y_i^{(t)}$,

then $X^{(t+1)} = \text{VNS}(X_i^{(t)})$.

If $Y^{(t+1)} < Y_i^{(t)}$, then $X_i^{(t+1)} = X^{(t+1)}$,

else produce random number r ($0 \leq r \leq 1$).

If $r < 30\%$, then $X_i^{(t+1)} = X^{(t+1)}$, else $X_i^{(t+1)} = X_m^{(t)}$,

else $X_i^{(t+1)} = X_i^{(t)}$, jump to Step 4.

Step 4 Implement swarming behavior.

(2) Swarming behavior is designed as follows:

Step 1 Calculate the set $S_i : S_i = \{X_k^{(t)} \mid |\rho(X_k^{(t)}, X_i^{(t)})| < \text{visual}\}$; and $N_f = |S_i|$.

Step 2 If $N_f \neq 0$, then calculate center position $X_c^{(t)}$:

$$X_c^{(t)} = \sum (X_j^{(t)} \mid j = 1, 2, \dots, N_f, X_j^{(t)} \in S_i).$$

Step 3 Renew $X_c^{(t)}$, retain only the position code that appears for the first time, put the nonexistent codes into duplicate position randomly.

Step 4 The following behaviors implement $[\text{rand}() \times \text{Step}]$ times:

If $Y_c^{(t)} < Y_i^{(t)}$ and $N_f \times Y_c^{(t)} < \delta Y_i^{(t)}$,

Then $X^{(t+1)} = \text{VNS}(X_i^{(t)})$.

If $Y^{(t+1)} < Y_i^{(t)}$, then $X_i^{(t+1)} = X^{(t+1)}$.

Else produce random number r ($0 \leq r \leq 1$).

If $r < 30\%$, then $X_i^{(t+1)} = X^{(t+1)}$.

Else $X_i^{(t+1)} = X_c^{(t)}$.

Else $X_i^{(t+1)} = X_i^{(t)}$.

Jump to Step 5.

Step 5 Implement preying behavior.

(3) Preying behavior is designed as follows:

Step 1 Set attempts: $k = 1$.

Step 2 Regard $X_i^{(t)}$ as initial state, call the operation VNS , implement $[\text{rand}() \times \text{visual}]$ times neighborhood search, and then the next state is $Y_v^{(t)}$.

Step 3 If $Y_v^{(t)} < Y_i^{(t)}$, then implement neighborhood search for $[\text{rand}() \times \text{Step}]$ times by VNS operations. Then $X_i^{(t+1)}$ is the new state.

Step 4 If $k = \text{Try_number}$, then jump to Step 5; otherwise, set $k = k + 1$, and jump to Step 2.

Step 5 Make RandomSwap operation to state $X_i^{(t)}$ for D times (D is the number of customer points), and get the new state $X_D^{(t)}$, if state $X_D^{(t)}$ is better than the last state, then $X_i(t+1) = X_D(t)$; otherwise, implement Turnover neighborhood search for one time to the optimal solution in the update bulletin board and get the new state $X_{\text{final}}(t)$, $X_i^{(t+1)} = X_{\text{final}}^{(t)}$.

$\text{Rand}()$ is random number, ranging from 0 to 1, visual vision field, step is the length of a step. Try_number represents the number of attempt. $\text{VNS}(X)$ represents self-designed neighborhood search.

3.3 AFSA Based on VRP Discretization

Based on the traditional AFSA, this paper adds four neighborhood searches and redefines the distance to design a complete discretized AFSA. The improved algorithm is as follows:

Step 1 (Initialization)

Determine the size of artificial fish swarm N , Within the variable feasible domain, generated N individual fish randomly, set $\text{Visual} = m$, $\text{Step} = 1$, $\delta(\delta > 1)$, $\text{Try_number} = \text{Tn}$.

Step 2 (Update bulletin board $X_c^{(t)}$)

The status of the fish i is $X_i^{(t)}$, the status of the fish j is $X_j^{(t)}$.

If $Y_i^{(t)} < Y_j^{(t)} (i \neq j)$, then $X_c^{(t)} = X_i^{(t)}$.

Else $X_c^{(t)} = X_j^{(t)}$.

Step 3 (Behavior choice)

Each AF imitate improved following and swarming behavior, and the default behavior is preying behavior.

Step 4 (Update bulletin board $X_c^{(t)}$)

The status of the fish i is $X_i^{(t)}$

If $Y_i^{(t)} < Y_c^{(t)}$, then $X_c^{(t)} = X_i^{(t)}$.

Else $X_c^{(t)} = X_c^{(t)}$.

Step 5 (Termination judgment)

If $Try_number = Tn$, then end, output $X_c^{(t)}$.

Else turn back to Step 3.

4 Simulation Experiment

4.1 Experimental Environment

Experiment software: VS2013

Programing language: C

Processor: AMD A6-3420 M APU with Radeon(tm) HD Graphics 1.50 GHz and Genuine Intel(R) CPU T2080 @1.73 GHz 1.73 GHz

Memory: 4.00 GB (2.74 GB available)

Operating system: Win7 32 bit operating system

4.2 Experimental Analysis and Setting of Algorithm Parameters

The maximum visual field in the traditional AFSA is *visualmin*, while Minimum visual field is *visualmax*, step length is *step*, degree of congestion is *delta*, there is no need to set different values for each problem. In other words, these parameters can be used to solve different problems with a fixed value. Like the size population, (*foodnumber*) and the maximum frequency of preying behavior (*try_number*), these parameters can be changed according to different problems. The settings of fixed parameter in the algorithm are shown in Table 2.

Table 2. Settings of fixed parameter

Visualmin	Visualmax	Step	Delta
2	4	2	4

In the algorithm, the parameters that need to be changed according to different problems are as follows: Population size (*foodnumber*) and the maximum frequency of preying behavior (*try_number*). The results from the tests are listed below to demonstrate the impact of the selection of these parameter values on the algorithm.

Foodnumber stands for population size. If foodnumber is too large, it is unfavorable to the convergence of population, and will also increase the algorithm time. So, according to the experiment, population size is set according to the number of customer points. If the maximum frequency of preying behavior (try_number) is too small, there will be a lot of missing, and is unfavorable to optimization. If it is too large, a lot of situations will happen again and waste time. After the experiment, setting try_number to be 50 is the most suitable.

The settings of variable parameter in the algorithm are shown in Table 3.

Table 3. The settings of variable parameter

VRP instance	Tradition		Improved		Known optimal solution
	Optimal value	Iteration number	Optimal value	Iteration number	
A-n32-k5.vrp	1261	5000	797	5000	784
A-n44-k6.vrp	1545	5000	943	5000	937
A-n45-k7.vrp	1872	5000	1155	5000	1146
A-n53-k7.vrp	1642	5000	1028	5000	1010
B-n57-k9.vrp	2393	5000	1636	5000	1598
B-n67-k10.vrp	1787	5000	1057	5000	1032
B-n68-k9.vrp	1763	5000	1288	5000	1272
B-n78-k10.vrp	2256	8000	1273	8000	1221

4.3 Comparison and Analysis of Experimental Results

The Comparison Between Traditional Algorithm and Improved Algorithm

Several instances from the VRP database are taken out for experimental analysis. See Table 4 for details.

Table 4. The result of comparison between traditional algorithm and improved algorithm

Foodnumber	Try_number
The amount of customer point	50

Result analysis: As it can be seen from Table 4, the improved AFSA is superior to the traditional AFSA in solving VRP. The improved AFSA can easily find the optimal solution. With the increase in the number of customer sites, the advantage of the improved optimization is more obvious and close to the known optimal solution of the VRP database. Far exceed the traditional AFSA. According to the above results analysis, we can draw the following conclusions. The improved AFSA has better precision and convergence effect. The improved algorithm can also avoid the local optimization and enlarge the search range. It can be seen that the improved method is more effective and feasible than the traditional method.

Comparison of AFSA with Other Algorithms

We select multiple instances from the VRP standard database web site, the improved AFSA is compared with some mainstream artificial intelligence algorithms that deal with the VRP. See Table 5 for details.

Table 5. Comparison of AFSA with other algorithms

VRP instance	TS [27]	PSO [21]	ACO [19, 22]	GA [13, 14, 17, 18, 20]	Improved algorithm	Known optimal solution
A-n53-k7.vrp	1412	2374	1231	1357	1028	1010
B-n67-k10.vrp	1487	2453	1284	1365	1057	1032
B-n68-k9.vrp	1544	2678	1401	1523	1288	1272
B-n57-k9.vrp	2019	3017	1824	1987	1636	1598
F-n72-k4.vrp	578	768	288	564	244	237

Result analysis: It can be seen from Table 5 that the improved AFSA is better than the other four algorithms, and the improved AFSA is close to the known optimal solution.

From the above results analysis, we can draw the following conclusions: Compared with other algorithms in the table, the improved AFSA has better ability to find the optimal solution and has higher accuracy. The improved AFSA is more competitive than other algorithms.

The Comparison Between Improved Algorithm and the Known Data on Database Website

In order to further test the performance of the improved AFSA proposed in this paper, we will use multiple VRP examples to test the data. See Table 6 for details.

Table 6. Experiment results of improved algorithm

VRP instance	City size	Known optimal solution	This article best solution	Error rate	Average value
A-n32-k5.vrp	32	784	797	1.65%	814
A-n44-k6.vrp	44	937	943	0.64%	961
A-n45-k7.vrp	45	1146	1155	0.78%	1187
A-n53-k7.vrp	53	1010	1028	1.78%	1071
B-n57-k9.vrp	57	1598	1636	2.37%	1652
P-n60-k10.vrp	60	744	757	1.71%	779
B-n68-k9.vrp	68	1272	1288	1.25%	1328
P-n70-k10.vrp	70	827	846	2.29%	872
F-n72-k4.vrp	72	237	244	2.95%	253
B-n78-k10.vrp	78	1221	1273	4.25%	1311
G-n262-k25.vrp	262	6119	6409	4.73%	6886

Result analysis: As it can be seen from Table 6, the improved AFSA is close to the known optimal solution when the customer points are small, and the error rate can be controlled below 1%. With the number of customers increasing, the error rate increases, but still within the acceptable range. From the average value, it can be seen that the improved AFSA has better stability and stronger robustness.

5 Summary

In this paper, the definition of the artificial fish's visual field, the improvement of the rear end, the gathering and the preying behavior of the artificial fish have been perfected. Adding part of the bulletin board in the preying behavior accelerates the convergence speed and avoids falling into the local optimum. It is proved by simulation experiment that the improved method can get better solution under the same conditions. These provide a better quality for the logistics and transport which make the transport more efficient.

Acknowledgements. The work was supported by the Special Scientific Research Fund of Food Public Welfare Profession of China (201513004-3), subproject of the National Key Research and Development Program of China (2017YFD0401102-02), the Guiding Scientific Research Project of Hubei Provincial Education Department (B2017078) and the Humanities and Social Sciences Fund Project of Hubei Provincial Education Department (17Y071).

References

1. Chen, P., Huang, H.K., Dong, X.Y.: A hybrid heuristic algorithm for the vehicle routing problem with simultaneous delivery and pickup. *Chin. J. Comput.* **31**(4), 565–573 (2008)
2. Alegre, J., Laguna, M., Pacheco, J.: Optimizing the periodic pick-up of raw materials for a manufacturer of auto parts. *Eur. J. Oper. Res.* **179**(3), 736–746 (2007)
3. Kim, G., Ong, Y.S., Heng, C.K., Tan, P.S., Zhang, N.A.: City vehicle routing problem (city VRP): a review. *IEEE Trans. Intell. Transp. Syst.* **16**(4), 1654–1666 (2015)
4. He, Y., Wen, J., Huang, M.: Study on emergency relief VRP based on clustering and PSO. In: 11th International Conference on Computational Intelligence and Security (CIS), pp. 43–47. IEEE (2015)
5. Ma, J., Tan, X.Z., Xu, W.X.: Study on VRP based on improved ant colony optimization and internet of vehicles. In: 2014 IEEE Conference and Expo Transportation Electrification Asia-Pacific (ITEC Asia-Pacific), pp. 1–6. IEEE (2014)
6. Garcie, J., Berlanga, A., Lopez, J.M.M.: Effective evolutionary algorithms for many-specifications attainment: application to air traffic control tracking filters. *IEEE Trans. Evol. Comput.* **13**(1), 151–168 (2009)
7. Hou, E.S.H., Ansari, N., Ren, H.: Genetic algorithm for multiprocessor scheduling. *IEEE Trans. Parallel Distrib. Syst.* **5**(2), 113–120 (1994)
8. Li, N., Zou, T., Sun, D.B.: Particle swarm optimization for vehicle routing problem. *J. Syst. Eng.* **19**(6), 596–600 (2004)
9. Ma, X.M., Liu, N.: Improved artificial fish-swarm algorithm based on adaptive vision for solving the shortest path problem. *J. Commun.* **35**(01), 1–6 (2014)



Three-Input and Nine-Output Cubic Logical Circuit Based on DNA Strand Displacement

Yanfeng Wang^{1,2}, Meng Li^{1,2}, Junwei Sun^{1,2}(✉), and Chun Huang^{1,2}

¹ Henan Key Lab of Information-Based Electrical Appliances,
Zhengzhou University of Light Industry, Zhengzhou 450002, China
junweisun@yeah.net

² School of Electrical and Information Engineering,
Zhengzhou University of Light Industry, Zhengzhou 450002, China

Abstract. The method of DNA strand displacement breaks the static thinking of DNA nanotechnology, which makes the biochemical cascade reaction, nanoscale motion and energy conversion widely used in the logic gate operating model. The cubic logical circuit of three-input and nine-output based on DNA strand displacement is designed in this article. The cubic logic circuit can be translated into the dual-rail logic circuit and the dual-rail logic circuit can be translated into the DNA seesaw logic circuit, then it can be simulated through the Visual DSD software. It demonstrated that the correctness of logic circuit through the simulation results. DNA strand displacement has gigantic capable of implementation of logical calculation which plays a momentous role in the acquirement of bio-computer, and it is most widely used in the majority computing systems. At the same time, the difficult problems in the construction of large-scale complex logic circuits can be solved, and have great significance to research.

Keywords: DNA strand displacement · Digital circuit
Logical gate operation of biochemical logical circuit · Simulation

1 Introduction

DNA strand displacement is a dynamic DNA nanotechnology, which is developed on the basis of DNA self-assembly technology [1, 2]. Toehold hybridization expedites strand displacement [3]. The dynamic operations of most DNA devices, including circuits, machines etc. and rely on networks of DNA strand-exchange reactions [4–6]. With the continuous development of science and technology, DNA computing has become a new field that combining computer science and molecular biology subject [7, 8]. Of course, lots of biochemistry circuits have been constructed [9, 10] and many problems have been solved by using DNA strand displacement technology [11, 12]. For instance, it shows how an important kind of nonlinear feedback controllers can be designed [13]. An architecture for the systematic construction of DNA circuits for analog computation be proposed [10]. A noncovalent DNA catalysis network resembles an allosteric enzyme be constructed [4] and so on. The digital circuits also have

their advantages, and the design of error-correction schemes digital circuits has developed better [14, 15].

Based on the strand displacement [1], molecular systems can exhibit autonomous brain-like behaviors [16]. Compared to previous articles [17], in 2011, the digital logic circuits were demonstrated, and the four-bit square-root circuit was achieved. There are some advantages, which can make our study more creative and stronger than the previous logic circuit. Firstly, the strategy based on DNA strand displacement was applied to the design of cubic circuit. Secondly, in this paper, the dual-rail circuits with high accuracy were used. Thirdly, the cubic circuit makes the input from 0 to 7 can produce corresponding output (can turn “ON” and “OFF” repeatedly as their inputs change [16]), and the output value is great, so the binary output is more [14], which makes the cubic circuit more complex than square root, thus the superiority of DNA strand displacement was reflect better. Integrating the logic gates into a circuit was regarded as a successful attempt [18].

The benefits of the cubic circuit and the specific research process and method are introduced in the first part of the article. Next, the background of DNA strand displacement and the significance of studying this circuit are introduced in the introduction. In the second section, the reaction mechanism of DNA strand displacement is briefly introduced. In the third section, the relevant theory and methodology of this paper are briefly introduced. The Digital logic circuit, the Truth table, the Dual-rail logic circuit, the Seesaw circuit and the simulation are given in Sect. 4. The application prospect of the technology was put forward in Sect. 5.

2 Mechanism of DNA Strand Displacement Reaction

As a carrier of genetic information, DNA is composed of four bases, through the base pairing to form double helix. DNA strand displacement refers to the reaction process in which one single-stranded DNA displaces the original binding strand in a part of the complex. The binding force of the double helix structure changes as the length of the complementary strand changes. The strand displacement reaction is a clever use of this feature, thereby realizing the process of replacing the long complementary strand with the short complementary strand. DNA strand displacement branch migration process is shown in Fig. 1, the single-stranded DNA molecule A reacts with multi-stranded DNA complex X to generate strand B and complex Y. The ‘toehold’ domains 3 and 3* promote strand displacement reactions: the coalescence sites of these single-stranded sites co-localize A and X, and allow the 2 domain “branch migration”. Branch migration is a random walk process. Among them, one domain replaces the other identical sequence by a series of reversible single nucleotide dissociation and hybridization steps. Upon completion of the branch migration, complex Y is formed and strand B is released.

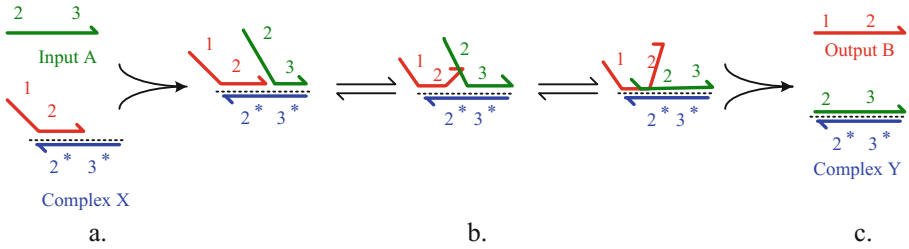


Fig. 1. DNA strand displacement reaction can be divided into three stages. a. It calls the initialization process: the area is the complementary double-stranded of 3 and 3* through a certain binding. b. It calls the branch migration process. c. The third stage is the generation process of output signal. The original strand falls off from the part of the double strand complex, forming an output.

3 Digital Logic Gates Implemented with the Seesaw DNA Motif

The two concepts are mentioned here. One is the amplifying gate and the other is the integrating gate. An integrating gate followed by an amplifying gate can compute either OR or AND. Threshold processing can be combined directly with the seesaw catalyst to support digital abstraction by converting the intrinsic analog signal to the desired “ON” or “OFF” value. Threshold is an important part of the design of the seesaw circuit. When there are two inputs, the AND gate’s threshold value is 1.2, the OR gate is 0.6. When there are three inputs, the AND gate’s threshold value is 2.2, the OR gate is 0.6. When there are four inputs, the AND gate’s threshold value is 3.2, the OR gate is 0.6 (Fig. 2).

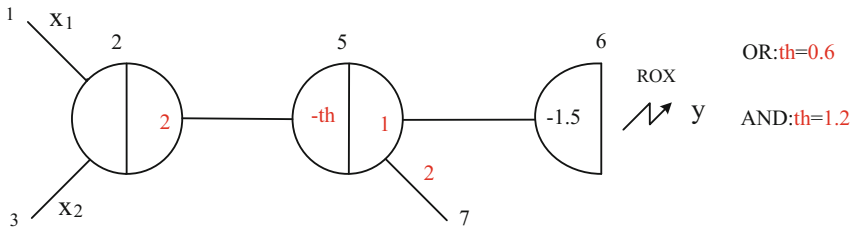


Fig. 2. A seesaw circuit calculates or abstracts the map according to the initial concentration of the threshold.

4 Digital Circuit and Biochemical Circuit

4.1 Truth Table

According to the conversion rules between binary and decimal, $0 = X_3^0 X_2^0 X_1^0 = 000$; $1 = X_3^0 X_2^0 X_1^1 = 001$; $2 = X_3^0 X_2^1 X_1^0 = 010$; $3 = X_3^0 X_2^1 X_1^1 = 011$; $4 = X_3^1 X_2^0 X_1^0 = 100$;

$5 = X_3^1 X_2^0 X_1^1 = 101$; $6 = X_3^1 X_2^1 X_1^0 = 110$; $7 = X_3^1 X_2^1 X_1^1 = 111$, this truth Table 1 clearly lists the cubic operations from 0 to 7, and it has reached a three-input, nine-output binary operation. The cube operation truth table can be listed as shown below. X_1, X_2, X_3 are three inputs; Y_1, Y_2, \dots, Y_9 are nine outputs, and X_3 and Y_9 are top digits. For example, when the input is 010, the output is 000001000, in other words, when the input is 4, the output is 64.

Table 1. Three-input cubic logic circuit truth table

X_3	X_2	X_1	Y_9	Y_8	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1
0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	1	0	0	0
0	1	1	0	0	0	0	1	1	0	1	1
1	0	0	0	0	1	0	0	0	0	0	0
1	0	1	0	0	1	1	1	1	1	0	1
1	1	0	0	1	1	0	1	1	0	0	0
1	1	1	1	0	1	0	1	0	1	1	1

4.2 Digital Logic Circuit

Based on the above truth table, the circuit was drawn by using VISIO as shown in Fig. 3. In order to calculate the output of three binary numbers, three non-gates, seven AND gates and six OR gates were designed. Through these logical gates to perform logical operations, the final three-input cube operation was implemented. In the final simulation diagram, the circuit is used to write the desired program. Digital circuits process information encoded by binary bits, and each bit has two possible values of “0” and “1”.

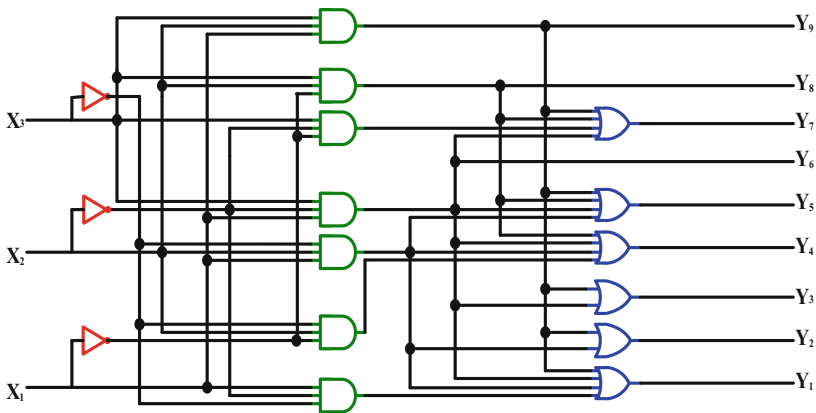


Fig. 3. Three bits binary cubic logic circuit.

4.3 Dual-Rail Logic Circuit

There are NOT gates shown in Fig. 3. Because the NOT gate must be operated in a low input signal reacted by an upstream gate in DNA strand displacement reaction, which should be released a high output signal, and a low input signal that can't be computed, so the NOT gate can't be reacted with others. In dual-rail logic circuit, the AND gate has four inputs and two outputs, the OR gate also has four inputs and two outputs. The inputs are as follows: X_1^0 , X_1^1 , X_2^0 , X_2^1 and the outputs are Y^0 and Y^1 (Fig. 4).

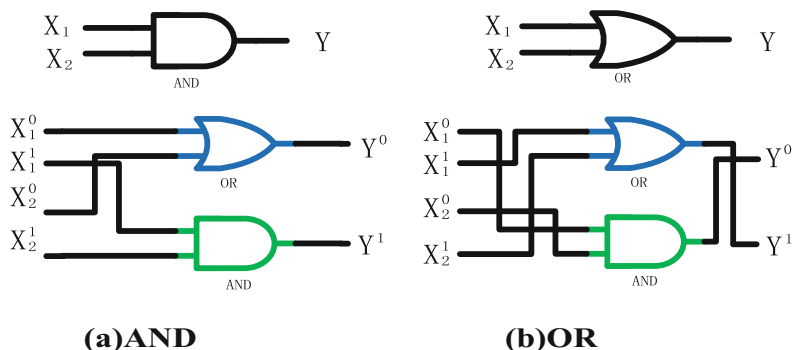


Fig. 4. This is the dual-rail plan of AND gate and OR gate. By using the two gates, the dual-rail logic circuit can be drawn.

In this section, each of original input signals is translated into two inputs, for instance, the X_i is translated into the X_i^0 and the X_i^1 . And they can be shown logic “OFF” and logic “ON”. For example, the cubic logic circuit of three inputs and nine outputs are designed in the Fig. 3. But in the dual-rail circuit, the cubic logic circuit of six inputs and eighteen outputs are designed. In addition, the dual-rail logic algorithm is adopted to avoid the false outputs, which can be gained uncertain computation results due to the absence of the input signals (Fig. 5).

4.4 Seesaw Circuit

As shown in the Fig. 6(a), when the logical value of input one or input two is “1”, then the logical value of output three is also “1”; if the logical value of input one and input two is “0”, then the logical value of output three is “0”; As shown in the Fig. 6(b), when the logical value of input one and input two is “1”, then the logical value of output three is “1”; if the logical value of input one or input two is “0”, then the logical value of output three is also “0”.

Seesaw circuit applied to the logic gates of biochemical reactions mainly includes the following four gates: amplifying gate, integrating gate, threshold gate and report gate. In the seesaw cascade circuit, amplifying gate with one input and seven outputs includes threshold and fuel produces multiple outputs. Following the integrate gate, the threshold gate is divided into two variants: AND gate and OR gate, which are different

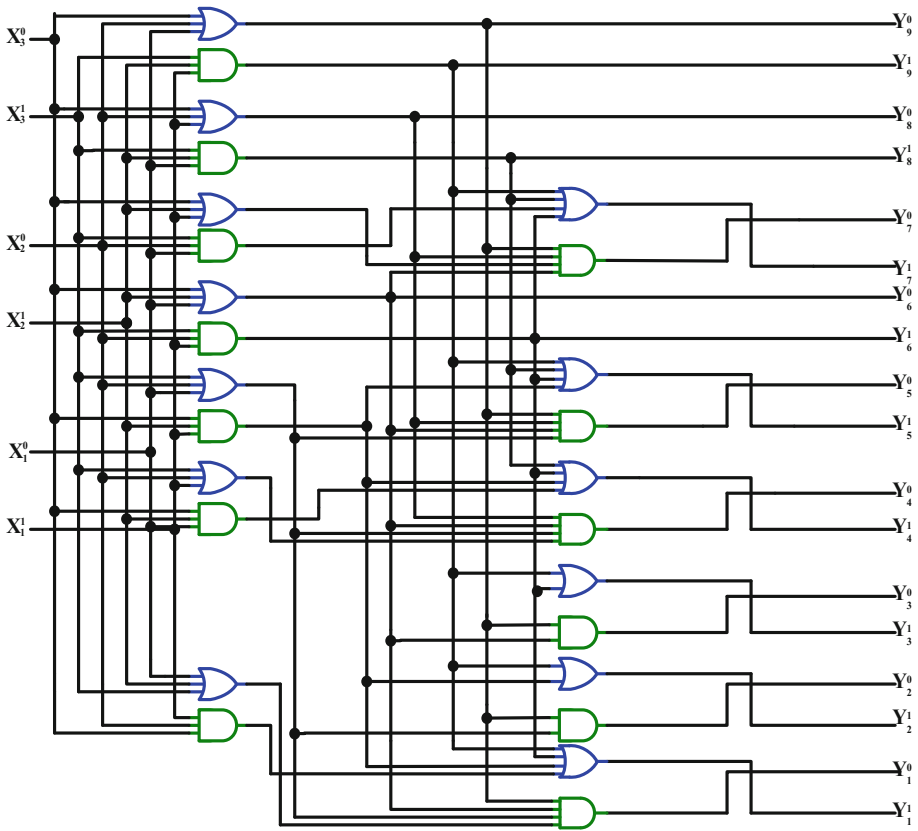


Fig. 5. 3-input and 9-output dual-rail logic circuit. This circuit is very similar with the biochemical circuit.

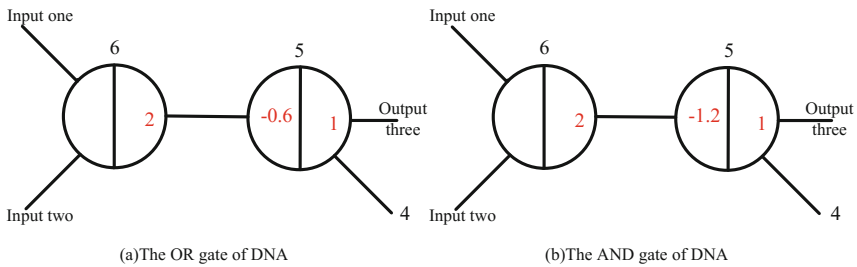


Fig. 6. Seesaw circuit applied to the logic gates of biochemical reactions.

for the threshold values. An AND (\wedge) or OR (\vee) gate was implemented in each pair of seesaw gates. One AND, NOT, OR, NAND, or NOR gate was implemented in each pair of dual-rail AND or OR gates. The whole seesaw cascade circuit is shown in Fig. 7.

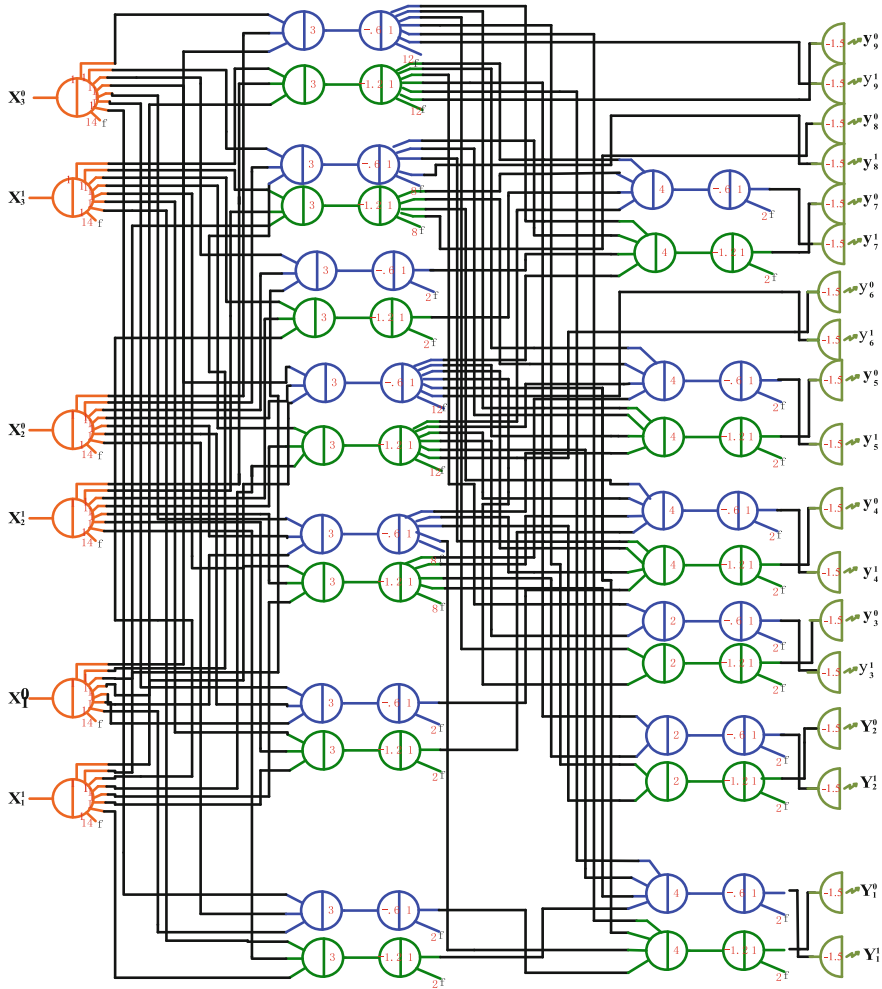


Fig. 7. The seesaw logic circuit of the three bit cubic logic circuit.

4.5 Simulation

By using the visual DSD simulation software, the following trajectory can be gained. The Visual DSD (DNA Strand Displacement) tool allows rapid prototyping and analysis of computational devices implemented using DNA strand displacement, in a convenient web-based graphical interface. The continuous-time Markov chain of a (sufficiently small) system can also be constructed and visualized by the tool, which is particularly useful for low-level debugging of individual circuit components. By turning “ON” and “OFF” repeatedly as their inputs change. That is to say, the input signal is changed as the change of “NO” and “OFF” in the simulation program. The OFF signals may be in the range 0 to 0.2 and the ON signals may be in the range

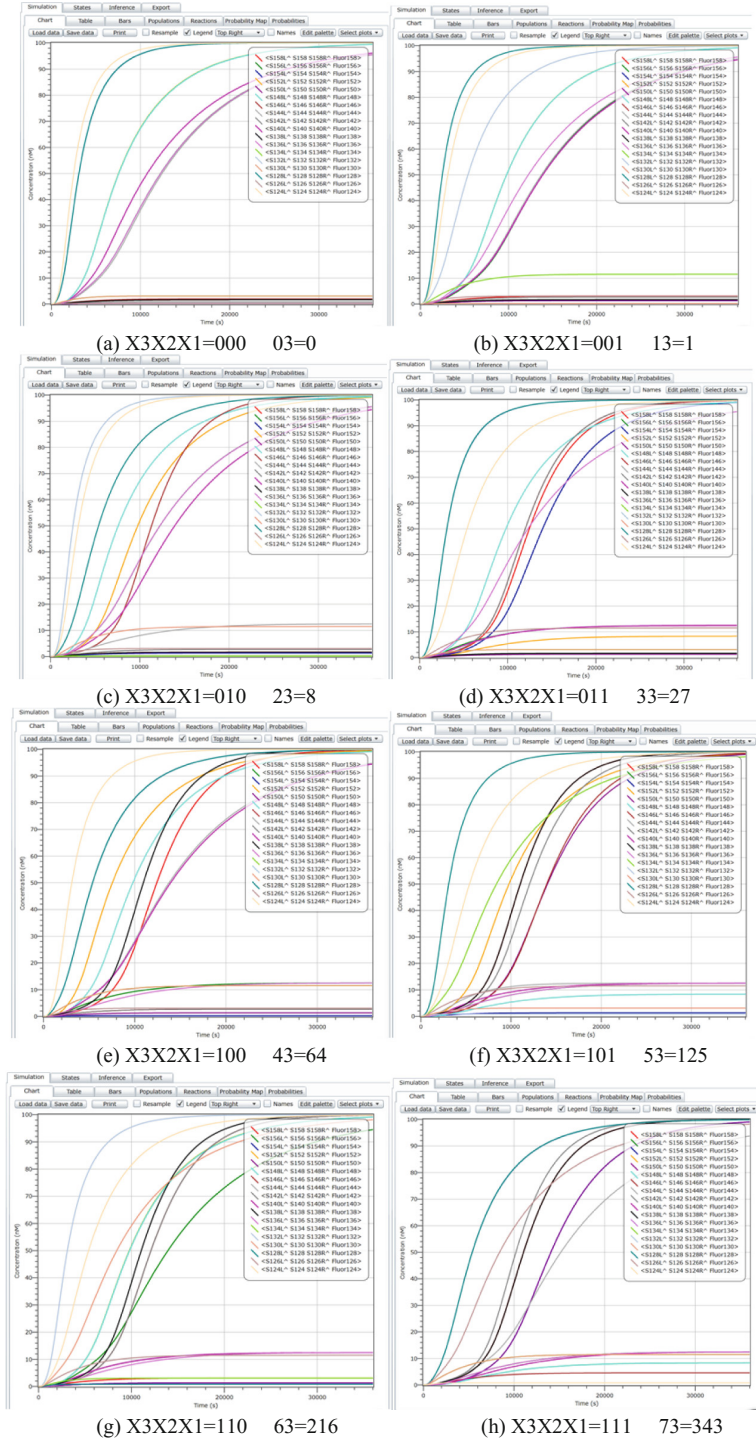


Fig. 8. The simulation results with Visual DSD. The threshold value of OFF is “0.1x”, so the threshold value of ON is “0.9x”.

0.8 to 1. At the same time, the provisions of the DNA strand concentration in the range of 0–100 nm, the corresponding logical value is “0”, the DNA strand concentration in the range of 900–1000 nm, the corresponding logic value “1”. In the schematic, each logic gate, input signals and output signals are labeled, and the logical relationship between them is written in software in a standard language. The software recognizes the program and performs simulation of the corresponding input signals. The simulation results of the three-input-nine-output cube circuit are shown in Fig. 7 (Fig. 8).

5 Conclusion

In this article, the cubic logic circuit based on DNA strand displacement was created. In the last section, it has demonstrated the feasibility of the cubic logic circuit through the DSD simulation software. Among them, it has got the simulation curve of each value by changing “OFF” and “ON” to change the input value, which can be converted to decimal to verify its correctness. DNA strand displacement technology can be used in large-scale circuits, which can provide the feasibility for the future applications. DNA strand displacement technology gradually shows a series of functional Nano-device innovation advantage. At present, it is the focus of domestic and foreign research on the use of DNA technology input and output function of the strand displacement logic gate, and has reached a tremendous development. Next it will focus on the functions and uses of DNA strand displacement technology.

Acknowledgment. The work is supported by the State Key Program of National Natural Science of China (Grant No. 61632002), the National Key R and D Program of China for International S and T Cooperation Projects (No. 2017YFE0103900), the National Natural Science of China (Grant Nos. 61603348, 61775198, 61603347, 61572446, 61472372), Science and Technology Innovation Talents Henan Province (Grant No. 174200510012), Research Program of Henan Province (Grant Nos. 172102210066, 17A120005, 182102210160), Youth Talent Lifting Project of Henan Province and the Science Foundation of for Doctorate Research of Zhengzhou University of Light Industry (Grant No. 2014BSJJ044).

References

1. Zhang, D.Y., Seelig, G.: Dynamic DNA nanotechnology using strand-displacement reactions. *Nat. Chem.* **3**(2), 103 (2011)
2. Qian, L., Winfree, E.: A simple DNA gate motif for synthesizing large-scale circuits. *J. R. Soc. Interface*, Article ID rsif.2010.0729 (2011)
3. Genot, A.J., Bath, J., Turberfield, A.J.: Combinatorial displacement of DNA strands: application to matrix multiplication and weighted sums. *Angew. Chem. Int. Ed.* **52**(4), 1189–1192 (2013)
4. Yang, X., Tang, Y., Traynor, S.M.: Regulation of DNA strand displacement using an allosteric DNA toehold. *J. Am. Chem. Soc.* **138**(42), 14076–14082 (2016)
5. Jung, C., Allen, P.B., Ellington, A.D.: A stochastic DNA walker that traverses a microparticle surface. *Nat. Nanotechnol.* **11**(2), 157 (2016)

6. Stojanovic, M.N., Stefanovic, D., Rudchenko, S.: Exercises in molecular computing. *Acc. Chem. Res.* **47**(6), 1845–1852 (2014)
7. Winfree, E.: DNA computing by self-assembly. In: 2003 NAE Symposium on Frontiers of Engineering, pp. 105–117 (2004)
8. Wang, Z., Wu, Y., Tian, G.: The application research on multi-digit logic operation based on DNA strand displacement. *J. Comput. Theor. Nanosci.* **12**(7), 1252–1257 (2015)
9. Sun, J., Li, X., Cui, G.: One-bit half adder-half subtractor logical operation based on the DNA strand displacement. *J. Nanoelectron. Optoelectron.* **12**(4), 375–380 (2017)
10. Song, T., Garg, S., Mokhtar, R.: Analog computation by DNA strand displacement circuits. *ACS Synth. Biol.* **5**(8), 898–912 (2016)
11. Chen, X.X., Dong, Y.F., Xiao, S.Y.: DNA and DNA computation based on toehold-mediated strand-displacement reactions. *Acta Phys.* **65**, 178106 (2016)
12. Eckhoff, G., Codrea, V., Ellington, A.D.: Beyond allostery: catalytic regulation of a deoxyribozyme through an entropy-driven DNA amplifier. *J. Syst. Chem.* **1**(1), 13 (2010)
13. Sawlekar, R., Montefusco, F., Kulkarni, V.V.: Implementing nonlinear feedback controllers using DNA strand displacement reactions. *IEEE Trans. Nanobiosci.* **15**(5), 443–454 (2016)
14. Sarpeshkar, R.: Analog versus digital: extrapolating from electronics to neurobiology. *Neural Comput.* **10**(7), 1601–1638 (1998)
15. Sauro, H.M., Kim, K.: Synthetic biology: it’s an analog world. *Nature* **497**(7451), 572 (2013)
16. Qian, L., Winfree, E., Bruck, J.: Neural network computation with DNA strand displacement cascades. *Nature* **475**(7356), 368 (2011)
17. Qian, L., Winfree, E.: Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**(6034), 1196–1201 (2011)
18. Zhu, J., Zhang, L., Dong, S.: Four-way junction-driven DNA strand displacement and its application in building majority logic circuit. *ACS Nano* **7**(11), 10211–10217 (2013)



A Simulated Annealing for Multi-modal Team Orienteering Problem with Time Windows

Yalan Zhou^{1,2(✉)}, Chen Li¹, and Yanyue Li³

¹ College of Information, Guangdong University of Finance and Economics, Guangzhou 510320, China

zhouylan@163.com

² Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Sun Yat-sen University, Guangzhou 510006, China

³ Department of Computer Science, Sun Yat-sen University, Guangzhou 510006, China

Abstract. Team orienteering problem (TOP) is NP-hard problem. Recently, a multi-modal team orienteering problem with time windows (MM-TOPTW) as a new extension of TOP is developed. Many real-world applications can be modeled as MM-TOPTW. In this paper, a simulated annealing (SA) is designed for MM-TOPTW. In the SA, a temperature reannealing scheme is adopted to get away the local optimum, and multiple neighborhood searches are carefully designed to improve solution. The computation results demonstrate the proposed algorithm can obtain better solution than the recently proposed algorithm for MM-TOPTW.

Keywords: Team orienteering problem with time windows · Multi-modal Simulated annealing

1 Introduction

Team orienteering problem (TOP) can be found in many application areas. It is a known NP-hard problem. The objective of TOP is to determine a set of paths, each path limited by a predefined time budget, that maximize the total collected score. The first heuristic methods for orienteering problem are described in 1984 [1]. Over the past several decades, many variants of TOP and a number of heuristic and metaheuristic approaches have been proposed to solve TOP [2]. One of the classical variants of TOP is team orienteering problem with time windows (TOPTW) [2, 3]. In this variant, each vertex is associated with a time window constraint and visiting the vertex only can start during this time window. Many metaheuristic algorithms have been developed for TOPTW, such as iterated local search [4], ant colony optimization [5], simulated annealing [6]. Recently, a new extension of TOP, a multi-modal team orienteering problem with time windows (MM-TOPTW) [7] is proposed. In the MM-TOPTW, several modes of transportation are available, such as walk, bicycle, bus, metro and taxi. But only one mode of transportation can be selected to travel between each two adjacent vertices. Many real-world application problems can be modeled as MM-TOPTW, such as tourist trip design problem [8], the maximum collection problem [9],

the bank robber problem [10], the home fuel delivery problem [11]. A two-level particle swarm optimization with multiple social learning terms (2L-GLNPSO) is presented to solve the problem [7]. 2L-GLNPSO showed competitive results with the state-of-art algorithms.

In this study, a simulated annealing for MM-TOPTW is proposed. In this SA, a temperature reannealing scheme is used to help SA to escape from being trapped into a local optimum and multiple neighborhood searches (MNS) are incorporated to improve solution. The experiment is carried on three instance groups and each group with 56 instances. The simulation results show the SA can produce better solution with less time than the recently proposed algorithm for MM-TOPTW called 2L-GLNPSO.

2 MM-TOPTW Formulation

MM-TOPTW is defined as follows. Given a directed graph $G = (N, E)$, where $N = \{1, \dots, n\}$ is a set of vertices, $E = \{(i, j): i \neq j \in N\}$ is the set of edges. Each vertex $i \in N$ is associated with a nonnegative score S_i , a starting service time O_i , an ending service time C_i , a service time d_i , a visiting cost w_i . The start and end vertex are fixed to vertex 0, and $S_0 = 0$. For each edge $(i, j) \in E$ is associated with a pre-specified time limit T_{\max} , a pre-specified travel budget limit B_{\max} , a transportation mode v in each edge is selected from the set of transportation options V .

A solution is represented by a two-level vector in this study. The upper level with $|Np| + |P| - 1$ dimensions represents the order of visited vertex in all visiting paths, except for the first and final vertex 0. $|Np|$ is the number of visited vertices. $|P| - 1$ is the number of zeros to separate the path. The lower level with $|Np| + |P|$ dimensions represents the transportation mode used at each edge. Figure 1 is an example of solution representation with the number of vertices $|N| = 10$ and the total paths $|P| = 2$, the set of transportation options $V = \{0, 1, 2\}$. Each vertex is associated with a time window, such as vertex 1 associate with time window $[15, 67]$ in Fig. 1(a). The first path visits vertex 1 using transportation mode 1, followed by vertices 3, 4 and 5 using transportation modes 2, 0 and 0, respectively. The path is then ended at the depot 0 by transportation mode 1. Vertex 2 cannot be visited by the first path because of the time window constraint. The sequence of the second path is 6, 10 and 8. The start and end vertex is 0. The transportation modes respectively are 1, 2, 1 and 0. Vertex 7 and 9 are excluded from the path for violating the time window constraint. The total dimension is 8 in the upper level and is 9 in the lower level in Fig. 1(b). The total score of the two paths in Fig. 1 is the score sum collected from visited vertex $Np = \{1, 3, 4, 5, 6, 8, 10\}$.

The objective of MM-TOPTW is to determine a set of paths P that maximizes the total collected score. Then the MM-TOPTW can be formulated with the following three decision variables: $m_{ijv} = 1$, if transportation mode v is available in a visit from vertex i to vertex j in a path, 0 otherwise. $x_{ijp} = 1$, visiting vertex i is immediately followed by visiting vertex j in path p , 0 otherwise. $y_{ip} = 1$ if vertex i is visited in path p , 0 otherwise.

$$\text{Max} \sum_{p \in P} \sum_{i \in N} S_i y_{ip} \quad (1)$$

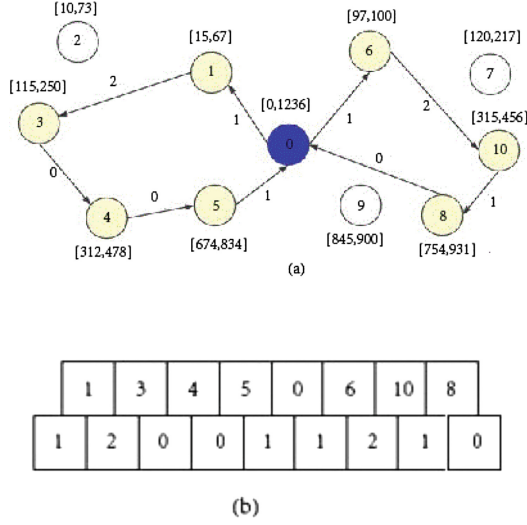


Fig. 1. Solution and its representation. (a) Solution. (b) Representation

Subject to

$$\sum_{p \in P} y_{ip} \leq 1 \quad \forall i \in N \quad (2)$$

$$\sum_{p \in P} \sum_{i \in N} x_{0ip} = \sum_{p \in P} \sum_{i \in N} x_{i0p} = |P| \quad (3)$$

$$\sum_{i,j \in N} \sum_{v \in V} t_{ijv} m_{ijv} + \sum_{i \in N} d_i y_{ip} \leq T_{max} \quad \forall p \in P \quad (4)$$

$$\sum_{i,j \in N} \sum_{v \in V} c_{ijv} m_{ijv} + \sum_{i \in N} w_i y_{ip} \leq B_{max} \quad \forall p \in P \quad (5)$$

$$b_{ip} + d_i + (t_{ijv} m_{ijv}) - b_{jp} \leq T_{max} (1 - x_{ijp}), \quad \forall i \in N, \forall j \in N, v \in V, p \in P \quad (6)$$

$$O_i \leq b_{ip} \leq C_i \quad \forall i \in N, \forall p \in P \quad (7)$$

where t_{ijv} and c_{ijv} are the travel time and travel cost of a visit using transportation mode v from vertex i to vertex j , respectively. b_{ip} is the start service time at vertex i in path p . In this formulation, constraint (2) ensures that all vertices (except for vertex 0) can be visited at most once, constraint (3) guarantees that each path starts and ends at vertex 0. Constraints (4) and (5) respectively limit the time and cost budget of each path. Constraint (6) determines the timeline of each path, which describes it has enough time to the next vertex after completing the preview vertex. Constraint (7) restricts the start of the service to the time windows.

3 Simulated Annealing for MM-TOPTW

The simulated annealing (SA) is a single point-based search metaheuristic that was introduced in 1983 [12]. The basic idea behind the SA is accepting worse solutions with a small probability during its iterations. The small probability partly is determined by the temperature T . The higher value of T , the larger probability of accepting a worse solution.

Algorithm 1 shows the pseudocode of the proposed SA. In the algorithm 1, the current temperature T is set to the initial temperature T_0 at the beginning and then is decreased at each iterations according to the formula $T = T \times dec1$, where $dec1$ in $(0, 1)$ is the rate of temperature reduction at each iteration. X_{best} denoted the current best solution and $f(X_{best})$ denoted the best objective function value obtained so far. $p = \exp(-(f(X_{new}) - f(X))/T)$ is the probability of accepting a worse solution [12].

In the Algorithm 1, a temperature reannealing scheme is introduced as following. The temperature T is reset ($T_0 = T_0 \times dec2$ and $T = T_0$) when its value is too low (less than a given temperature value T_{min}), where $dec2$ in $(0, 1)$ is the rate of initial temperature reannealing at each iteration.

In the Algorithm 1, multiple neighborhood searches, as $MNS(X)$, are employed to generate a new solution. The following four neighborhood searches in $MNS(X)$ are considered.

Swap: This operator randomly selects two vertices separately from the visited vertices and the non-visited vertices, and then exchanges the two vertices. Figure 2(a) shows an example of the swap operator. The visited vertices set $\{0, 1, 2, 3, 4, 5, 8\}$ are denoted as yellow nodes. The non-visited vertices set $\{6, 7, 9\}$ are denoted as white nodes. Vertex 2 and vertex 6 are selected to exchange in Fig. 2(a).

Algorithm 1 The proposed Simulation Annealing

```

initialization  $X, T = T_0$ ;
 $X_{best} = X$  and  $f(X_{best}) = f(X)$ ;
while stopping condition is not met do
  generate a new solution  $X_{new}$  by  $MNS(X)$ ;
  if  $f(X_{new}) > f(X)$ 
     $X = X_{new}$ ;
  else if  $\text{random}(0, 1) < p$ 
     $X = X_{new}$ ;
  end if;
  if  $f(X) > f(X_{best})$ 
     $X_{best} = X$ 
  if  $T > T_{min}$   $T = T \times dec1$ 
  else  $T_0 = T_0 \times dec2$ 
    if  $T_0 > T_{min}$  then
       $T = T_0$ ;
    end if
  end if
  end if
end while
output  $X_{best}$ 

```

Insertion: This operator randomly selects vertex i from the non-visited vertices and then inserted it into the position before randomly selected vertex j from the constructed path. Figure 2(b) shows an example of the insertion operator. Vertex 9 is selected and inserted to the position before vertex 3.

Inversion: This operator randomly selects two vertices from a randomly selected path, and then reversing the sequence between the two vertices. Figure 2(c) shows an example of the inversion operator. Vertex 4 and 8 are selected from the selected path 2. The order of visited vertices between vertex 4 and 8 is reversed.

Mode Change: This operator randomly selects an edge from the constructed path and randomly changes the edge’s transportation mode from the set of transportation modes V . Figure 2(d) shows an example of the mode change operator. Edge (2, 5) in path 1 is selected and changes its transportation mode 2 to 0.

Each kind of the neighborhood searches has equal probability to be selected to perform at each iteration.

Only feasible solutions are considered in the algorithm. Therefore, the Constraints (2)–(7) should be checked once there are some changes are happened in a path. The objective function is the total collected score from all visited vertices, which must be reevaluated after a new feasible solution is generated.

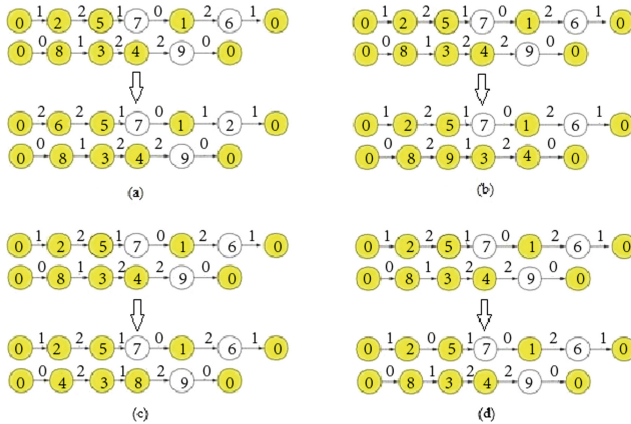


Fig. 2. Examples of four neighborhood searches. (a) An example of the swap operator. (b) An example of the insertion operator. (c) An example of the inversion operator. (d) An example of the mode change operator

4 Experimental Results

The simulations were implemented in C++ language and carried out a Windows 7 PC with 2.7 GHz CPU and 4 GB RAM. The MM-TOPTW test instances are generated from Solomon’s database of vehicle routing problem with time windows [13]. These test instances are divided into three groups: small-sized instances with 25 vertices, medium-sized instances with 50 vertices and large-sized ones with 100 vertices. Each group consists of 56 instances.

Based a preliminary test, the parameter values of SA in this study were set as follows: $T_0 = 10000$, the stopping condition used is the number of iterations reaching 1000000, $T_{\min} = 0.1$, $dec1 = 0.999$ and $dec2 = 0.01$. All the results presented in this study were obtained by running 30 iterations.

To the best of our knowledge, 2L-GLNPSO is the newest heuristic algorithm and has the best performance to solve MM-TOPTW. In order to verify the performance, the proposed SA is tested on MM-TOPTW test instances and compared with the 2L-GLNPSO. The experiment results of 2L-GLNPSO are obtained from Ref. [7].

Table 1 shows statistical results and the average *gap* in terms of objective function value between the proposed SA and 2L-GLNPSO on three group instances with three different number of path $|P| = 1, 2, 3$. Statistical results on three group instances are summarized as “*w/t/l*”, which means that the proposed SA in terms of objective function value obtains better than, equal to and worse than 2L-GLNPSO on *w*, *t* and *l* instances, respectively. The “*gap*” is defined as the relative difference of objective function value in percentage between SA and 2L-GLNPSO. It is calculated as follows.

$$gap = \frac{f(SA) - f(2L - GLNPSO)}{f(2L - GLNPSO)} \times 100\%$$

The results in column *w/t/l* of Table 1 show that in terms of objective function value, the proposed SA significantly outperforms 2L-GLNPSO in most of test instances, while it is outperformed by 2L-GLNPSO only 1 in small-sized instances, 1 in medium-sized instances and 7 in large-sized instances, respectively. The results in column average “*gap*” show the proposed SA significantly outperforms 2L-GLNPSO, especially in large-sized instances, the average gap obtained is 4.3% for $|P| = 1$, 7.5% for $|P| = 2$ and 6.5% for $|P| = 3$, respectively.

Table 1. Statistical results comparison and the average gap between SA and 2L-GLNPSO.

Instance	$ P = 1$		$ P = 2$		$ P = 3$	
	<i>w/t/l</i>	<i>gap</i>	<i>w/t/l</i>	<i>gap</i>	<i>w/t/l</i>	<i>gap</i>
Small-sized	29/27/0	2.1%	20/35/1	1.6%	5/51/0	0.1%
Medium-sized	47/9/0	4.1%	35/21/0	3.8%	27/28/1	4.6%
Large-sized	38/12/6	4.3%	56/0/0	7.5%	43/12/1	6.5%

Table 2 shows computational time (seconds) of SA and 2L-GLNPSO on three group test instances with three different number of path $|P| = 1, 2, 3$. The computation time of the proposed SA increases a little with the increasing vertices, but it is still within an acceptable range even with large-sized ones. The 2L-GLNPSO consumes more computation time than the proposed SA on medium-size and large-size data instances.

In summary, the proposed SA provides better results than 2L-GLNPSO in terms of objective function value and computational time.

Table 2. Average computational time (seconds) of SA and 2L-GLNPSO.

Instance	$ P = 1$		$ P = 2$		$ P = 3$	
	SA	2L-GLNPSO	SA	2L-GLNPSO	SA	2L-GLNPSO
Small-sized	18.06	8.5	16.4	11.53	15.06	18.00
Medium-sized	21.064	21.34	18.826	32.56	18.388	39.90
Large-sized	26.102	85.37	23.822	119.73	23.068	164.15

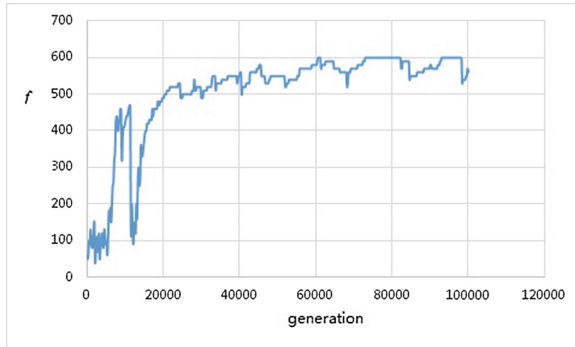
**Fig. 3.** The evolution process of objective function value

Figure 3 plots the convergence behavior of the proposed SA on large-sized instance C101 with 100 vertices and the number of path $|P| = 2$. At the beginning, the fluctuation of the obtained objective function value is very obvious. Before about 5000 generations, the objective function value obtained is small because it is a large probability to accept a worse solution at this stage. After about 20000 generations, it is the temperature reannealing scheme to help the SA to escape from the local optimum.

5 Conclusion

We develop a SA to solve a new extension of TOP, named MM-TOPTW, and compared with the recent approach. Computation experiments show the proposed SA can obtain better results than the 2L-GLNPSO with less time. In the future, this study can be extended from several aspects. Firstly, it can be extended to solving MM-TOPTW with multiple objectives. These objectives are sometimes conflict (e.g. maximizing the collected score, minimizing the time and cost). Secondly, unfeasible solutions may be maintained and utilized by constraint handling technique for possible improvement during the search process [14].

Acknowledgments. This work was supported in part by the Foundation for Distinguished Young Talents in Higher Education of Guangdong, China, under Grant Yqgdufe1404, and in part by the Program for Characteristic Innovation Talents of Guangdong under Grant 2014KTSCX127, and in part by the Opening Project of Guangdong High Performance Computing Society under Grant 2017060109, in part by the Foundation of Key Laboratory of Machine Intelligence and Advanced Computing of the Ministry of Education under Grant MSC-201606A, in part by the China Scholarship Council under Grant 201608440449.

References

1. Tsiligirides, T.: Heuristic methods applied to orienteering. *J. Oper. Res. Soc.* **35**(9), 797–809 (1984)
2. Gunawan, A., Lau, H.C., Vansteenwegen, P.: Orienteering problem: a survey of recent variants, solution approaches and applications. *Eur. J. Oper. Res.* **255**, 315–332 (2016)
3. Kantor, M.G., Rosenwein, M.B.: The orienteering problem with time windows. *J. Oper. Res. Soc.* **43**(6), 629–635 (1992)
4. Vansteenwegen, P.: Iterated local search for the team orienteering problem with time windows. *Comput. Oper. Res.* **36**(12), 3281–3290 (2009)
5. Montemanni, R., Weyland, D., Gambardella, L.M.: An enhanced ant colony system for the team orienteering problem with time windows. In: *International Symposium on Computer Science and Society 2011*, vol. 2011, pp. 381–384. IEEE Computer Society (2011)
6. Lin, S.W., Vincent, F.Y.: Solving the team orienteering problem with time windows and mandatory visits by multi-start simulated annealing. *Comput. Ind. Eng.* **114**, 195–205 (2017)
7. Yu, V.F., Jewpanya, P., Ting, C.J., Redi, A.P.: Two-level particle swarm optimization for the multi-modal team orienteering problem with time windows. *Appl. Soft Comput.* **61**, 1022–1040 (2017)
8. Vansteenwegen, P., Souffriau, W., Berghe, G.V., et al.: The city trip planner: an expert system for tourists. *Expert Syst. Appl.* **38**(6), 6540–6546 (2011)
9. Arkin, E.M., Mitchell, J.S.B., Narasimhan, G.: Resource-constrained geometric network optimization. In: *Fourteenth Symposium on Computational Geometry*, vol. 1998, pp. 307–316. ACM (1998)
10. Butt, S.E., Cavalier, T.M.: A heuristic for the multiple tour maximum collection problem. *Comput. Oper. Res.* **21**(1), 101–111 (1994)
11. Golden, B.L., Levy, L., Vohra, R.: The orienteering problem. *Nav. Res. Logist.* **34**(3), 307–318 (1987)
12. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. *Science* **220**(4598), 671–680 (1983)
13. Solomon, M.M.: Algorithms for the vehicle routing and scheduling problems with time window constraints. *Oper. Res.* **35**, 254–265 (1987)
14. Kobeaga, G., Merino, M., Lozano, J.A.: An efficient evolutionary algorithm for the orienteering problem. *Comput. Oper. Res.* **90**, 42–59 (2018)



Discrete Harmony Search Algorithm for Flexible Job-Shop Scheduling Problems

Xiuli Wu^(✉) and Jing Li

School of Mechanical Engineering, University of Science and Technology
Beijing, Beijing 100083, China
wuxiuli@ustb.edu.cn

Abstract. As an emerging algorithm, harmony search (HS) algorithm has been applied into the continuous optimization field and shows amazing performance. The paper aims to study its performance when solving discrete optimization problems. Since the flexible job shop scheduling problem (FJSP) is a typical discrete optimization problem, we propose a discrete harmony search (DHS) algorithm to solve the FJSP. Constructing new solutions by dealing with vector components separately in original HS is inappropriate to combinatorial optimization problems, hence DHS generates new solutions by dealing with solution as a whole. Based on DHS this paper proposes an improved discrete harmony search (IDHS) algorithm. A learning process is added when generating a new solution in IDHS. The candidate solution from harmony memory has the possibility to be adjusted by learning from the current best solution and less possibility not to participate in the learning process, which can help accelerate convergence speed and avoid premature. Computational results show that both DHS and IDHS can search the optimal solution for small and medium scale instances with limited time, but IDHS is more effective in solving large-scale instances.

Keywords: Flexible job-shop scheduling · Discrete harmony search
Improved discrete harmony search · New solution construction
Large-scale instances

1 Introduction

The flexible job shop scheduling problem (FJSP) is developed from the classic job shop scheduling problem and has more flexible constraint conditions. The FJSP problem is consistent with the multi-type and small-batch manufacturing mode of modern enterprises, so the relevant research has important practical value. Meanwhile, the FJSP problem belongs to NP hard problems and its solution space explodes with the increase of problem scale. Traditional precise mathematical methods can't solve this kind of combinatorial optimization problem with high difficulty, while the emerging heuristic algorithms can effectively solve the problem and obtain satisfactory solutions in acceptable time.

To solve the FJSP problem effectively, previous studies often utilize or improve heuristic algorithms, such as genetic algorithm (GA) [1, 2], particle swarm optimization (PSO) [3], ant colony algorithm (ACA) [4], improved bacteria foraging optimization

algorithm (IBFOA) [5], tabu search algorithm (TS) [6] and differential evolution algorithm (DE) [7]. Previous studies also combine two or more algorithms to get better performance [8].

Harmony search algorithm (HS) is a relatively new heuristic algorithm proposed by Korean scholar Geem based on the natural musical improvisation process [9, 10]. At first it is used to solve continuous function optimization problems. HS has fewer parameters compared with many other heuristic algorithms, simple structure, fast speed and strong robustness. Thus, it is gradually extended to combinatorial optimization problems in engineering applications.

HS also has great potential to solve the FJSP problem. Yuan proposes a hybrid harmony search algorithm (HHS) based on the integrated approach for the FJSP problem [11]. Gao designs a Pareto-based grouping discrete harmony search algorithm (PGDHS) to solve the multi-objective FJSP problem [12]. Maroosi introduces a parallel framework based on membrane computing to improve the harmony search [13].

This paper attempts to design a discrete harmony search algorithm for the FJSP problem and provide a new idea to optimize the workshop scheduling. The rest of this paper is organized as follows: Sect. 2 briefly formulates the FJSP problem. Section 3 proposes a discrete harmony search algorithm (DHS) to solve the FJSP problem. Section 4 proposes an improved discrete harmony search (IDHS) algorithm based on the original one. Section 5 reports the experimental results of these two algorithms. Section 6 concludes the paper.

2 The Proposed Model for FJSP

The FJSP problem is the scheduling of multiple jobs on multiple machines. Each operation can be completed by more than one machine and the process time of each operation varies with machine. The FJSP problem mainly consists of two subproblems: One is to arrange machine resources for operation; The other is to determine the starting and completion time of each job's operations.

According to the number of optimization indexes, the FJSP problem can be divided into single objective and multi-objective optimization problem. In general, the maximum completion time (makespan) always determines production efficiency and economic benefits of enterprises. In this paper, makespan is the only optimization index. The notations are listed in Table 1.

The FJSP problem can be formulated as follows:

$$Z = \min\{\max(C_{ij})\} \quad (1)$$

$$C_{ij} \leq S_{uv} \quad (2)$$

$$\sum_{k=1}^m x_{ijk} = 1 \quad (3)$$

$$S_{ij} + x_{ijk} \times p_{ijk} \leq C_{ij} \quad (4)$$

Table 1. Notations for FJSP

Variables	Description
n	Job quantity
N_i	The operation number of job i
O_{ij}	Operation j of job i
m	Machine quantity
M_K	Machine k in machine set M
p_{ijk}	The process time of O_{ij} in machine k
S_{ij}	The starting time of O_{ij}
C_{ij}	The completion time of O_{ij}
x_{ijk}	=1, if O_{ij} is processed in machine k =0, otherwise
y_{ijk}	The process position of O_{ij} in machine k

$$S_{i(j+1)} \geq C_{ij} \quad (5)$$

$$S_{ij} \geq 0 \quad (6)$$

$$C_{ij} \geq 0 \quad (7)$$

$$p_{ijk} \geq 0 \quad (8)$$

For all equations from (1) to (8), $i = 1, 2, \dots, n, j = 1, 2, \dots, N_i, k = 1, 2, \dots, m$. For Eq. (2), $x_{ijk} = 1, x_{uvk} = 1, y_{ijk} = z, y_{uvk} = z + 1, u = 1, 2, \dots, n, v = 1, 2, \dots, N_i$.

The Eq. (1) is the objective function to minimize the maximum completion time. The constraint (2) ensures that one machine can only process one operation at a time. The constraint (3) indicates that one operation can only be processed by one machine. (4) indicates that interruption is not allowed in processing. (5) ensures that each job has fixed procedure sequence. (6), (7) and (8) ensures that variables are nonnegative.

3 The DHS Algorithm for FJSP

3.1 The Framework of DHS

Initially the standard HS is designed for continuous optimization problems. Each decision variable of solution vector is independent of each other and has a continuous range of values. Before proposing the algorithm for the FJSP problem, the basic flow of the standard HS will be briefly introduced. It firstly generates *HMS* (Harmony Memory Size) candidate solutions or harmonies and stores them in *HM* (Harmony Memory). Then in each iteration, it constructs a complete new solution by generating its decision variables one by one. For each decision variable, there is the possibility *HMCR*

(Harmony Memory Consideration Rate) to select a value from HM randomly, and the possibility $1-HMCR$ to generate a random value in its value range. If the value of a decision variable is selected from HM , there is also the possibility PAR (Pitch Adjustment Rate) to adjust it slightly within the range of bw (Band Width). Then compare the new solution with the worst candidate solution of HM . The better one of the two will be retained in HM .

Combinatorial optimization problems have discrete solution space and the decision variables of a solution aren't independent of each other on most occasions. For the FJSP problem, it is not suitable to construct a new solution by dealing with the components of a harmony separately. Therefore, this section proposes a discrete harmony search (DHS) algorithm to solve the FJSP problem. DHS generates new solutions by dealing with solution as a whole.

The main flow chart of DHS is shown in Fig. 1.

And the framework of DHS consists of four basic steps:

Step 1: Initialize the FJSP problem. And the algorithm parameters HMS , $HMCR$, PAR and $Tmax$ (the maximal iteration) are set in this step.

Step 2: Initialize HM randomly. Each harmony in HM is a candidate solution.

Step 3: Start the iteration and update HM repeatedly. The pseudo-code of the iteration process is as follows:

```

For  $Iter=1$  to  $Tmax$  do
  For  $i=1$  to  $HMS$  do
    If  $rand(0-1) < HMCR$  then
       $X^{new} = X^a$ 
      Adjust  $X^{new}$  by  $PAR$ 
    Else
      Generate  $X^{new}$  randomly
    Endif
    Evaluate  $X^{new}$ 
    If  $X^{new}$  is better than  $X^{worst}$ 
      Replace  $X^{worst}$  with  $X^{new}$ 
    Endif
  Endfor
Endfor

```

In the pseudo-code above, X^a is a candidate solution selected randomly from HM . The adjustment operation of X^{new} is illustrated in Sect. 3.2 (3). X^{worst} is the current worst solution in HM .

Step 4: End the iteration and output the best solution in HM .

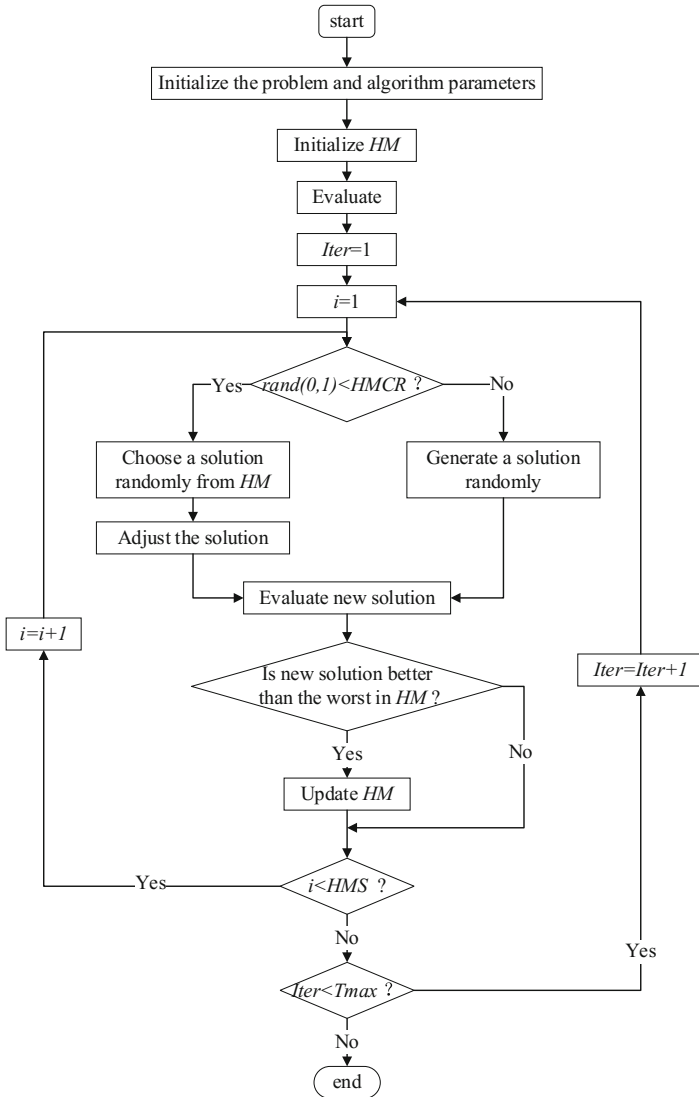


Fig. 1. Flow chart of DHS

3.2 The Details of DHS

(1) Encoding

The operation encoding method is used in DHS. Figure 2 is an example of harmony. Numbers in the harmony represent jobs. Any random job sequence is feasible. Hence, this encoding method can guarantee the feasibility of all solutions.

harmony	1	3	2	1	3	2	3
operation	O_{11}	O_{31}	O_{21}	O_{12}	O_{32}	O_{22}	O_{33}

Fig. 2. An example of harmony

(2) Decoding

DHS uses full-active decoding strategy. It will make full use of machine’s idle time. When the algorithm evaluates a solution, the earliest completion time of an operation on all selectable machines will be calculated and the previous operation of the same job must be completed before the scheduling. Then select the earliest machine which can start processing.

(3) Constructing new solution

When the algorithm generates a new solution, there is the possibility $HMCR$ to select a candidate solution randomly from HM , and also the possibility $1-HMCR$ to generate a new solution randomly. In the former case, each component of the selected solution has possibility PAR to be rearranged randomly with other components, which is the adjustment operation of DHS. Figure 3 illustrates the adjustment operation with an example. The original solution has 10 components and is shown in the first line. The value of PAR is 0.4, which means that there is the possibility 0.4 for each component to be rearranged. In the second line, the white components which are selected by PAR will be rearranged randomly with each other while the gray ones will be retained. The reshaped solution is shown in the third line. Unlike the standard HS, the object of this adjustment operation is the solution itself rather than decision variables of the solution. This operation can guarantee the feasibility of all new solutions.

1	3	2	1	4	2	3	4	3	2
1	3	2	1	4	2	3	4	3	2
4	3	2	1	1	2	4	3	3	2

Fig. 3. The adjustment operation

4 The IDHS Algorithm for FJSP

This section proposes an improved discrete harmony search algorithm (IDHS) based on DHS. The only difference between DHS and IDHS is the way of constructing a new solution after selecting a candidate solution from HM . The candidate solution has a great possibility to be adjusted slightly according to the current best solution. In order

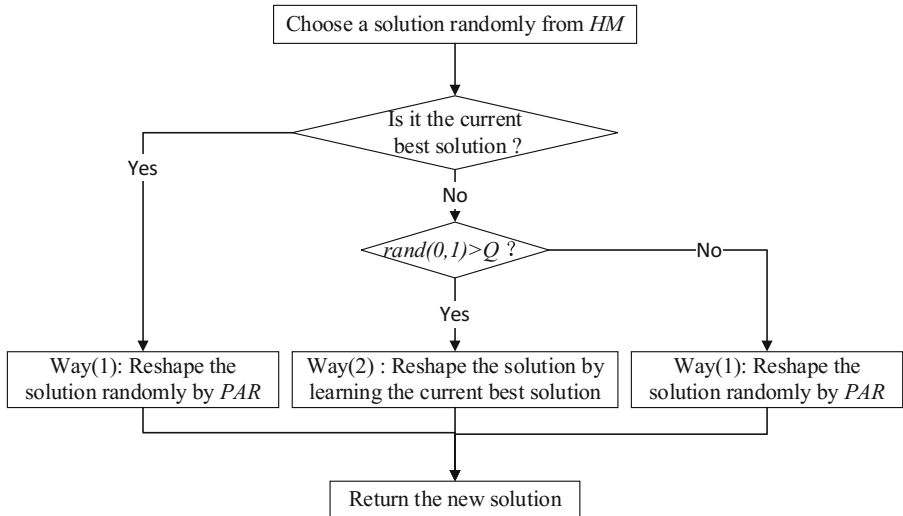


Fig. 4. Two ways of adjusting the candidate solution from *HM*

to avoid the error learning which could lead to local optimization, the candidate solution also has a smaller possibility not to participate in the learning process. Figure 4 shows two ways of adjusting the candidate solution from *HM*.

Way (1): Each component of the selected solution from *HM* has possibility *PAR* to be rearranged randomly with each other.

Way (2): Way (2) is a learning process. Compare the selected solution with the current best solution and retain the components whose value is the same in both solutions. The rest of the components in the selected solution all have possibility *PAR* to be rearranged randomly like way (1).

If the selected solution from *HM* is the current best solution, reshape it by way (1) to construct a new solution. Else generate a random number between 0 and 1 and compare it with Q . If the random number is larger, then construct a new solution by way (2). Else the new solution will be constructed by way (1).

The learning process way (2) is designed to help improve the convergence speed. Q is a decimal between 0 and 1. In order to ensure that the learning process holds a large proportion, the value of Q is 0.2 in IDHS.

5 The Experimental Results

In this section, the performance of DHS and IDHS is tested through two sets of famous FJSP benchmarks: Kacem instances [14] and Brandimarte instances [15]. The experiments are carried out under the Intel Core i3-2350 M, 2.30 GHz CPU, 2.00 GB RAM, Win10 32 bit operating system and MATLAB 2012b.

5.1 Parameters Setting

There are three main parameters in the two algorithms: harmony memory size (*HMS*), harmony memory consideration rate (*HMCR*) and pitch adjustment rate (*PAR*). We set 3 levels for each parameter and thus use the L9(3⁴) orthogonal table (Table 2). For each setting, we run DHS 10 times to test MK02 from Brandimarte instances. To distinguish the effect of different parameter settings, take the average evaluation value of *HM* in the 100th iteration as the result and record it in Table 3. The analyzed results are in Table 4. Results show that *HMS* plays an important role in algorithm’s performance while *HMCR* and *PAR* have less obvious effect. According to the results, the best parameter setting is determined as follows: *HMS* = 20, *HMCR* = 0.9, *PAR* = 0.3.

Table 2. L9 (3⁴) orthogonal table

Number	<i>HMS</i>	–	<i>HMCR</i>	<i>PAR</i>
1	20	1	0.85	0.3
2	20	2	0.9	0.45
3	20	3	0.95	0.6
4	40	1	0.9	0.6
5	40	2	0.95	0.3
6	40	3	0.85	0.45
7	80	1	0.95	0.45
8	80	2	0.85	0.6
9	80	3	0.9	0.3

Table 3. The results of the orthogonal experiment

	1	2	3	4	5	6	7	8	9
1	29.95	30.20	29.95	30.63	30.60	31.13	32.38	32.35	31.73
2	29.35	30.00	29.90	30.73	30.80	31.18	32.26	32.65	31.90
3	30.35	29.90	30.45	30.75	31.13	30.48	32.26	32.44	32.34
4	30.30	30.00	30.05	30.73	30.85	30.83	32.33	32.39	32.09
5	29.95	30.15	30.15	30.73	30.95	30.65	32.44	32.48	32.24
6	29.95	30.30	29.65	30.68	31.08	31.03	32.45	32.61	33.01
7	29.95	29.85	29.90	30.50	30.55	30.95	32.53	32.45	32.91
8	30.15	29.85	30.00	30.70	30.73	30.63	32.06	32.53	32.66
9	29.55	29.90	29.90	30.53	31.25	30.35	32.24	32.19	32.26
10	30.15	30.50	30.25	30.90	30.63	30.70	32.69	32.40	32.23
Avg.	29.97	30.07	30.02	30.69	30.86	30.79	32.36	32.45	32.34

Table 4. The analyzed results

	<i>HMS</i>	–	<i>HMCR</i>	<i>PAR</i>
<i>k1</i>	30.02	31.00	31.07	31.05
<i>k2</i>	30.78	31.06	31.03	31.07
<i>k3</i>	32.38	31.05	31.08	31.05
The best	<i>HMS</i> (1)	<i>HMCR</i> (2)	<i>PAR</i> (1)	

5.2 Benchmark Tests

Each instance with DHS and IDHS is tested 10 times. The maximal iteration $Tmax$ is set according to the problem scale. The results are shown in Tables 5 and 6. Both the two algorithms can easily search optimal solution for small and medium scale problems with limited iterations. But for large scale problems IDHS can search better solutions than DHS. Figure 5 compares the convergence of the best and average value of MK08 with IDHS and DHS. In general, IDHS has a faster convergence speed and can search a more satisfactory solution in shorter time. Therefore the improvement has obvious effect. Figure 6 shows the best Gantt chart of MK09.

Table 5. Results of Kacem instances with DHS and IDHS

Instance	Scale $n \times m$	$Tmax$	Theoretical optimal	DHS			IDHS		
				Best	Avg	RD	Best	Avg	RD
Kacem1	4×5	100	11	11	11	0	11	11	0
Kacem2	6×5	200	31	31	31.7	0	31	31.6	0
Kacem3	8×8	200	14	14	14	0	14	14	0
Kacem4	8×8	200	11	11	11	0	11	11	0
Kacem5	10×10	200	7	7	7	0	7	7	0
Kacem6	15×10	200	11	11	11.9	0	11	11.7	0

Note: Theoretical optimal means the optimal solution of the previous researches.

RD means relative deviation and can be calculated by the following formula:

$$RD = \frac{best - theoretical\ optimal}{theoretical\ optimal} \times 100\% \quad (9)$$

Table 6. Results of Brandimarte instances with DHS and IDHS

Instance	Scale $n \times m$	$Tmax$	Theoretical optimal	DHS			IDHS		
				Best	Avg	RD	Best	Avg	RD
MK01	10×6	300	40	40	40.9	0	40	41.1	0
MK02	10×6	300	26	28	28.4	7.7%	<u>27</u>	28	3.8%
MK03	15×8	300	204	204	204	0	204	204	0
MK04	15×8	1000	60	67	67	11.7%	<u>64</u>	66.7	6.7%
MK05	15×4	1000	172	178	179.4	3.5%	<u>176</u>	177.9	2.3%
MK06	10×15	2000	58	66	66.5	13.8%	<u>62</u>	65.1	6.9%
MK07	20×5	2000	139	147	147.8	5.8%	<u>143</u>	145.2	2.9%
MK08	20×10	500	523	523	523.1	0	523	523	0
MK09	20×10	2000	307	311	313.5	1.3%	307	310.5	0
MK10	20×15	2000	197	229	232.1	16.2%	<u>226</u>	229	14.7%

Note: When neither of DHS and IDHS search the theoretical optimal solutions, the better value of the two will be underlined in this table.

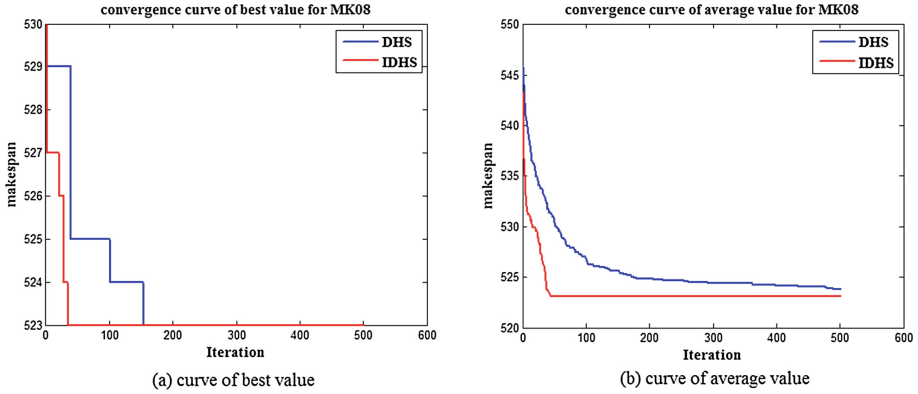


Fig. 5. Convergence curve of MK08 with IDHS and DHS

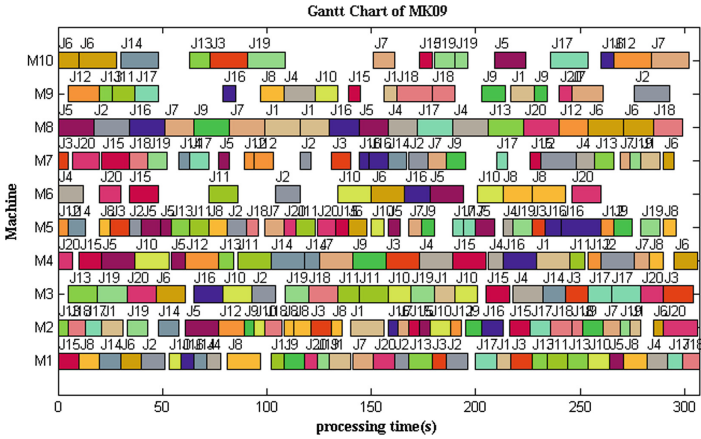


Fig. 6. Best Gantt chart of MK09

6 Conclusion

We propose a discrete harmony search (DHS) algorithm to solve the FJSP problem. Results show that DHS can easily search optimal solutions for small and medium scale problems with limited time but can't search optimal solutions for large-scale problems. Based on DHS we propose an improved discrete harmony search (IDHS) algorithm and compare the performance of these two algorithms. Experimental results show that IDHS can search better solutions with faster convergence speed.

References

1. Ahmadi, E., Zandieh, M., Farrokh, M.: A multi objective optimization approach for flexible job shop scheduling problem under random machine breakdown by evolutionary algorithms. *Comput. Oper. Res.* **73**, 56–66 (2016)
2. Zhang, G.H., Dang, S.J.: Research on flexible job-shop scheduling problem considering job movement time. *Appl. Res. Comput.* **34**(8), 2329–2331 (2017)
3. Chen, M., Hu, L.Y., Liu, J.F.: Multi-objective flexible job shop scheduling problem based on particle swarm optimization. *Mechatronics* **1**, 11–15 (2017)
4. Zhao, B.X., Gao, J.M., Chen, K.: Two-stage hybrid pareto ant colony algorithm for multi-objective flexible job shop scheduling. *J. Xi'an Jiaotong Univ.* **50**(7), 145–151 (2016)
5. Wu, X.L., Zhang, Z.Q., Du, Y.H.: Improved bacteria foraging optimization algorithm for flexible job shop scheduling problem. *Comput. Integr. Manuf. Syst.* **21**(5), 1262–1270 (2015)
6. Lu, H.D., He, W.P., Zhou, X.: An integrated tabu search algorithm for the lot streaming problem in flexible job shops. *J. Shanghai Jiaotong Univ.* **46**(12), 2003–2008 (2012)
7. Wang, W.L., Fan, L.X., Xu, X.L.: Multi-objective differential evolution algorithm for flexible job shop batch scheduling problem. *Comput. Integr. Manuf. Syst.* **19**(10), 2481–2492 (2013)
8. Azzouz, A., Ennigrou, M., Said, L.B.: A self-adaptive hybrid algorithm for solving flexible job-shop problem with sequence dependent setup time. *Procedia Comput. Sci.* **112**, 457–466 (2017)
9. Geem, Z.W., Kim, J.H., Loganathan, G.V.: A new heuristic optimization algorithm: harmony search. *Simulation* **76**(2), 60–68 (2001)
10. Lee, K.S., Geem, Z.W.: A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. *Comput. Methods Appl. Mech. Eng.* **194**(36), 3902–3933 (2005)
11. Yuan, Y., Xu, H., Yang, J.D.: A hybrid harmony search algorithm for the flexible job shop scheduling problem. *Appl. Soft Comput.* **13**, 3259–3272 (2013)
12. Gao, K.Z., Suganthan, P.N., Pan, Q.K.: Pareto-based grouping discrete harmony search algorithm for multi-objective flexible job shop scheduling. *Inf. Sci.* **289**, 76–90 (2014)
13. Maroosi, A., Muniyandi, R.C., Sundarajan, E.: A parallel membrane inspired harmony search for optimization problems: a case study based on a flexible job shop scheduling problem. *Appl. Soft Comput.* **49**, 120–136 (2016)
14. Kacem, I., Hammadi, S., Bome, P.: Approach by localization and multi-objective evolutionary optimization for flexible job-shop scheduling problems. *IEEE Syst. Man Cybern. Soc.* **32**(1), 1–13 (2002)
15. Brandimarte, P.: Routing and scheduling in a flexible job shop by tabu search. *Ann. Oper. Res.* **41**(3), 157–183 (1993)



Barebones Particle Swarm Optimization with a Neighborhood Search Strategy for Feature Selection

Chenye Qiu¹(✉) and Xingquan Zuo²

¹ School of Internet of Things,
Nanjing University of Posts and Telecommunications, Nanjing, China
qiuchenye@njupt.edu.cn

² Key Laboratory of Trustworthy Distributed Computing and Service,
Ministry of Education, Beijing University of Posts and Telecommunications,
Beijing, China

Abstract. Feature selection is a vital step in many machine learning and data mining tasks. Feature selection can reduce the dimensionality, speed up the learning process, and improve the performance of the learning models. Most of the existing feature selection methods try to find the best feature subset according to a pre-defined feature evaluation criterion. However, in many real-world datasets, there may exist many global or local optimal feature subsets, especially in the high-dimensional datasets. Classical feature selection methods can only obtain one optimal feature subset in a run of the algorithm and they cannot locate multiple optimal solutions. Therefore, this paper considers feature selection as a multimodal optimization problem and proposes a novel feature selection method which integrates the barebones particle swarm optimization (BBPSO) and a neighborhood search strategy. BBPSO is a simple but powerful variant of PSO. The neighborhood search strategy can form several steady sub-swarms in the population and each sub-swarm aims at finding one optimal feature subset. The proposed approach is compared with four PSO based feature selection methods on eight UCI datasets. Experimental results show that the proposed approach can produce superior feature subsets over the comparative methods.

Keywords: Feature selection · Multimodal optimization
Barebones particle swarm optimization · Neighborhood search strategy

1 Introduction

Feature selection is an important data preprocessing step in many machine learning and data mining tasks [1]. Datasets with large numbers of features are always involved in such problems. However, not all the features are useful since some features are redundant or irrelevant which would not only bring additional computational burden, but also degrade the performance of the classification models [2]. Feature selection aims to select a small subset of features while retaining necessary and sufficient information to describe the target class. Feature selection can improve the classification accuracy, reduce the computational cost, and speed up the learning process [3].

Feature selection can be generally divided into three categories according to the feature subset evaluation criterion: filter approach [4], wrapper approach [5], and hybrid approach [6]. The filter approach evaluates the feature subsets with the intrinsic characteristics of the data, such as mutual information, information gain, and correlation [7]. The wrapper approach employs a learning algorithm to calculate the classification accuracies of the feature subsets. Generally speaking, the wrapper approach can achieve higher classification accuracy than the filter approach. The filter approach is much more computational efficient and shows better generalization ability since the feature selection process is independent of any classifier. The hybrid approach tries to take advantages of both the wrapper and the filter approaches.

Feature selection is a very difficult task due to the large search space. The search space grows exponentially with the number of features. For a dataset with n features, there are 2^n candidate feature subsets [8]. Therefore, the traditional exhaustive search is impractical in most of the cases. Due to the inefficiency of the traditional search methods, various nature inspired meta-heuristics have been applied to select feature subsets due to their strong global search ability, such as particle swarm optimization (PSO), genetic algorithm (GA), and ant colony optimization (ACO). These population based evolutionary algorithms can explore the entire search space in an acceptable time and have shown promising results in feature selection problems [9–11].

However, most of these methods aim at finding the best feature subset and neglect the fact that there may exist multiple optimal feature subsets in many high-dimensional datasets. Most of the feature selection algorithms can only generate one optimal feature subset in a single run of the algorithm and they cannot locate multiple optimal feature subsets simultaneously. The problem of locating multiple optimal solutions is the so-called multimodal optimization which is very common in many real world applications [12]. Classical meta-heuristic methods which can only obtain single optimum are not suitable for multimodal optimization. Niching method [13] is proposed for solving multimodal optimization. Niching method is the technique of finding and preserving multiple optima in the searching process of the meta-heuristics. It can maintain several sub-groups in the decision space and prevent the whole population from converging to a single peak. Hence, those distant individuals with similar fitness values will be kept. Some representative niching methods are crowding, fitness sharing, speciation, and clearing [14–16]. In recent years, niching methods have gained widespread research interest and have been used in a wide range of real-world optimization tasks [17].

However, the researches on using niching method in feature selection are relatively few. To the best of our knowledge, Kamyab and Eftekhari [18] first studied the use of niching methods in feature selection. They applied several well-known niching methods, such as DFS, r2PSO, r3PSO, r2PSO-lhc and r3PSO-lhc to several meta-heuristic based feature selection methods. Experimental results show that a suitable niching method can enhance the quality of the selected feature subsets. However, some problems still exist when applying niching methods for feature selection. Firstly, most niching methods work well in low-dimensional optimization problems, and their performance in high-dimensional feature selection problems would decrease rapidly. Besides, the niching parameter is a major impediment in applying niching method in

real world applications, and it is very difficult to set the proper niching parameter which can perform well in different datasets.

In order to solve the abovementioned problems and fully investigate the potential of niching method in feature selection problem, this paper proposes a novel feature selection method which integrates barebones particle swarm optimization (BBPSO) [18] with a neighborhood search strategy. As a variant of PSO, BBPSO is almost a parameter-free algorithm. In order to realize stable niches in BBPSO, a nearest neighborhood search strategy is introduced into BBPSO. Then the proposed approach is used to select feature subsets. The rest of this paper is organized as follows. Section 2 introduces some background knowledge briefly. Section 3 presents the feature selection method based on BBPSO with a neighborhood search strategy. Section 4 includes experimental results and analyses. The conclusions are given in Sect. 5.

2 Background

2.1 Barebones Particle Swarm Optimization

BBPSO is a simple but powerful optimizer. BBPSO eliminates the velocity term in the original PSO and uses the Gaussian distribution to explore the search space based on the global best particle ($gbest$) and each particle's personal best ($pbest$). Each particle updates its position as follows:

$$x_{id}^{t+1} = N\left(\frac{pbest_{id}^t + gbest_d^t}{2}, |pbest_{id}^t - gbest_d^t|\right). \quad (1)$$

where x_{id}^t is the value of the d th dimension of particle i in cycle t ; $pbest_{id}^t$ is the d th dimension of the $pbest$ of particle i in cycle t ; $gbest_d^t$ is the d th dimension of the $gbest$ in cycle t . According to (1), the position of each particle is randomly generated by the Gaussian distribution with the mean of $(pbest + gbest)/2$ and the variance of $|pbest - gbest|$. Kennedy [18] also proposed an alternative version called BBPSO-Exp, where the position of each particle is updated by:

$$x_{id}^{t+1} = \begin{cases} N\left(\frac{pbest_{id}^t + gbest_d^t}{2}, |pbest_{id}^t - gbest_d^t|\right), & R < 0.5 \\ pbest_{id}^t, & \text{otherwise} \end{cases} \quad (2)$$

where R is a random number in the range of $[0,1]$. It means that the d th dimension of particle i has 50% possibility to change to the position of its corresponding $pbest$. Hence, the BBPSO-Exp inclines to search around the $pbest$ position.

Compared with the canonical PSO, BBPSO does not need the velocity term and other corresponding controllable parameters, such as the inertia weight and cognitive weight. Hence, it can be applied to various real world optimization problems without specifying the controllable parameters. Zhang *et al.* [19] used BBPSO with a new local leader updating strategy and uniform combination to solve feature selection problem. In [20], BBPSO with an adaptive chaotic jump strategy and a new global best updating mechanism (BBPSO-ACJ) was proposed for feature selection. These researches show

BBPSO is an effective alternative in feature selection problem. However, these BBPSO based feature selection methods can only locate single optimal feature subset.

2.2 Multimodal Optimization

Multimodal optimization aims at locating multiple global and local optima in a single objective optimization problem. It is a much more challenging task than finding a single optimum. Many real-world problems can be modeled as multimodal optimization. The classical evolutionary algorithm (EA) initializes the population randomly and improves the fitness values of the individuals gradually, and finally arrives at the final optimal solution. However, the natural tendency of these algorithms is against the goal of multimodal optimization.

Niching methods have been proposed to detect and preserve multiple optimal solutions. Niching method modifies the searching behavior of EA with the aim of maintaining multiple sub-groups within a single population. Each sub-population aims at finding one global optimal solution and the whole population can locate multiple optimal solutions. Classical niching methods, such as fitness sharing and crowding, are designed for GA. More recently, many niching methods are proposed for other relatively new EAs, such as PSO, ACO, and differential evolution (DE) [21]. PSO is a population based meta-heuristics which is known for its fast convergence speed and ease of implementation. Many PSO based approaches are proposed for multimodal optimization problems, such as fitness Euclidean distance ratio PSO (FERPSO) [22], speciation-based PSO [23], ring topology PSO [24], and locally informed PSO [25].

2.3 Niching Methods for Feature Selection

The aim of feature selection is to choose a small subset of features which is sufficient to describe the target class. In many datasets, there may exist more than one optimal feature subset. EA have been successfully used in feature selection due to its global search ability and computational efficiency. However, existing EA based feature selection methods can only locate one optimal feature subset in a run of the algorithm. Therefore, they may neglect other potential optimal feature subsets. In order to local multiple feature subsets in a single run of the algorithm, this paper will model feature selection as a multimodal optimization problem.

The proposed method owns several benefits: (1) Many datasets involve more than one optimal feature subsets according to a pre-defined feature evaluation criterion. Finding multiple high-quality feature subsets can better explore all the possible combinations of features and reveal the hidden characteristics in the dataset. (2) In feature selection problems, the optimal feature subset in the training dataset may not perform well in the test dataset. In this case, locating multiple feature subsets can provide users more options. When one feature subset does not work well, an alternative subset can be adopted immediately. (3) Even in the dataset with only one optimal feature subset, the niching method can also effectively improve the diversity of the population and prevent the feature selection algorithm from being trapped in the local optima, especially when involving a large number of features.

3 Proposed Approach

This section describes the proposed multimodal feature selection algorithm based on BBPSO with a neighborhood search strategy (called BBPSO-NS). In the canonical BBPSO, particles can learn from all the other particles in the swarm and all the particles may become the leading particle in the swarm to guide the search of other particles. Therefore, BBPSO is prone to converge to one peak quickly. In this study, a neighborhood search strategy is introduced to BBPSO to locate multiple global optimal solutions. By using the neighborhood search strategy, each particle would search in its neighboring area. The whole swarm would form several stable niches and avoid converging to one peak. The details of the algorithm will be discussed in the remaining part of this section.

3.1 Particle Representation

BBPSO is originally designed for continuous optimization problems. Feature selection is a combinatorial optimization problem. In order to extend BBPSO to feature selection problem, a particle decoding scheme is employed to translate a particle into a feature subset. Take particle $x_i = \{x_{i1}, x_{i2}, \dots, x_{iD}\}$ for example, D is the dimension of the search space, i.e., the original number of features in the dataset. Since the position generated by (1) is a vector of real numbers, each dimension of the position needs to be transformed to a discrete value. In this paper, the position is restricted in the range of $[0,1]$ and it can be transformed into its corresponding feature subset as follows:

$$A_{id} = \begin{cases} 1, & \text{if } x_{id} > 0.5 \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where A_{id} denotes the selected feature. $A_{id}=1$ means the d th feature is chosen. Otherwise, this feature is not included in this feature subset.

3.2 Fitness Function

The wrapper approach is employed in this study to evaluate the quality of the feature subsets. The K nearest neighbor algorithm (KNN) is used to compute the classification performance of the feature subsets. KNN is one of the most popular classification algorithms and it is very simple since there is only control parameter in KNN, i.e., the number of neighbors. KNN is widely used in wrapper based feature selection methods due to its computational efficiency and robustness. The fitness value of each particle is computed as follows:

$$F(x_i) = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

where TP (True Positive) is the number of positive instances that are correctly classified. FP (False Positive) is the number of negative instances that are wrongly classified. TN (True Negative) is the number of negative instances that are correctly

classified. FN (False Negative) is the number of positive instances that are wrongly classified. The larger the fitness value is, the more accurate the feature subset is.

3.3 The Neighborhood Search Strategy

As shown in (1), BBPSO is a parameter free algorithm which alleviates the task of setting control parameters in PSO. It shows promising results for single-optimum optimization problems. Due to its special search mechanism, all the particles are prone to converge to one peak quickly. Therefore, BBPSO cannot locate multiple peaks in its single run. In order to extend BBPSO to multimodal optimization problems, a neighborhood search strategy is introduced to BBPSO. Instead of learning from the global best particle, each particle adopts the information from its neighbors. By using the neighborhood search strategy, the swarm would form several stable sub-swarms and each particle will search for better positions in its neighboring area. Therefore, the entire swarm would not converge to one peak.

Let the initial population be $\{x_1, x_2, \dots, x_n\}$ which consists n individuals in the D -dimensional feature space. Before updating the positions of each particle, a similarity matrix is computed to indicate the similarity between any two particles. Each element in the matrix is computed by:

$$S_{ij} = \|x_i - x_j\| \quad (5)$$

where S_{ij} is the i th row and j th column of the similarity matrix. $\|x_i - x_j\|$ denotes the Euclidean distance between the particles x_i and x_j .

For each particle i , find its $nsize$ nearest particles according to the similarity matrix. Then the particle with the highest fitness value among the $nsize$ particles is chosen as the local best for particle i . The position of particle i is updated as follows:

$$x_{id}^{t+1} = N\left(\frac{pbest_{id}^t + lbest_d^t}{2}, |pbest_{id}^t - lbest_d^t|\right) \quad (6)$$

where $lbest$ denotes the local best particle. The neighborhood size $nsize$ is the only parameter in this algorithm. $nsize$ is set as 5 here. If $nsize$ is too large, the algorithm is similar to the global version PSO and it cannot locate multiple optimal feature subsets. If $nsize$ is too small, it may not be able to get enough information from other particles. By using this neighborhood search strategy, each particle would learn from its neighboring particles. Hence, the population would not be attracted to single optimal solution.

As shown in Fig. 1, particle 5 is the global best particle in this generation. If running the canonical BBPSO, other particles in the population would be attracted to fly toward particle 5. In this case, all the particles would converge to one peak and cannot locate other peaks. By using the neighborhood search strategy, each particle would search within its vicinity area by the guidance of its local best and the whole population would form several steady niches. Moreover, an important advantage of this method is that it works well even when the global peaks are not evenly distributed.

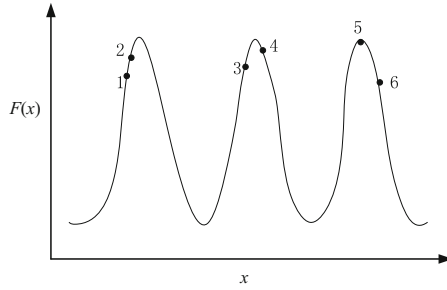


Fig. 1. Illustration of the neighborhood search strategy

3.4 Implementation of the Proposed Method

Based on the descriptions of the proposed algorithm, Algorithm 1 shows the pseudo-code of the proposed approach.

Algorithm 1: Pseudo-code of the proposed BBPSO-NS

1. Divide the dataset into training set and test set;
2. Initialize the position of each particle;
3. Evaluate all the particles in the swarm;
4. Initialize the personal best of each particle;
5. Compute the similarity matrix;
6. **For** each iteration **do**
7. **For** each particle i **do**
8. Choose the local best of particle i ;
9. Update the position of particle i with (6);
10. **End**
11. Evaluate all the particles in the swarm;
12. Update the personal best of each particle;
13. **End**
14. Compute the classification accuracies of optimal feature subsets in the test set;
15. Return the optimal feature subsets and their corresponding classification accuracies.

4 Experiments

4.1 Datasets

In order to testify the effectiveness of the proposed BBPSO-NS, eight datasets from UCI machine learning repository are chosen for comparison, as shown in Table 1. These datasets appear considerable diversity in terms of feature numbers and instances. Before the feature selection process, all the data are normalized to $[0,1]$. For each dataset, all the instances are randomly divided into two parts: 70% are used as the

training set and the rest are used as the test set. In this paper, KNN is employed to calculate the classification accuracy and K is set as 5. The fitness value of each feature subset is calculated through a 10-fold cross validation on the training set. After the training process, the obtained feature subset is testified on the test set with KNN.

Table 1. Datasets

Dataset	# Features	# Instances	# Classes
Glass	9	214	2
Wine	13	178	3
Heart	13	270	2
Australia	16	690	2
Germany	24	1000	2
Ionosphere	34	351	2
Sonar	60	208	2
Musk1	166	476	2

4.2 Parameter Settings and Comparative Algorithms

In order to testify the performance of the proposed BBPSO-NS, four PSO based feature selection methods are used for comparison, including binary PSO (BPSO) [26], binary PSO with catfish effect (BPSO-CE) [27], barebones PSO [19], and competitive swarm optimizer (CSO) [29]. These state-of-the-art feature selection algorithms show promising results and they are often chosen for comparative studies.

The experiments are performed on a computer with Intel(R) Core(TM) i5-6500 at 3.2 GHz and 8.00 GB of RAM and the operating system is MS Windows 10. All the algorithms are coded with MATLAB. For all the five algorithms, the maximum number of iterations is empirically set to be 50 and the population size is set to 20. For BPSO and BPSO-CE, both of the cognitive weight and social weight are set to 2 and the upper and lower bounds of velocity are all set to 6 and -6 , respectively. The time decreasing inertia weights are given by $w_{max} = 0.9$ and $w_{min} = 0.4$ in BPSO and BPSO-CE. For each dataset, all the five methods are repeated 20 independent runs to remove the impact of the random factors.

4.3 Results

Classification Accuracy. This section compares the classification performance of BBPSO-NS and other PSO based feature selection methods. Table 2 shows the mean classification accuracies and the standard deviations in the test set in 20 independent runs. The best mean classification accuracy in each dataset is shown in **boldface**. Moreover, the classification accuracy of KNN in each dataset using all the features is also shown in Table 2 (*i.e.* Without FS).

Table 2 shows that all the feature selection algorithms can achieve better classification accuracies than using all the features. This demonstrates that feature selection is

Table 2. Mean classification accuracies and standard deviations of the five algorithms

Dataset	Without FS	BPSO	BPSO-CE	BBPSO	CSO	BBPSO-NS
Glass	63.08	76.92	74.07	75.05	71.08	76.92
		4.87	7.51	5.81	7.84	4.87
Wine	94.44	97.04	96.67	97.04	97.03	98.34
		1.66	0.78	1.29	2.11	1.37
Heart	81.48	82.96	84.44	83.21	84.2	85.67
		2.03	2.68	4.12	2.16	2.05
Australia	83.62	84.23	83.86	84	84.23	84.53
		0.22	0.96	1.06	1.23	0.96
German	68	72.07	72.29	73.19	72.53	74.45
		2.88	1.79	2.9	1.6	1.2
Ionosphere	79.25	84.91	83.68	84.72	84.72	87.72
		2.038	2.09	2.26	2.84	1.6
Sonar	73.02	72.06	77.1	76.67	77.46	79.05
		3.82	4.19	3	2.88	4.15
Musk1	80.42	83.92	84.92	84.42	83.52	85.31
		2.08	1.61	2.13	2.78	2.04
Average	77.91	81.76	82.13	82.29	81.85	84.0
		2.45	2.7	2.82	2.93	2.28

an effective data pre-processing step in classification problems. Table 2 indicates that BBPSO-NS achieves the highest classification accuracy for all the datasets compared to other four methods. For example, in the Ionosphere dataset, BBPSO-NS gets the mean classification accuracy 87.72, while the second best is 84.91 which is obtained by BPSO. In terms of the average performance in all the eight datasets, BBPSO-NS places the 1st with 84.0 while BBPSO places the 2nd with 82.29. Moreover, BBPSO-NS obtains the lowest standard deviation among the five methods. It can be concluded that the proposed method shows better robustness than other methods. This can be attributed to the neighborhood search strategy used in BBPSO-NS. This method can improve the diversity of the population and prevent the algorithm from being trapped into local optima.

Analysis on the Number of Selected Features. Table 3 shows the number of original features and the number of features selected by the five feature selection methods. For each dataset, the value shown in Table 3 is the mean value of the 20 independent runs. BBPSO-NS obtains the smallest feature subsets in 5 out of 8 datasets. In terms of the average feature subset size, BBPSO-NS places the 1st with the value of 18.29.

Analysis on the Number of Optimal Feature Subsets. The neighborhood search strategy can help the population to form several stable niches and finally locate multiple optimal solutions. Therefore, it is expected that the proposed BBPSO-NS can obtain multiple high quality feature subsets in a single run of the algorithm. In order to demonstrate this, we show the results of a random run of BBPSO-NS in the Wine

Table 3. Number of the selected features

Dataset	All	BPSO	BPSO-CE	BBPSO	CSO	BBPSO-NS
Glass	9	4.6	4	4.14	3.9	4.2
Wine	13	8.6	7.4	7.8	7.8	6.8
Heart	13	8	7.3	7.3	7.9	5.28
Australia	16	6.88	6.29	6.14	6.1	6.27
German	24	10.67	10.29	11.14	11.9	8.27
Ionosphere	34	10.7	10.6	9.4	12.6	6.9
Sonar	60	29.6	30.43	29.8	28.9	27.4
Musk1	166	82.14	85.29	80.43	84.71	82.04
Average	41.59	20.03	20.08	19.52	20.36	18.29

dataset. This dataset contains 13 features. In the training process, the algorithm finds 5 feature subsets with the accuracy of 100 and 4 of them are different. Table 4 shows the IDs of the particles, accuracies in the training set (Acc. (Train)), accuracies in the test set (Acc. (Test)), and the 4 different optimal feature subsets. From Table 4, we can find that the 4 feature subsets with equal accuracies in the training set show quite different performance in the test set. BBPSO-NS can provide users more options. When one feature subset is not suitable, other feature subsets can be adopted immediately.

Table 4. Generated feature subsets and accuracies

ID	Acc. (Train)	Acc. (Test)	Feature subset							
4	100	90.32	3	4	5	7	9	10	11	13
14	100	97.58	1	2	7	8	10	11	13	
16	100	97.58	1	2	7	8	11	13		
19	100	95.16	3	4	7	9	10	11	13	

Analysis on the Computational Time. Table 5 presents the mean computational time of the five methods in 20 independent runs. The shortest computational time for each dataset is marked in **boldface**. CSO shows extremely high computational efficiency. BBPSO-NS needs relatively more computational time than BBPSO. This is due to the neighborhood search strategy in BBPSO-NS. Each particle needs to choose its local leader according to the similarity matrix which costs much computational time. However, BBPSO-NS shows better performance than other methods in terms of classification accuracy and the number of selected features. There is a trade-off between the computational time and the quality of feature subsets.

Table 5. Mean computational time (in seconds)

Dataset	BPSO	BPSO-CE	CSO	BBPSO	BBPSO-NS
Glass	6.63	7.16	3.34	6.19	6.89
Wine	7.41	7.93	4.09	7.69	8.36
Heart	14.3	14.94	8.31	15.64	14.84
Australia	31.88	35.74	16.79	32.3	32.67
German	103.95	111.99	58.18	109.65	113.45
Ionosphere	13.71	12.79	6.57	15.7	13.04
Sonar	6.3	6.98	3.59	6.74	7.31
Musk1	46.1	51.42	28.03	47.95	53.07
Average	28.79	31.12	16.11	30.23	31.2

5 Conclusions

This paper proposes a novel feature selection method based on BBPSO with a neighborhood search strategy. BBPSO is a simplified version of PSO which possesses powerful global search ability. On the basis of BBPSO, a neighborhood search strategy is employed to form several steady sub-swarms and each particle can search for high quality feature subsets in its vicinity area. Furthermore, this strategy can improve the population diversity of BBPSO and prevent the algorithm from being trapped into local optima. To verify the proposed BBPSO-NS, four state-of-the-art PSO based feature selection methods are used for comparison on eight UCI datasets. Experimental results show the superiority of BBPSO-NS over other comparative methods in terms of classification accuracy and the number of selected features. Moreover, BBPSO-NS can obtain multiple high quality feature subsets in a single run of the algorithm. In the future, we will investigate effective local mutation operator to improve the local exploitation ability of the sub-swarms. Another perspective is to apply the proposed approach in high-dimensional datasets to testify its effectiveness.

Acknowledgement. This work was supported by the Natural Science Foundation of Jiangsu Province under Grant No. BK20160898 and the NUPTSF under Grant No. NY214186.

References

1. Liu, H., Yu, L.: Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans. Knowl. Data Eng.* **17**, 491–502 (2005)
2. Dash, M., Liu, H.: Feature selection for classification. *Intell. Data Anal.* **1**(4), 131–156 (1997)
3. Pal, M., Foody, G.M.: Feature selection for classification of hyperspectral data by SVM. *IEEE Trans. Geosci. Remote Sens.* **48**(5), 2297–2307 (2010)
4. Su, C.T., Lin, H.C.: Applying electromagnetism-like mechanism for feature selection. *Inf. Sci.* **181**(5), 972–986 (2011)
5. Lipo, W., Nina, Z., Feng, C.: A general wrapper approach to selection of class dependent features. *IEEE Trans. Neural Netw.* **19**, 1267–1278 (2008)

6. Chakraborty, D., Pal, N.R.: A neuro-fuzzy scheme for simultaneous feature selection and fuzzy rule-based classification. *IEEE Trans. Neural Netw.* **15**, 110–123 (2004)
7. Xu, L., Hung, E.: Distance-based feature selection on classification of uncertain objects. In: Wang, D., Reynolds, M. (eds.) *AI 2011. LNCS (LNAI)*, vol. 7106, pp. 172–181. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25832-9_18
8. Gheyas, I.A., Smith, L.S.: Feature subset selection in large dimensionality domains. *Pattern Recognit.* **43**, 5–13 (2010)
9. Xue, B., Zhang, M., Browne, W.N.: Particle swarm optimisation for feature selection in classification: novel initialisation and updating mechanisms. *Appl. Soft Comput.* **18**, 261–276 (2014)
10. Moradi, P., Gholampour, M.: A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy. *Appl. Soft Comput.* **43**, 117–130 (2016)
11. Ghaemi, M., Feizi-Derakhshi, M.R.: Feature selection using forest optimization algorithm. *Pattern Recognit.* **60**, 121–129 (2016)
12. Li, X., Epitropakis, M., Deb, K., et al.: Seeking multiple solutions: an updated survey on niching methods and their applications. *IEEE Trans. Evol. Comput.* **21**(4), 518–538 (2017)
13. Mahfoud, S.W.: Niching methods for genetic algorithms. Ph.D. dissertation, Department of Computer Science, University Illinois Urbana-Champaign, Urbana (1995)
14. Dejong, K.A.: An analysis of the behavior of a class of genetic adaptive systems. Ph.D. thesis, University of Michigan, Ann Arbor, MI, USA (1975)
15. Cioppa, A.D., Stefano, C.D., Marcelli, A.: Where are the niches? Dynamic fitness sharing. *IEEE Trans. Evol. Comput.* **11**(4), 453–465 (2007)
16. Petrowski, A.: An efficient hierarchical clustering technique for speciation. Technical report, Institution of Nat. Telecommunication, Evry, France (1997)
17. Petrowski, A.: A clearing procedure as a niching method for genetic algorithms. In: *Proceedings on the 3rd IEEE Congress of Evolutionary Computation*, Nagoya, Japan, pp. 798–803 (1996)
18. Kamyab, S., Eftekhari, M.: Feature selection using multimodal optimization techniques. *Neurocomputing* **171**, 586–597 (2016)
19. Kennedy, J.: Bare bones particle swarms. In: *Proceedings on 2003 IEEE Swarm Intelligence Symposium*, Indiana, USA, pp. 80–87. IEEE (2003)
20. Zhang, Y., Gong, D., Hu, Y., Zhang, W.: Feature selection algorithm based on bare bones particle swarm optimization. *Neurocomputing* **148**, 150–157 (2015)
21. Qiu, C.: Bare bones particle swarm optimization with adaptive chaotic jump for feature selection in classification. *Int. J. Comput. Intell. Syst.* **11**(1), 1–14 (2018)
22. Epitropakis, M.G., Li, X., Burke, E.K.: A dynamic archive niching differential evolution algorithm for multimodal optimization. In: *Proceedings on IEEE Evolutionary Computation Congress*, Cancún, Mexico, pp. 79–86. IEEE (2013)
23. Li, X.: Multimodal function optimization based on fitness-Euclidean distance ratio. In: *Proceedings on Genetic Evolutionary Computing Conference*, pp. 78–85. ACM, London, U. K. (2007)
24. Zhai, Z., Li, X.: A dynamic archive based niching particle swarm optimizer using a small population size. In: *Proceedings on Australasian Computer Science Conference*, Perth, Australia, pp. 1–7 (2011)
25. Li, X.: Niching without niching parameters: Particle swarm optimization using a ring topology. *IEEE Trans. Evol. Comput.* **14**(1), 150–169 (2010)
26. Qu, B.Y., Suganthan, P.N., Das, S.: A distance-based locally informed particle swarm model for multimodal optimization. *IEEE Trans. Evol. Comput.* **17**(3), 387–402 (2013)

27. Kennedy, J., Eberhart, R.C.: A discrete binary version of the particle swarm algorithm. In: Proceedings on the 1997 Systems Man and Cybernetics Conference, pp. 4104–4108. IEEE (1997)
28. Chuang, L.Y., Tsai, S.W., Yang, C.H.: Improved binary particle swarm optimization using catfish effect for feature selection. *Expert. Syst. Appl.* **38**, 12699–12707 (2011)
29. Gu, S., Cheng, R., Jin, Y.: Feature selection for high-dimensional classification using a competitive swarm optimizer. *Soft. Comput.* **22**(3), 811–822 (2018)



The Chinese Postman Problem Based on the Probe Machine Model

Jing Yang¹(✉), Zhixiang Yin¹, Jianzhong Cui¹, Qiang Zhang^{1,2},
and Zhen Tang¹

¹ School of Mathematics and Big Data,
Anhui University of Science and Technology, Huainan, Anhui, China
jyangh82@163.com

² School of Computer, Dalian University of Technology,
Dalian, Liaoning, China

Abstract. The probe machine model is a new computational model. Its computation depends only on DNA molecules, and it is a parallel computing model from the bottom. Its data placement is nonlinear and combined with the parallelism of biochemical reactions, which greatly improves the effective computing power of the model. The Chinese postman problem is a NP complete problem in combinatorial optimization. In this paper, we will try to solve this problem by using the probe machine model, so as to improve the effectiveness of the problem computation. The postman problem is corresponded to a connected graph G . According to the vertex construct data cell and connection probe, and put it into the computing platform. After the calculation, the solution of the problem is detected by the detector. It is the smallest weight sum of the generalized Euler closure. This model shows the superiority and versatility of the probe machine model compared with other computing models.

Keywords: Chinese postman problem · Probe machine
NP-complete problem · Data fiber

1 Introduction

With the development of bionic computing, various computing models emerge as the times require. The generation of the probe machine model has subverted people's understanding of the existing computing models. The probe machine is a tool for detecting certain substances, and its concept derives from such fields as biology, computer science, electronics, information security, archaeology, and so on. In 2016, Xu [1] and his research team reported major breakthrough article "Probe Machine" in the journal of IEEE Transactions on Neural Networks & Learning Systems. His paper presented this computing model beyond Turing machine [2] that is called the probe machine. It is computing model from the underlying whole of parallel. It has only undergone a biological operation to find all solutions of problems for the NP-complete problem, such as Hamilton Problem, Vertex graph coloring problem, and so on [1]. The placement of its data is nonlinear and combined with the parallelism of biochemical reactions, which greatly improves the effective computing power of the model. Its data

placement mode is free space placement mode, and any one pair of data can be directly processed. Professor Xu Jin pointed out that it surpassed the traditional DNA computing model. All NP-complete problems based on the Turing machine are equivalent in polynomial time, which means that it has no NP-completely problem puzzled mankind in the probe model.

2 The Chinese Postman Problem

The Chinese postman problem is an important issue in operations research. The Chinese postman problem is the mail delivery problem of the postman in a certain area. The postman starts from the post office every day, and walks all the streets of the area. Lastly he returns to the post office. The question is how he should arrange the route of sending the letter to the shortest route. This problem was first proposed by Chinese scholar Guan Meigu in 1960, and given the method—"the work method on the odd and even point map" [3, 4], which is called "the problem of Chinese postman" by the international [4]. In 1973, Hungarian mathematicians Edmonds and Johnson provided an effective algorithm for Chinese postman problem [5]. It is also called the arc route problem [6, 7]. It has a wide range of applications in public utilities [8]. So many scholars have studied its algorithm [9–13]. But there are no more effective algorithms at present.

The Chinese postman problem can be abstracted as a problem in the graph theory model. For a given connected undirected graph $G = (V, E, \omega)$, $V = \{v_1, v_2, \dots, v_n\}$ is the set of vertices of the G . $E = \{e_1, e_2, \dots, e_m\}$ is the set of arcs of the G , and $\omega = \{\omega_1, \omega_2, \dots, \omega_m\}$ is the set of arc's weight of the G . $|V| = n$, $|E| = m$. The problem requires a loop to pass at least once each side and it satisfies the minimum total weight. That is to search the smallest weight sum of the generalized Euler closure.

3 The Probe Machine Model

The bioreactor principle of the probe machine model is also molecular self-assembly. The so-called molecular self-assembly is the process of spontaneously forming thermodynamic stable, structurally defined, specific aggregates or supramolecular structures by non covalent bonds under the equilibrium conditions, such as molecular and nanoparticle. The molecular self-assembly is a process of self-improvement and self-improvement from simple to complex, from disorder to order, from multiple components to single components. It is also a highly organized, highly ordered, structured, functional and informational complex system. The probe model also combines these advantages, but it is different from the common biological probe: it is an abstract concept designed to associate two data.

The probe machine is composed by the database, the probe libraries, data controller, the controller probe, probe computing, computing platform, the detector, the true solution, and residual memory support recycling. Record separately $X, Y, \sigma_1, \sigma_2, \tau, \lambda, \eta, Q, C$. The molecular self-assembly refers to molecule and nanoparticle as structural units in the equilibrium conditions, through non covalent interactions

spontaneously formed in the process of associating special aggregates or super-molecular structure determine the structure of thermodynamically stable, on and on. Molecular self-assembly is a process of continuous self modification and self perfection from simple to complex, from disorder to order, and from multiple components to a single component. It is a highly organized, highly ordered, structured, functional and information-based complex system. The probe machine model also incorporates these advantages, but it differs from the usual biological probes: it is an abstract concept designed to correlate two data (Fig. 1).

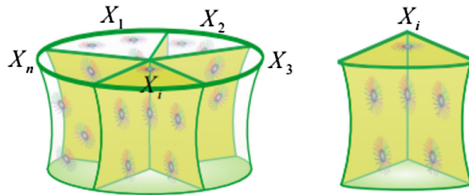


Fig. 1. The data base of the probe machine.

The database of probe machine consists of n data pools, each of which contains large amounts of x_i elements. Each x_i consists of data cells and data fibers. The data cell has only one, and the data fiber has p_i kinds, as shown in Fig. 2.

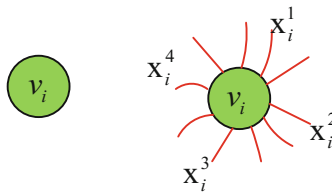


Fig. 2. Data cell and data fiber.

4 Case Simulation

The problem is that a postman delivers letters within the jurisdiction of nine streets. The streets are two-way traffic. And the distance between each street has been given. How do we choose the shortest way to get at least once in every street of the jurisdiction? This example can be abstracted as a problem of graph theory. That is, there is a connected graph $G = \langle V, E \rangle$. The vertex set is $V = \{v_0, v_1, v_2, v_3, v_4, v_5, v_6\}$. The arc set is $E = \{e_{01}, e_{12}, e_{23}, e_{34}, e_{24}, e_{45}, e_{56}, e_{15}, e_{06}\}$, where e_{ij} represents the arc between the v_i vertex and the v_j vertex. The arc weight set of the graph G is $\omega = \{\omega_{01}, \omega_{12}, \omega_{23}, \omega_{34}, \omega_{24}, \omega_{45}, \omega_{56}, \omega_{15}, \omega_{06}\} = \{1, 2, 2, 2, 2, 1, 3, 1, 2\}$. Where ω_{ij} corresponds to the weight of arc e_{ij} , and because of the undirected property of bidirectional streets and graphs, e_{ij} and e_{ji} denote the same arc, that is, $\omega_{ij} = \omega_{ji}$. Thus

the Chinese postman problem is transformed into finding a generalized Euler closure in graph G, as shown in Fig. 3.

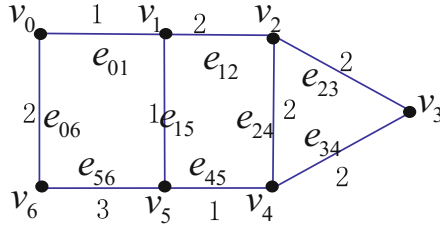


Fig. 3. The jurisdiction of nine streets

Under the probe machine model, the graph G is a simple connected graph. The set $V(G)$ represents the vertex set of the graph G, and the set $E(G)$ represents the arc set of the graph G. That is, $E(G) = \{V_i V_j \mid V_i V_j \in E(G), i, j = 1, 2, \dots, n\}$, and $E^2(V_i) = \{V_l V_j V_k \triangleq x_{ilj}, i \neq j, l, l \neq j\}$. V_l, V_j with V_i are the vertices of the arc, and it is a set of two long roads centered on V_i . On the basis of $E^2(V_i)$, the database X of connecting probe machine is

$$X = \cup_{i=1}^n E^2(V_i) = \cup_{i=1}^n \{x_{ilj} \mid v_l, v_j \in \Gamma(v_i); i \neq j, l; l \neq j\}.$$

Each of these data x_{ilj} has two data fibers, x_{ilj}^i and x_{ilj}^j respectively.

4.1 Basic Algorithm

- Step 1: Search all the generalized Euler closures of the diagram G corresponding to the Chinese postman problem;
- Step 2: Keep those the generalized Euler closures from the fixed vertex of G and back to the fixed vertex;
- Step 3: Keep the generalized Euler closures at least after all arcs;
- Step 4: Find the generalized Euler closure of the weight sum minimum, and then find the solution of the Chinese postman problem.

4.2 Probe Model

The probe is a small segment of single strand DNA or RNA fragment (about 20 bp to 500 bp), which is used to detect nucleic acid sequences complementary to them. The database consists of N data pools, each consisting of data cells and data fibers, with only one data cell and a data fiber, as shown in Fig. 4.

Fiber length can be used to represent weight. V_i is used as nanoparticle, and the arc connected to V_i is used as data fiber to construct data cell. The structure of the data fiber is divided into three parts. The first part is connected with the data cell. The second part is the arc weight, the DNA number of the arc is [the arc weight *5] (if the arc weight

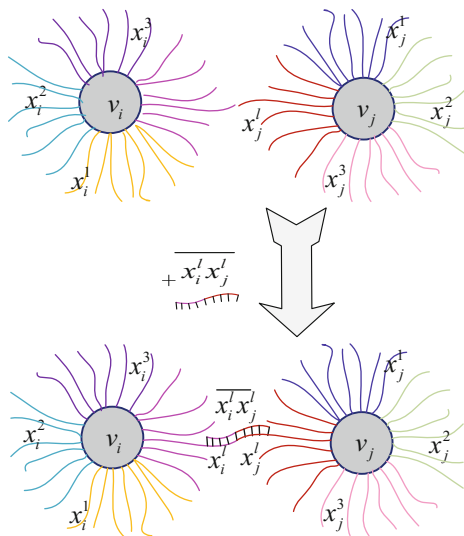


Fig. 4. Connecting probe.

value is larger also can be expressed by [the weight value/10] etc.). The third part is used to construct the probe, as shown in Fig. 5.

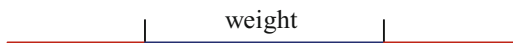


Fig. 5. Data fiber with weight.

The connecting probe is not directionality. $\overline{x_i^l x_j^l}$ and $\overline{x_j^l x_i^l}$ are the same probe. If there is a directed graph, the probe has a direction. $\overline{x_i^l x_j^l}$ and $\overline{x_j^l x_i^l}$ are not the same probe.

Step 1: Build the database

The Chinese postman problem with 7 vertices is abstracted as a weighted connected graph. The vertex set is $V = \{v_0, v_1, v_2, v_3, v_4, v_5, v_6\}$. Building a database of connecting probe is

$$\begin{aligned}
 X &= E^2(v_0) \cup E^2(v_1) \cup E^2(v_2) \cup E^2(v_3) \cup E^2(v_4) \cup E^2(v_5) \cup E^2(v_6), \\
 E^2(v_0) &= \{x_{016}\}, E^2(v_1) = \{x_{102}, x_{105}, x_{125}\}, \\
 E^2(v_2) &= \{x_{213}, x_{214}, x_{234}\}, E^2(v_3) = \{x_{324}\}, \\
 E^2(v_4) &= \{x_{423}, x_{425}, x_{435}\}, E^2(v_5) = \{x_{514}, x_{516}, x_{546}\}, \\
 E^2(v_6) &= \{x_{605}\}.
 \end{aligned}$$

There are 15 kinds of data. So there are 30 kinds of data fibers.

$$\begin{aligned}
\mathfrak{S}(x_{016}) &= \{x_{016}^1, x_{016}^6\}, \mathfrak{S}(x_{102}) = \{x_{102}^0, x_{102}^2\}, \mathfrak{S}(x_{105}) = \{x_{105}^0, x_{105}^5\}, \\
\mathfrak{S}(x_{125}) &= \{x_{125}^2, x_{125}^5\}, \mathfrak{S}(x_{213}) = \{x_{213}^1, x_{213}^3\}, \mathfrak{S}(x_{214}) = \{x_{214}^1, x_{214}^4\}, \\
\mathfrak{S}(x_{234}) &= \{x_{234}^3, x_{234}^4\}, \mathfrak{S}(x_{324}) = \{x_{324}^2, x_{324}^4\}, \mathfrak{S}(x_{423}) = \{x_{423}^2, x_{423}^3\}, \\
\mathfrak{S}(x_{425}) &= \{x_{425}^2, x_{425}^5\}, \mathfrak{S}(x_{435}) = \{x_{435}^3, x_{435}^5\}, \mathfrak{S}(x_{514}) = \{x_{514}^1, x_{514}^4\}, \\
\mathfrak{S}(x_{516}) &= \{x_{516}^1, x_{516}^6\}, \mathfrak{S}(x_{546}) = \{x_{546}^4, x_{546}^6\}, \mathfrak{S}(x_{605}) = \{x_{605}^5, x_{605}^0\}
\end{aligned}$$

7 kinds of nanoparticles (2.5 nm) are produced as 7 data cells, and the DNA sequences corresponding to the 30 types of data fibers are coded, and then the corresponding DNA strands were synthesized. The DNA strand (data fiber) is embedded in the corresponding nanoparticle.

Step 2: Structure probe library

The graph G is an undirected connected graph, so the probe used is a connected probe.

$$\begin{aligned}
Y_{02} &= \{\overline{x_{016}^1 x_{213}^1}, \overline{x_{016}^1 x_{214}^1}\}, \\
Y_{05} &= \{\overline{x_{016}^6 x_{564}^6}, \overline{x_{016}^6 x_{516}^6}, \overline{x_{016}^1 x_{516}^1}, \overline{x_{016}^1 x_{514}^1}\}, \\
Y_{13} &= \{\overline{x_{102}^2 x_{324}^2}\}, \\
Y_{14} &= \{\overline{x_{102}^2 x_{423}^2}, \overline{x_{102}^2 x_{425}^2}, \overline{x_{125}^2 x_{423}^2}, \overline{x_{125}^2 x_{425}^2}, \overline{x_{125}^5 x_{425}^5}\}, \\
Y_{24} &= \{\overline{x_{213}^3 x_{435}^3}, \overline{x_{213}^3 x_{432}^3}\}, \\
Y_{25} &= \{\overline{x_{214}^4 x_{514}^4}, \overline{x_{214}^4 x_{514}^4}\}, \\
Y_{35} &= \{\overline{x_{324}^4 x_{514}^4}, \overline{x_{324}^4 x_{546}^4}\}, \\
Y_{16} &= \{\overline{x_{605}^0 x_{102}^0}, \overline{x_{605}^0 x_{105}^0}, \overline{x_{605}^5 x_{105}^5}\} \\
Y_{46} &= \{\overline{x_{435}^5 x_{605}^5}, \overline{x_{425}^5 x_{605}^5}\}
\end{aligned}$$

There are 9 sub-probe libraries, 23 probes. 23 connection probes are constructed on the basis of step 1. The structure of the probe is on the basis of the Xu Jin's probe principle [1].

Step 3: Generate results

In order to find the generalized Euler closure of the graph G , the probe is put in the computing platform each time, and the strand after each extraction contains the corresponding arc. In Fig. 6, connect the vertex v_0 and the vertex v_2 with a connecting probe $\overline{x_{016}^1 x_{214}^1}$. It can generate DNA fragments $v_6 \rightarrow v_0 \rightarrow v_1 \rightarrow v_2 \rightarrow v_3$ and $v_6 \rightarrow v_0 \rightarrow v_1 \rightarrow v_2 \rightarrow v_4$. In this way, we obtain the generalized Euler closure. Through the detection technology, under the AFM (electron microscope), we can detect

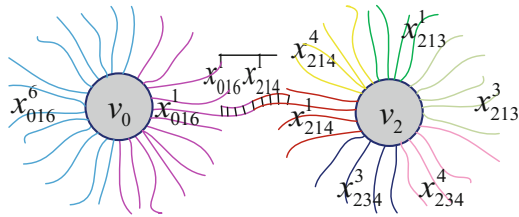


Fig. 6. The connecting probe connects the vertex v_0 and the vertex v_2 .

the smaller order polymer. That is, the solution is that we want to find. And read out the solution through the reading technology of the solution. In this example, we get the postal route with a weight of 19.

5 Conclusion and Analysis

This paper uses the probe machine model to solve the Chinese postman problem. The probe machine model is a parallel computing model from the bottom and it has strong parallelism. To deal with the problem of the Chinese postman problem, after probe operation we can obtain all possible solutions. For the generalized Chinese postman problem, it can be abstracted as digraph or probe machine model. It is a new computing mode, and greatly reduces the complexity of the operation process. It only needs one step or limited step calculation to get the solution of the problem. However, from the current detection technology, the detection of the solution (polymer rapid reading) has some limitations. With the development of molecular biology technology, it is believed that the nano probe machine proposed by Professor Xu Jin will be realized in the near future.

Acknowledgment. This project is supported by National Natural Science Foundation of China (No. 61702008, No. 61672001) and Anhui Provincial Natural Science Foundation (No. 1808085MF193).

References

1. Xu, J.: Probe machine. *IEEE Trans. Neural Netw. Learn. Syst.* **27**(7), 1405–1416 (2016)
2. Turing, A.M.: On computable numbers, with an application to the entscheidungsproblem. A correction. In: *Alan Turing His Work and Impact*, **s2-42**(1), pp. 13–115 (2013)
3. Guan, M.G.: Operation method of odd even point diagram. *J. Math.* **10**(3), 263–266 (1960)
4. Guan, M.G.: A historical review for the research and development of Chinese postmen. *J. Oper. Res.* **19**(3), 1–7 (2015)
5. Edmonds, J.: The Chinese postman problem. *Oper. Res.* **13**(Suppl.), 1–73 (1965)
6. Eiselt, H.A., Gendreau, M., Laporte, G.: Arc routing problems, part 1: the Chinese postman problem. *Oper. Res.* **43**, 231–242 (1965)
7. Eiselt, H.A., Gendreau, M., Laporte, G.: Arc routing problems, part 2: the rural postman problem. *Oper. Res.* **43**, 399–414 (1965)

8. Stricker, R.: Public sector vehicle routing: the Chinese postman problem. Massachusetts Institute of Technology (1970)
9. Gordenko, M.K., Avdoshin, S.M.: The mixed Chinese postman problem. Труды ИСП РАН **29**(4) (2017)
10. Gutin, G., Jones, M., Sheng, B.: Parameterized complexity of the k -arc Chinese postman problem. J. Comput. Syst. Sci. **84**, 107–119 (2017)
11. Gutin, G., Jones, M., Sheng, B.: Parameterized complexity of the k -arc Chinese postman problem. In: Schulz, A.S., Wagner, D. (eds.) ESA 2014. LNCS, vol. 8737, pp. 530–541. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44777-2_44
12. Han, A.L., Zhu, D.M.: DNA computing model based on a new scheme of encoding weight for Chinese postman problem. J. Comput. Res. Dev. **44**(6), 1053–1062 (2007)
13. Li, W., Wang, L.: DNA calculation of Chinese postman problem. J. Comput. Appl. **29**(7), 1880–1883 (2009)



Research on Pulse Classification Based on Multiple Factors

Zhihua Chen¹, An Huang², and Xiaoli Qiang¹(✉)

¹ Institute of Computing Science and Technology, Guangzhou University,
Guangzhou 510006, Guangdong, China

qiangxl@gzhu.edu.cn

² School of Automation, Huazhong University of Science and Technology,
Wuhan 430074, Hubei, China

Abstract. Pulse diagnosis is an important part of the theoretical system of traditional Chinese medicine, but is subject to the doctor's subjective assumptions and other factors and is difficult to teach. Therefore, to achieve objective and accurate pulse classification and further improve pulse diagnosis and treatment, this paper presents a pulse classification method based on multi-factor analysis. Pulse wave data were collected from each person in the static state, after which the cosine similarity theorem and principal components analysis were used to identify the pulse type after extracting the characteristics of the pulse waveform. Compared with previous methods, this method has the advantages of high recognition rate, comprehensive pulse classification, and inclusion of multiple factors. This method has been proven to be a good reference for digitalization, visualization, and automatic diagnosis of pulse in Chinese medicine.

Keywords: Pulse classification · Feature extraction · Cosine theorem
Principal components analysis

1 Introduction

With the aging of the Chinese population and the development of medical technology, traditional Chinese medicine (TCM) is gradually attracting attention. The traditional diagnostic process in Chinese medicine mainly relies on the doctor's own practical experience and the feeling when cutting off the pulse and can be influenced by personal subjective judgment [1]. Instead, modern advanced scientific equipment can be used to extract characteristic signals from the human body, and effective signal analysis methods can be combined to achieve scientific, intuitive, and accurate extraction of the pulse signal's characteristic parameters [2]. The extracted information can then be analyzed and studied to determine the health status of the human body, which is an important goal of incorporating digitalization and intelligence into TCM diagnosis.

In recent years, the most common methods used in pulse wave analysis have been time domain analysis, frequency domain analysis, and time-frequency analysis [3]. Luo et al. related changes in waveform area to the presence of cardiovascular disease in human beings, but this relationship has a certain sensitivity [4]. Some researchers have

developed classifications using the pulse-wave slope threshold, but the method can identify only some kinds of pulse waves well [5]. In addition, Sugawara et al. used an N-point moving average method to carry out pulse-wave identification, but their method had many problems, such as not effectively identifying various waveforms or incorrect recognition [6].

In this research, the pulse wave characteristics were extracted by time-domain analysis and wavelet analysis in accordance with the eight kinds of pulse conditions defined in the “near lake vein” [7]. The cosine similarity theorem and principal components analysis were used to classify the eigenvalues of each quantization. At present, four pulse patterns (wiry, soft, smooth, and unsmooth) can be accurately identified with a recognition rate of 92.5%. The main purpose of this paper is to provide intelligent algorithm support for the remote diagnosis and treatment of Chinese medicine, and provide a preliminary reference for patients.

2 Selection of Pulse Type and Classification Principle

2.1 Selection of Pulse Type

According to the viewpoint of ancient Chinese medicine on the pulse and referring to current pulse research, the pulse can be divided according to various characteristics into many categories [8]. This paper addresses the identification of four common kinds of pulse; Table 1 shows their names and codes.

Table 1. Four kinds of pulse and codes.

Pulse condition	Wiry pulse	Soft pulse	Smooth pulse	Unsmooth pulse
Code	Y_1	Y_2	Y_3	Y_4

2.2 Pulse Classification Principle

According to the viewpoint of traditional Chinese medicine on the definition of each type of pulse, the finger is transformed into a measurement index [9]. Among these four types of pulse, an unsmooth pulse means arrhythmia, in which the pulse rate may be fast, but may also be relatively slow; a soft pulse shows a faster pulse rate and smaller amplitude; a smooth pulse shows a higher diastolic wave and a short peak time, and so on [10]. Combining the definition of each waveform and the pulse waveform chart and using theoretical analysis and experimental verification, a pulse chart can be constructed by extracting the characteristic features shown in Table 2.

3 Study of Pulse Classification Methods

3.1 Cosine Similarity Theorem

In this research, after the characteristics were extracted by time-domain and wavelet analysis, the extracted characteristic index was compared with the standard index of the

four kinds of pulse in Fig. 1 to evaluate the pulse type. The concrete method uses the cosine similarity theorem. A similarity measure is used to calculate the degree of similarity of multiple individuals. The smaller the measurement, the lower is the degree of similarity [11].

Table 2.

Sequence	Features	Significance	Extraction method
1	Frequency	Heartbeat speed	Domain analysis
2	Crest number	Number of peaks per cycle	Domain analysis
3	Elastic index	Elasticity of blood vessels	Domain analysis
4	Resistance index	Smooth blood condition	Domain analysis
5	Amplitude	Amplitude of the pulse wave	Domain analysis
6	Waveform area	Peripheral resistance	Domain analysis
7	Main wave	Systolic time	Frequency domain analysis
8	Dicrotic wave	Diastolic time	Frequency domain analysis

In the triangular vector representation, if the vector a is (x_1, y_1) , and the vector b is (x_2, y_2) , then the cosine theorem can be represented as the form shown in Fig. 1.

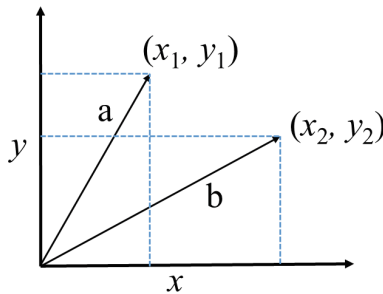


Fig. 1. Sketch of the cosine similarity theorem.

The cosine of the angle between vectors a and b is calculated as shown in Eq. (1):

$$\cos(\theta) = \frac{a * b}{\|a\| \times \|b\|} = \frac{(x_1, y_1) * (x_2, y_2)}{\sqrt{x_1^2 + y_1^2} \times \sqrt{x_2^2 + y_2^2}}. \tag{1}$$

Equation (1) can be extended so that if the vectors a and b are not two-dimensional, but n-dimensional, the cosine of the calculation is still correct. Assume that a and b are two n-dimensional vectors and that a is $(x_1, x_2, x_3 \dots)$, and b is $(y_1, y_2, y_3 \dots)$; then the cosine of the angle between a and b is given in Eq. (2):

$$\cos(\theta) = \frac{a * b}{\|a\| \times \|b\|} = \frac{\prod_{i=1}^n (x_i, y_i)}{\prod_{i=1}^n \sqrt{x_i^2 + y_i^2}}. \tag{2}$$

In this research, the standard waveforms of four kinds of pulse shapes (wiry, soft, smooth, and unsmooth) were constructed, and the characteristic set of each was extracted, as shown in Table 3 (0 means that the parameter could not be calculated by definition).

Table 3. Characteristic waveform parameters for the four types of pulse.

Pulse wave	Time domain						Frequency domain	
	Frequency	Crest number	Elastic index	Resistance index	Amplitude	Waveform area	Main wave	Dicrotic wave
Y_1	67.136	2.900	0.234	0.386	80.919	0.410	0.147	0.433
Y_2	81.608	2	0	0	91.30	0.380	0.208	0.595
Y_3	66.677	3.360	0.298	0.470	111.90	0.356	0.142	0.427
Y_4	90.459	1.900	0	0	57.496	0.438	0.192	0.507

However, due to the uncertainty of each feature, the parameter weights for the various pulse characteristics must be determined according to the degree of discreteness of the characteristic parameters extracted from different pulse signals. The weighting factor when using the cosine similarity theorem is determined primarily by the standard deviation of the individual characteristics of the four standard waveforms. Therefore, the calculation of the cosine similarity theorem after weighting proceeds as follows:

- (1) Normalization of characteristic parameters. Each characteristic parameter is changed from 0 to 1, and the error caused by the larger characteristic parameter is reduced.

Let a total of q characteristic parameters be defined as $z_i = (z_{i1}, z_{i2}, z_{i3}, z_{i4}), i = 1, 2, \dots, q$, so that the maximum value of z_i is max and the minimum value is min. The new result is $z_i^* = (z_{i1}^*, z_{i2}^*, z_{i3}^*, z_{i4}^*)$ obtained according to the normalization definition. The specific normalization is defined as in Eq. (3):

$$z_i^* = \frac{z_i - \min}{\max - \min}. \tag{3}$$

- (2) Finding the degree of dispersion and determining the weighting factor. According to the newly generated normalized data, the standard deviation is used to determine the degree of discretization of each feature.

The mean \bar{z}_i^* of q^*z is calculated, and then the weighting factor d is calculated as shown in Eq. (4):

$$d_i = \sum_{j=1}^4 |z_i^* - \bar{z}_i^*|. \tag{4}$$

- (3) Weighted cosine similarity calculation. The value of the weighted cosine similarity is determined based on the determined weighting coefficients a_i for each feature. The concrete calculation method is shown in Eq. (5):

$$\cos(\theta) = d * \frac{a * b}{\|a\| * \|b\|} = d_i * \frac{\prod_{i=1}^n (x_i, y_i)}{\prod_{i=1}^n \sqrt{x_i^2 + y_i^2}} \tag{5}$$

According to the above definition of the weighted cosine calculation step, the weighting coefficient matrix calculated by the cosine similarity is: $d = (1.608, 1.616, 1.785, 1.821, 1.191, 1.365, 1.168, 1.440)$. The average pulse wave characteristic values of the sample values are combined with Eq. (5) to calculate the cosine similarity of the test sample pulse wave characteristic value and the average value. The pulse type is then chosen according to the maximum value, which represents the type of the pulse wave.

3.2 Principal Components Analysis

The cosine similarity theorem has many parameters, and the default weights of all parameters are equal [12, 13]. However, in real-world classification, a greater number of characteristics will disturb the demarcation of classification standards, and the importance of each index may also vary. Therefore, a method is needed that can automatically weight or remove effects depending on the importance of particular features [14, 15]. Therefore, this paper also uses principal components analysis for examining the data to achieve this goal.

After the characteristic of the waveform data were extracted, principal components analysis of the feature matrix was performed. The detailed steps in the analysis were as follows:

- (1) Feature centralization. This involves subtracting the mean from each of the data values. ‘‘One dimension’’ represents a feature, and the mean value after transformation is 0. If the p -dimensional stochastic vector after the original data normalization is $x = (x_1, x_2, \dots, x_p)^T$, with n samples, $n > p$, a sample matrix of size $p * n$ can be constructed as shown in Eq. (6):

$$x = (x_{i1}, x_{i2}, \dots, x_{ip})^T, i = 1, 2, \dots, n. \tag{6}$$

The sample matrix is subjected to the normalized transformation shown in Eq. (7):

$$z_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}, i = 1, 2, \dots, n; j = 1, 2, \dots, p, \tag{7}$$

where $\bar{x}_j = \frac{\sum_{i=1}^n x_{ij}}{n}$, $s_j^2 = \frac{\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2}{n-1}$, and the standardized array Z is obtained.

- (2) Calculating the correlation coefficient matrix C of the normalized matrix Z. As shown in Eq. (8):

$$C = [c_{ij}]_{p \times p} = \frac{Z^T Z}{n - 1} \tag{8}$$

- (3) Calculating the eigenvalues and eigenvectors of the covariance matrix C. As shown in Eq. (9):

$$|C - \lambda I_p| = 0. \tag{9}$$

After the p-valued roots have been obtained, the principal components are determined. The value of m is determined so that $\frac{\sum_{j=1}^m \lambda_j}{\sum_{j=1}^p \lambda_j} \geq 0.85$ and the comprehensive utilization of information can be greater than 85%. Here, the value 0.85 is determined according to the need, with a greater value indicating a smaller information loss. For each $\lambda_j, j = 1, 2, \dots, m$, the unit characteristic vector b_j is obtained by solving Eq. (10):

$$(C - \lambda_j E) * b_j = 0. \tag{10}$$

- (4) According to the feature values, the feature vectors are selected from large to small, and the new data sets are obtained and transformed into principal components. As shown in Eq. (11):

$$U_i = Z_i^T b_j, j = 1, 2, \dots, m, \tag{11}$$

Where U_i is the important principal component for section i, $i = 1, 2, \dots$

Principal components analysis is based on examining sample data by analyzing existing samples to determine the components and the corresponding thresholds. Data are then extracted from other samples for validation. This method needs a large amount of data as a basis, depending on the actual data characteristics. In this research, data from 40 people were selected as a sample for analysis, and data from 40 other people were selected to verify the effect. According to steps described above, eight coefficients were obtained, which from large to small were: 741.074, 223.412, 0.480, 0.010, 0.002,

0.00074, and 0.00007. Because the value of m was determined using $\frac{\sum_{j=1}^m \lambda_j}{\sum_{j=1}^p \lambda_j} \geq 0.85$,

m was determined as 2, and the contribution rate was greater than 99.5%. The eigenvectors could be reconstructed corresponding to the two previous eigenvalues:

$\lambda_1 = 741.074$, corresponding to the feature vector $\beta_1 = (-.2579 \ 0.0871 \ -0.0259 \ -2.649e-04 \ 0.8661 \ -0.01282 \ -0.01916 \ 0.03224)$, and $\lambda_2 = 223.412$, corresponding to the feature vector $\beta_2 = (-0.8659 \ 0.01710 \ 0.03484 \ 0.01272 \ -0.2580 \ -0.01736 \ -0.03867 \ 0.01905)$. The first and second principal components were obtained using the β_1 and β_2 vectors, and the classification criteria were determined accordingly, as shown in Fig. 2.

Through the data analysis described above, it is clear that the first and second principal components are not sufficient to distinguish effectively among the four pulse types. Therefore, the first and second principal components were used to define straight lines that could be used to distinguish the pulse types. The classification procedure is as follows:

- (1) If the feature point is above $y_1 = -0.6792 * x_1 - 72.01$ (blue line in the figure), proceed to step 2. If it is below the line, proceed to step 3.
- (2) If the feature point is above $y_2 = 0.475 * x_2 + 129.1$ (red line in the figure), then the pulse wave is a smooth pulse, otherwise it is a wiry pulse.
- (3) If the feature point is above $y_3 = 1.85 * x_3 + 147.3$ (black line in the figure), then the pulse wave is an unsmooth pulse, otherwise it is a soft pulse.

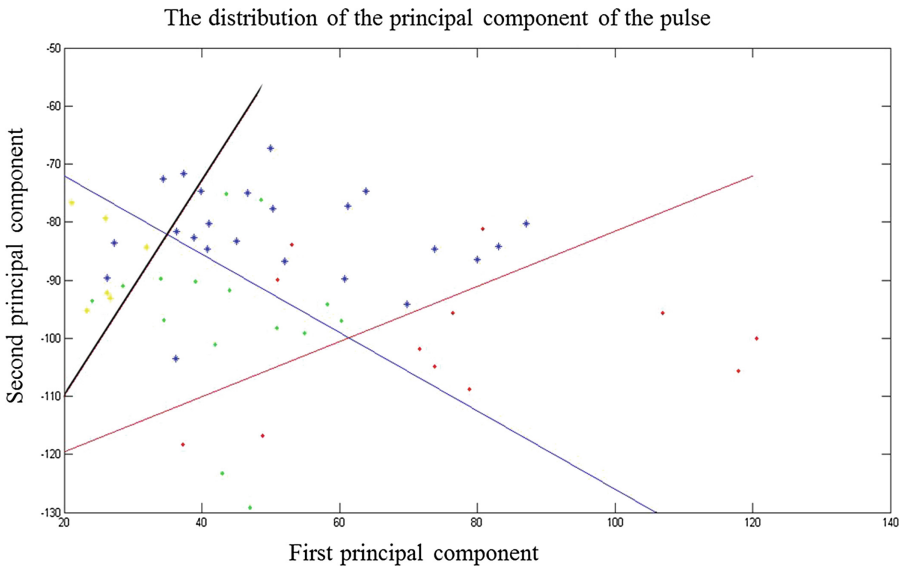


Fig. 2. Distribution of principal components for the four kinds of pulse. (Color figure online)

4 Analysis of Results

4.1 Cosine Similarity Results

After the preceding theoretical discussion, actual data were needed to verify the effect. The validation used 80 data samples, of which 52 were men and 28 women between

the ages of 28 and 40. Among them, there were 20 cases each of the four kinds of pulse (wiry, soft, smooth, and unsmooth). Table 4 shows the statistical recognition rates.

Table 4. Use of the cosine similarity theorem for correct recognition of the four kinds of pulse.

Name	Number	Number in sample	Number identified	Recognition rate
Wiry pulse	X01-X20	20	18	90%
Soft pulse	R21-R40	20	17	85%
Smooth pulse	H41-H60	20	19	95%
Unsmooth pulse	S61-S80	20	17	85%
Total		80	71	88.75%

The average similarity rates can reach 88.75% by using the cosine similarity theorem.

4.2 Principal Component Analysis Results

To verify the principal components analysis, the same 80 data samples were used. However, 40 of these data samples were used to determine the demarcation criteria for principal components analysis, and the remaining 40 samples were used to validate the analysis. Table 5 gives the statistical rates of correct recognition for the four pulse types.

Table 5 Principal components analysis: statistical recognition rates of the four kinds of pulse. Table 5 shows that principal components analysis can achieve an average rate of similarity as high as 92.5% between the test and the TCM evaluation.

Table 5. Principal components analysis: statistical recognition rates of the four kinds of pulse.

Name	Number	Number in sample	Number identified	Recognition rate
Wiry pulse	X11-X20	10	9	90%
Soft pulse	R31-R40	10	9	90%
Smooth pulse	H51-H60	10	10	100%
Unsmooth pulse	S71-S80	10	9	90%
Total		40	37	92.5%

5 Conclusions

In this paper, efforts have been made to identify extracted eigenvalues by means of the cosine similarity theorem and principal components analysis, and two classification methods were selected to verify the experiment. The average accuracies of the two

methods were 88.75% and 92.5%. There is a certain error in diagnosis by doctors in traditional Chinese medicine, but the results of these experiments were similar to those obtained by doctors. Therefore, these two methods have important significance for pulse diagnosis in general and as a reference for traditional Chinese medicine pulse diagnosis.

Acknowledgement. The authors are grateful for the support from the National Nature Science Foundation of China (61632002, 61379059, and 61572046), and the Natural Science Foundation of Guangdong Province of China (2018A030313380).

References

1. Wang, Y., Chang, C.C., Chen, J.C., et al.: Pressure wave propagation in arteries. *IEEE Eng. Med. Biol. Mag.* **16**(1), 51–56 (1997)
2. Liu, R.: New feature extraction and classification of wrist pulse. East China University of Science and Technology (2010, in Chinese)
3. Wang, A.M., Zhang, W.L.: Classification study of TCM pulse diagrams based on fuzzy attribute syntax. In: *Proceedings of China Biomedical Engineering*, pp. 333–334 (1987). in Chinese
4. He, S.D., Luo, Z.C.: Frequency characteristics analysis of transmission model parameters of circular fluid lines. *J. Beijing Polytech. Univ.* **2**, 004 (1984)
5. Zhang, M.L., Li, X.F., Xu, J.L., et al.: Pulse wave feature extraction based on improved slope thresholding method. *Electron. Meas. Technol.* **40**(4), 96–99 (2017)
6. Sugawara, R., Horinaka, S., Yagi, H., et al.: Central blood pressure estimation by using N-point moving average method in the brachial pulse wave. *Hypertens. Res.* **38**(5), 336–341 (2015)
7. Yuan, R., Lin, Y.: Traditional Chinese medicine: an approach to scientific proof and clinical validation. *Pharmacol. Ther.* **86**(2), 191–198 (2000)
8. Raghu, P.P., Yegnanarayana, B.: Supervised texture classification using a probabilistic neural network and constraint satisfaction model. *IEEE Trans. Neural Netw.* **9**(3), 516–552 (1998)
9. Wang, H.Y., Xu, S.: Automatic pulse recognition method based on Bayesian classifier. *Chin. J. Biomed. Eng.* **28**(5), 735–742 (2009)
10. Thakker, B., Vyas, A.L., Farooq, O., et al.: Wrist pulse signal classification for health diagnosis. In: *4th International Conference on Biomedical Engineering and Informatics*, pp. 1799–1805. *IEEE* (2011)
11. Xia, P., Zhang, L., Li, F.: Learning similarity with cosine similarity ensemble. *Inf. Sci.* **307**, 39–52 (2015)
12. Tian, X., Guo, Y.: A cosine theorem based algorithm for similarity aggregation of ontologies. In: *International Conference on Signal Processing Systems*, pp. V2–16. *IEEE* (2010)
13. Kulkarni, A.H., Patil, B.M.: Template extraction from heterogeneous web pages with cosine similarity. *Int. J. Comput. Appl.* **87**(3), 4–8 (2014)
14. Moore, B.: Principal component analysis in linear systems: controllability, observability, and model reduction. *IEEE Trans. Autom. Control* **26**(1), 17–32 (2003)
15. Maaten, L.V.D.: Probabilistic Principal Components Analysis. *Dictionary of Bioinformatics and Computational Biology*, pp. 299–307. Wiley, Hoboken (2013)



Hybrid Invasive Weed Optimization and GA for Multiple Sequence Alignment

Chong Gao¹, Bin Wang^{1(✉)}, Changjun Zhou¹, Qiang Zhang^{1,2(✉)},
Zhixiang Yin³, and Xianwen Fang³

¹ Key Laboratory of Advanced Design and Intelligent Computing,
Dalian University, Ministry of Education, Dalian 116622, China
wangbin@dlu.edu.cn, zhangq30@gmail.com

² School of Computer Science and Technology,
Dalian University of Technology, Dalian 116024, China

³ School of Mathematics and Big Data,
Anhui University of Science and Technology, Huaian 232001, China

Abstract. Multiple sequence alignment is one of fundamental problems in bioinformatics, and to design a targeted and effective algorithm for multiple DNA, RNA or protein sequences. The research is to find out the maximum similarity matching between them, whether it should be homologous. In this paper, the invasive weed optimization (IWO) algorithm is combined with GA for multiple sequence alignment, in which IWO algorithm is used to improve the ability of global search. Furthermore, the optimal preservation strategy is used into the proposed algorithm. Comparing two test sequence sets, the results show that the proposed algorithm is effective and reliable.

Keywords: Multiple sequence alignment · Invasive weed optimization
Genetic algorithm

1 Introduction

Bioinformatics [1] is a cross discipline with comprehensive utilization of computer science, biology, mathematics and other subjects of interdisciplinary knowledge. Its basic task is to analyze the various biological macromolecular sequences, from the huge amount of sequence information acquiring knowledge such as gene structure, function and evolution. Comparison is a kind of basic bioinformatics sequence analysis method, which to find the nucleic acid and protein sequences contains the function, structure and evolution of information has very important significance.

Sequence alignment is actually using a particular mathematical model or algorithm to find the maximum number of matching bases between two or more sequences. Sequence alignment algorithm is based on a given scoring matrix or function to calculate sequence of two or more strings optimal comparison. Sequence alignment can be divided into double sequence alignment [2] and multiple sequence alignment [3]. Multiple sequence alignment algorithm in general can be divided into three categories: gradual alignment algorithm [4], precise alignment algorithm [5, 6] and iterative algorithm [7, 8].

Due to the exponential growth of DNA or protein database capacity, when comparing the sequence of more than two, multiple sequence alignment based on the basic dynamic programming algorithm [9–11] is staggering amount of calculation, which makes the multiple sequence alignment become more complicated. Reduce the algorithm complexity is an important aspect of the multiple sequence alignment. Genetic algorithm in solving the problem of sequence alignment, it is easy to fall into local optimum, instability and so on. The design of genetic operators directly affects the convergence speed and the quality of the solution, so it is a key to design genetic algorithm.

In recent years, more and more improved intelligent algorithms are applied to solve the problem of multiple sequence alignment. Boyce et al. [12] studied tradeoff and found that, because of in the early approach information was lost, and the most common multiple sequence alignment programs generated the alignments were unstable. Although the effect was very obvious with large amounts of sequences, it also could be seen with data sets in the order of one hundred sequences. Orobitz et al. [13] used high-performance computing (HPC) resources and techniques were crucial. They applied HPC techniques in T-Coffee and integrated three innovative solutions into T-Coffee, the results showed that it was able to improve the scalability which was execution time and the number of sequences to be aligned. Katoh et al. [14] selected a simple hill-climbing approach as the default, which based on comparisons of the objective scores and benchmark scores between the two approaches. At last, they studied that the simple hill-climbing approach was faster, and therefore was adopted as the default. DeBlasio et al. [15] developed a greedy approximation algorithm that found near optimal sets of size given an optimal solution of size. They found that the coefficients for the estimator performed well in practice. Mirarab et al. [16] introduced a new and highly scalable algorithm, PASTA, which was for multiple sequence alignment estimation. They presented a study on biological and simulated data with more than 200,000 sequences, it showed that the algorithm produced accurate alignments and was able to analyze much larger datasets.

In this paper, we propose an improved algorithm which combines IWO algorithm with genetic algorithm to research multiple sequence alignment. Because IWO algorithm has strong robustness, good convergence and also combines the ideology of global and local search, we combine it with genetic algorithm to enhance the availability and reliability of proposed algorithm. Furthermore, it can enrich the diversity of the population, and have good global optimization capability. In addition, the paper also uses the optimal preservation strategy into the algorithm. Comparing with the results of other works, our algorithm is effective and can get better results.

2 Multiple Sequence Alignment

A length of l sequence is made up of a string with l characters, and the characters in the string are drawn from a finite alphabet Σ . A sequence set is composed of n sequences is $S = (s_1 s_2 \cdots s_n)$, among them, $s_i = s_{i1} s_{i2} \cdots s_{il}$ ($1 \leq i \leq n$), $s_{ij} \in \Sigma$ ($1 \leq j \leq l_i$), and l_i is the i -th sequence length. A multiple sequence alignment S can be defined as a matrix,

$$A = (a_{ij}), 1 \leq i \leq n, 1 \leq j \leq l, \text{ and } \max(l_i) \leq l \leq \sum_{i=1}^n l_i.$$

The matrix has the following characteristics:

- (1) If you delete the empty space “-”, the corresponding sequence in each row of the A is the same as that of the sequence group S .
- (2) $a_{ij} \in \Sigma \cup \{-\}$, “-” representative of empty space.
- (3) There is no column composed of empty space in A .

The general idea of sequence alignment is to align the sequences up and down, by inserting spaces in the sequence to make the same amino acid residues or bases in the sequence as many as possible up and down. For example, the double sequence alignment before and after the changes in the space as shown in Fig. 1.

After comparing the sequence of length is L . The same as the double sequence alignment, the result of the multiple sequence alignment score can also be used to measure similarity. The similarity score with formula is expressed as: [18]

$$Score_{Align} = \sum_{i=1}^L \sum_{j=1}^{n-1} \sum_{k=j+1}^n V_{score}(b_i, b_j) \tag{1}$$

$V_{score}(b_i, b_j)$, it represents a comparison of two bases in a column. n indicates the number of rows, and L is a column number.

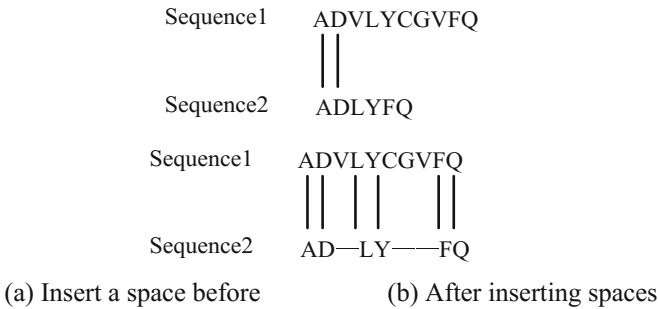


Fig. 1. The change of sequence comparison when adding gaps.

The scoring rules for multiple sequence alignment are shown in the formula (2) [19].

$$V(b_i, b_j) = \begin{cases} 6 & \text{if } b_i, b_j \in \Sigma \text{ and } b_i = b_j \\ 1 & \text{if } b_i, b_j \in \Sigma \text{ and } b_i \neq b_j \\ 0 & \text{if } b_i \in \Sigma \text{ and } b_j \text{ is a space} \\ 1 & \text{if } b_i \in \Sigma \text{ and } b_j \text{ is a space extension} \end{cases} \tag{2}$$

In the multiple sequence alignment, we usually adopt sum of pairs score (SPS) to measure the performance of the multiple sequence alignment procedure. SPS indicates the ratio of residues to correct alignment. Assuming that we have a test alignment which has S sequences consisting of K columns. The i -th column in the alignment can be represented as $A_{1i}A_{2i} \cdots A_{Ni}$. For each pair of residues A_{ji} and A_{ki} , we define $P_{jki} = 1$ that if residues A_{ji} and A_{ki} are aligned with each other in the reference alignment, otherwise $P_{jki} = 0$. The score of SP_i for the i -th column is defined as: [20]

$$SP_i = \sum_{j=1}^S \sum_{k=1, j \neq k}^S P_{jki} \quad (3)$$

Then SPS for the alignment is that:

$$SPS = \frac{\sum_{i=1}^K SP_i}{\sum_{i=1}^{K_r} SP_{ri}} \quad (4)$$

Here, K_r is the number of columns in the reference alignment and SP_{ri} is the score SP_i which is the i -th column in the reference alignment.

3 Description of Algorithm

3.1 IWO Algorithm

IWO algorithm is a bionic simulation of the propagation process of a new random search optimization algorithm. In basic IWO, weeds are the feasible solution of the problem, and the population is the collection of all weeds. The implementation of IWO algorithm is to experience the four steps of the weed population initialization, reproduction, spatial dispersal and competitive exclusion rules.

3.1.1 Initialization

A certain number of weeds in D dimension diffusion distribution in a random way, in general, the weeds in the initial population size may be adjusted according to the practical problems.

3.1.2 Reproduction

The number of individuals (weeds) in the process of breeding is related to the degree of fitness of the weeds, the high fitness of weeds produce more seeds, and the low degree of fitness produce fewer seeds. The number of seeds produced by the weeds is: [17]

$$num = \frac{fit - fit_{\min}}{fit_{\max} - fit_{\min}} (seed_{\max} - seed_{\min}) + seed_{\min} \quad (5)$$

In the formula, fit is the fitness for the current weed; fit_{\max} and fit_{\min} are respectively corresponding to the maximum and minimum of weed in current population fitness values; $seed_{\max}$ and $seed_{\min}$ respectively represent a weed can produce seeds of maximum and minimum values.

3.1.3 Spatial Dispersal

The seed produced by the weed grows into weeds according to a certain step size, and the average value is 0, the standard deviation is $stepLen$. With the increasing number of evolution, current standard deviation is calculated as follows: [17, 21]

$$stepLen = \frac{(iter_{\max} - iter)^n}{(iter_{\max})^n} (stepLen_{init} - stepLen_{final}) + stepLen_{final} \quad (6)$$

Where $iter$ is the current iterations number; $iter_{\max}$ is the maximum iterations number; $stepLen_{init}$ and $stepLen_{final}$ represent the initial standard deviation and the final standard deviation; n is the nonlinear factor.

3.1.4 Competitive Exclusion

Weed after several generations of breeding, the population size will increase rapidly, and environment carrying capacity is limited, difficult to support the weeds uncontrolled expansion, at this time, the evolution law of survival of the fittest also dominates the weed population late only to adapt to the degree of seeds in order to obtain enough resources to survive. After several generations of evolution, the weeds and seeds achieve the maximum population size. In the process of the algorithm, all the weeds and their offspring are sorted according to the fitness value by descending sort, and only the best individuals can survive, and the rest will be rejected by the environment.

3.2 Improved Algorithm

Aiming at the shortcoming of traditional genetic algorithm, this paper mainly made two improvements: IWO and the optimal preservation strategy. We use IWO to replace the GA selection operation. IWO is taking into account the global search and local search, and both can be adjusted according to the number of iterations. The competitive exclusion mechanism in IWO can preserve useful information, which can avoid premature convergence and local optimum.

3.2.1 IWO

Based on genetic algorithm to initialize the population, and then select $NIND$ individuals (weeds) for IWO operation.

According to the formula (1) calculate the $NIND$ weeds fitness value, and find the fitness of the maximum and minimum values. According to the formula (5), the number of seeds produced by each weed is calculated. The value of the current standard deviation is calculated from the given parameters, which computational method as shown in formula (6).

In the initial stage of evolution, the standard deviation is relatively large, the distribution of a wide range of seed, it showing that global search algorithm; Finally,

the standard deviation becomes smaller, at this time seeds mainly distribute in the parent around, which showing that local search algorithm. This mechanism can balance the global searching and local solution well, and in a certain extent, it can avoid premature convergence.

In multiple sequence alignment, the location of the space is random, so we take it as a handle object. Calculating the space position of each weed, we can use formula (7) calculate seeds by each weed. We use the Cauchy distribution to replace the normal distribution in the IWO, and improve the convergence of the algorithm as well as possible to ensure the robustness of the algorithm. Formula (7) is shown as follows: [17, 21]

$$weed = \text{mod}(X_1 + \text{round}(\text{stepLen} * \text{trnd}(1, \text{lin}, \text{row})), L) \quad (7)$$

Where X_1 represents the space position of each weed. lin is the number of rows for sequence, and row represent the number of spaces in a row. L is the length of the sequence. $\text{trnd}(1, \text{lin}, \text{row})$ represents the standard Cauchy distribution.

Calculating the fitness values of all the seeds produced by each weed, then compare with the fitness value of the weed itself, and replace the weed with the largest fitness value.

3.2.2 Optimal Preservation Strategy

When the initial population, the preservation of the best individual a_1 in the population. Through the IWO, crossover and mutation operation, then find out the worst individual b_1 and best individual temp in the population. Comparing the size of a_1 and temp , with the largest individual replace the worst individual b_1 and ensure the best individual is not damaged.

3.3 Algorithm Flow

According to genetic algorithm and IWO algorithm, in this paper, we propose a hybrid algorithm based on IWO and genetic algorithm for multiple sequence alignment.

The main steps of algorithm as follows:

- Step 1: Initializing the genetic and IWO parameters;
- Step 2: Initializing population. Each individual has c (c refers to the number of insert spaces) sub individuals, and choose one of the biggest fitness value of the individual as an individual in the parent population;
- Step 3: Using the method of Sect. 3.2.1 to carry on the IWO operation and to produce the seeds;
- Step 4: Using single-point crossover method for crossover operation. Choosing a random point t ($t < L$, L represents the sequence number of rows.), before t lines from the individual A, and the $(L - t)$ lines from the individual B;
- Step 5: A certain mutation probability of randomly selecting one individual from the parent, then delete all the rows of individual spaces, at last re-insert the same random number of spaces, and generate a new individual;

Step 6: After completing the crossover and mutation operation, we put the resulting population and the population that has not evolved together, and the individual with a good fitness value of *NIND* is selected. After using the method of Sect. 3.2.2 to perform the operation of the optimal preservation strategy;
 Step 7: Judge whether the algorithm termination condition is achieved, if achieved, output the optimal sequence alignment score, otherwise, go to step 3.

Based on the method of flow chart is shown in Fig. 2.

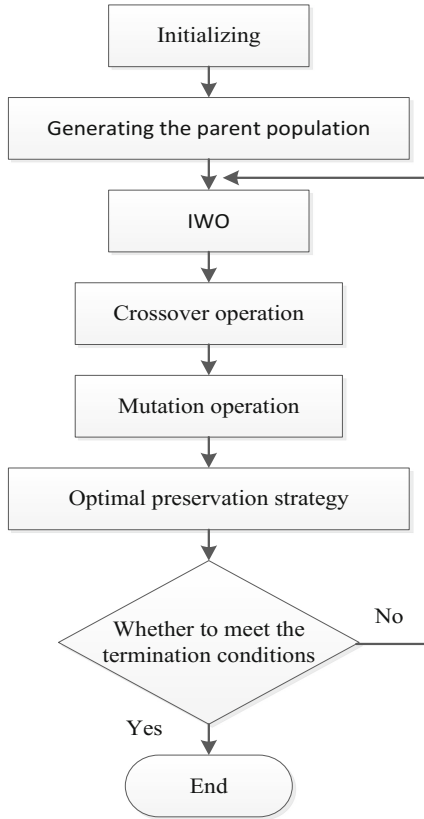


Fig. 2. Flow of the algorithm.

4 Experimental Data and Results

4.1 Experimental Data

In the algorithm, the algorithm mentioned is run in the MATLAB R2012a and we can use some real data on the internet which come from (<http://www.ncbi.nlm.nih.gov/>).

Table 1. Sequence code [19].

Name	Sequence code
TAR	L28864.1_329-385
	M93259.1_9532-9588
	AF443088.1_8897-8953
	AF196710.1_461-517
	AJ286133.1_8742-8798
IRES_PICO	AF230973.1_399-650
	D00627.1_394-645
	AF524867.1_393-644
	AY186745.1_373-624
	AJ295195.1_354-605

We use the data from BALiBASE. BALiBASE is a benchmark database for multiple sequence alignment. In the paper, we use two data sets, and each of them has five sequences of equal length. Specific sequence code is shown in Table 1 [19].

Table 2. Genetic algorithm parameters.

Parameter	TAR	IRES_PICO
Population size	50	100
Crossover probability	0.8	
Mutation probability	0.01	

Table 3. IWO parameters.

Parameters symbol	Symbol meaning	Parameters value
$seed_{max}$	Maximum number of seeds	5
$seed_{min}$	Minimum number of seeds	3
$iter_{max}$	The number of iterations	200
$stepLen_{init}$	The initial standard deviation	50
$stepLen_{final}$	The final standard deviation	0.005
n	Nonlinear modulation index	3

We need to set the parameters of genetic algorithm and IWO. The genetic algorithm parameters are given in the following Table 2 and details of IWO parameters are shown in Table 3.

4.2 Experimental Results

In order to verify the feasibility of our proposed algorithm, we use the two group of RNA sequence sets to analyze. At the same time, in order to compare the fairness, the algorithm must have the same parameter adjustment. In the paper, we put the sequence

Table 4. Experimental results.

Sequence sets	Optimal results	Improved genetic algorithm [19]	Chaos genetic algorithm [22]	This paper
TAR	Alignment score	3206	3261	3223
	SPS	0.93	0.984	0.988
IRES_PICO	Alignment score	13457	13850	13861
	SPS	0.86	0.996	0.997

alignment score as the main measure of the standard. According to the above tables that set the parameter values, the results of this paper and the results of the compared paper are shown in Table 4.

In this paper, we provide two groups of contrast data to compare the results of this paper. The first set of data comes from the results of the Zhang et al. [19] paper. The second set of data is from the chaos genetic algorithm [22], we implement this algorithm and use it to handle multiple sequence alignment.

In TAR sequence sets: for the first group of data, the best sequence alignment score in different parameters of Zhang et al. [19] paper is 3206, we use the chaos genetic algorithm [22] to get the optimal alignment score is 3261, and in this paper we get the optimal score is 3223 by hybrid algorithm, although the result is less than the chaos genetic algorithms, the SPS is better than the others. In IRES_PICO sequence sets, from the table we can see that the results obtained by using the hybrid algorithm are better than those of the other two. SPS is using to measure the performance of the multiple sequence alignment procedure, so we can get a good result by using hybrid IWO and genetic algorithm. It also shows that the feasibility of the new algorithm to deal with multiple sequence alignment.

5 Conclusion

In this paper, in order to make up for the defect of traditional genetic algorithm, IWO algorithm is introduced. We combine the IWO and genetic algorithm together to deal with multiple sequence alignment, so a new hybrid evolutionary algorithm is proposed. We apply this new algorithm to the multiple sequence alignment, and select the appropriate parameter values from the IWO algorithm to get good results. Through the IWO, the diversity of population is enriched and the algorithm has the good ability of global optimization. The optimal preservation strategy is also added to the algorithm. The experimental results show that our hybrid algorithm is superior to other methods. At the same time, the results verify the feasibility and effectiveness of the proposed algorithm.

Although IWO has greatly improved in the breadth of the search, the performance of the algorithm are greatly influenced by the parameter. This will lead to parameter optimization problem. We need to further explore the discrete aspects in the future. In

the future, we will use the results from our recent works [23–26], and improve the method for multiple sequence alignment.

Acknowledgment. This work is supported by the National Natural Science Foundation of China (Nos. 61425002, 61751203, 61772100, 61702070, 61672121, 61572093), Program for Changjiang Scholars and Innovative Research Team in University (No. IRT_15R07), the Program for Liaoning Innovative Research Team in University (No. LT2015002).

References

1. Zhang, C.T.: Current status and prospects of bioinformatics. *World Sci. Technol. Res. Dev.* **22**(6), 17–20 (2000)
2. Wu, D.M., Chen, J.: Research on algorithm of pairwise alignment. *Comput. Eng. Appl.* **44**(36), 48–50 (2016)
3. Zou, Q., Guo, M.Z., Han, Y.P.: Development of multiple sequence alignment algorithms. *China J. Bioinform.* **04**, 311–314 (2010)
4. Carrillo, H., Lipman, D.J.: The multiple sequence alignment problem in biology. *SIAM J. Appl. Math.* **48**(5), 1073–1082 (1988)
5. Hogeweg, P., Hesper, B.: The alignment of sets of sequences and the construction of phylogenetic trees: an integrated method. *J. Mol. Evol.* **20**(2), 175–186 (1984)
6. Taylor, W.R.: A flexible method to align large numbers of biological sequences. *J. Mol. Evol.* **28**(1–2), 161–169 (1988)
7. Thompson, J.D., Higgins, D.G., Gibson, T.J.: CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucl. Acids Res.* **22**(22), 4673–4680 (1994)
8. Notredame, C., Higgins, D.G., Heringa, J.: T-coffee: a novel method for fast and accurate multiple sequence alignment. *J. Mol. Biol.* **302**(1), 205–217 (2000)
9. Hu, Y.Q.: Research Foundation and Application. Harbin Institute of Technology Press, Harbin (1987)
10. Gan, Y.A., Tian, F., Li, W.Z.: Operations Research. Tsinghua University Press, Beijing (1994)
11. Wang, Y.X.: Planning and Network of Operations Research. Tsinghua University Press, Beijing (1993)
12. Boyce, K., Sievers, F., Higgins, D.G.: Instability in progressive multiple sequence alignment algorithms. *Algorithm Mol. Biol.* **10**(1), 1–10 (2015)
13. Orobítg, M., Guirado, F., Cores, F.: High performance computing improvements on bioinformatics consistency-based multiple sequence alignment tools. *Parallel Comput.* **42**, 18–34 (2015)
14. Katoh, K., Toh, H.: Parallelization of the MAFFT multiple sequence alignment program. *Bioinform. Oxf. J.* **26**(15), 1899–1900 (2010)
15. DeBlasio, D., Kececioglu, J.: Parameter advising for multiple sequence alignment. *BMC Bioinform.* **16**(2), 516–518 (2015)
16. Mirarab, S., Nguyen, N., Warnow, T.: PASTA: ultra-large multiple sequence alignment. In: Sharan, R. (ed.) RECOMB 2014. LNCS, vol. 8394, pp. 177–191. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-05269-4_15
17. Mehrabian, A.R., Lucas, C.: A novel numerical optimization algorithm inspired from weed colonization. *Ecol. Inform.* **1**(4), 355–366 (2006)

18. Li, S.Z., Mo, Z.S., Zhang, X.: Multiple sequence alignment based on immune genetic algorithm. *J. Wuhan Univ.* **50**(5), 537–541 (2004)
19. Zhang, Y., Achawanantakun, R.: An improved genetic algorithm for multiple sequence alignment. Project report of CSE848, Fall (2010)
20. Song, X.L.: Research of multiple sequence alignment algorithm based on quantum genetic algorithm and improved immune genetic algorithm. Master thesis, Jilin University, Jilin (2007)
21. Luo, D.F., Luo, D.J.: The research of DNA coding sequences based on invasive weed optimization. *Sci. Technol. Eng.* **13**, 3545–3551 (2013)
22. T. J. E.: Timetabling problem research on chaos genetic algorithm. Master thesis, Harbin Engineering University, Harbin (2009)
23. Yang, J., et al.: Entropy-driven DNA logic circuits regulated by DNAzyme. *Nucl. Acids Res.* (2018). <https://doi.org/10.1093/nar/gky663>
24. Wang, B., et al.: Constructing DNA barcode sets based on particle swarm optimization. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **15**, 999–1002 (2018)
25. Pan, L., Wang, Z., Li, Y., Xu, F., Zhang, Q., Zhang, C.: Nicking enzyme-controlled toehold regulation for DNA logic circuits. *Nanoscale* **9**(46), 18223–18228 (2017)
26. Wang, B., Xie, Y., Zhou, S., Zheng, X., Zhou, C.: Correcting errors in image encryption based on DNA coding. *Molecules* (2018). <https://doi.org/10.3390/molecules23081878>



RNA Sequences Similarities Analysis by Cross-Correlation Function

Shanshan Xing¹, Bin Wang^{1(✉)}, Xiaopeng Wei^{1,2}, Changjun Zhou¹,
Qiang Zhang^{1,2(✉)}, and Zhonglong Zheng³

¹ Key Laboratory of Advanced Design and Intelligent Computing,
Dalian University, Ministry of Education, Dalian 116622, China
wangbin@dlu.edu.cn, zhangq30@gmail.com

² School of Computer Science and Technology,
Dalian University of Technology, Dalian 116024, China

³ College of Mathematics, Physics and Information Engineering,
Zhejiang Normal University, Jinhua 321004, China

Abstract. According to concept of dinucleotide, this paper proposes a graphical representation of dinucleotide. In this way, the graphical representation can not only show sequence one-to-one, but also can be easily converted into the original sequence. Then this paper extracts cross-correlation function to characterize the degree of similarity from representation of dinucleotide. After applying our approach to nine kinds of viruses, it can be found that our conclusion is almost consistent with the reported data. After the analysis with the method of inter-class, it can be found that our data can classify different viruses well. Our approach can more easily extract data and distinguish the different classes than previous results.

Keywords: RNA sequences · Graphical representation of dinucleotide
Similarities degree

1 Introduction

Biological sequences analysis is a very important element in bioinformatics. The core issue of biological sequence analysis is the comparison between different types of biological sequences [1]. The ultimate goal is to find and determine conserved regions and variation rules between the different biological sequences. Then we find their different function, structural characteristics and other differences.

The most common comparison method is the sequence alignment, and it provides a very clear pattern of the relationship between two or more sequences of residues [2, 3]. Sequence alignment is an approach that by searching for a series of single trait or trait pattern in the sequence to compare the two (pairwise alignment) or more (multiple sequence alignment) sequence. The two sequences are written in two lines to be aligned. Similar or the same traits of residues or bases are placed in the same columns, and different traits are either placed in the same column as a mismatch, or in another sequence corresponding to an interval. In a preferred arrangement, the placement of spacing and different traits should be possible to make the same or similar traits vertical

alignment. Currently sequence alignment methods mostly used dynamic programming algorithms. The efficiency of dynamic programming algorithm increased exponentially with the number of sequences.

Another sequences comparison method with the rapid development is alignment-free sequence analysis in recent years. Alignment-free sequence analysis include the graphical representation [4–6], and statistical methods [7, 8] et al. The method based on K-word was a kind of classical statistical methods [9, 10]. But the statistical methods ignored the chemical structure and properties of biomolecules. The main process of the graphical representation approach was to map sequence to the geometry. Such the complex relationships of biological sequences were able to visualize. Then they used numerical characteristics depict graphic representation. So they did further analysis and research on biological sequences generated based on the numerical characteristics. An advantage of alignment-free sequence analysis was to circumvent the problem of selecting complete genome sequence of multiple genes. Secondly, alignment-free sequence analysis had lower computational complexity and the less time-consuming. There were also using information theory methods, such as the complexity of Kolmogorov [11], Kullback-Leibler deviation method [12], the probability method [13].

RNA is a kind of important biological macromolecule in biological system, and it plays an important role in the life system. RNA secondary structures refer to stem-loop structures formed by RNA single-stranded folding back forms part of the base pairs and single-stranded itself [14]. RNA structure information for analyzing genetic information of the virus and controlling their function plays a very important role. RNA secondary structure plays an important role in the study of interactions between mRNA and protein and the treatment mRNA stabilization [15]. RNA similarity analysis is the analysis of RNA secondary structure similarity.

There are many algorithms to analyze similarity degree of RNA secondary structures [16–20]. Similarity degree analysis of RNA secondary structure can be divided into three categories: alignment methods, tree comparison methods and methods based on graphical representation. Alignment method was a kind of sequence alignment method. Tree comparison method was based on the topological invariants of tree structures, but it ignored the structural information and order information in bases. Many graphical representation methods have been proposed and applied to the RNA similarity calculation.

This paper presents RNA graphical representation based on a graphical representation of dinucleotide [21, 22]. Four bases in sequences are respectively represented by four numbers. Then sequences are transformed into a series of numerical sequences using dinucleotide. And finally it is graphic representation. We convert the sequence into a dinucleotide pattern and can visually see the change in sequence. At the same time, we can also easily convert the original sequence by the graphical representation. On the bases of the graphical representation, we extract cross-correlation function of sequences from the graphical representation. Wherein a sequence as the base sequence, then we calculate cross-correlation functions between every sequence and the base sequence. We obtained 5-dimensional cross-correlation functions value which composed 5-dimensional vector. Then we calculate the Euclidean distance between every two vectors and the Euclidean distance is as similarity value to characterize RNA sequence similarity degree. Finally, we use the method sequence clustering analysis to

verify the validity of our approach. It is easier to distinguish the different class than previous results.

2 Graphical Representation of RNA Secondary Structure

Typically, RNA secondary structure is composed of four bases A, U, G, C. In this way, RNA secondary structures are transformed into elementary sequences, called characteristic sequences [23]. For example, the substructure of ‘LRMV-3’ (see Fig. 1) corresponds to the characteristic sequence ‘GUUCCUAUUCUCUCUCAGGAGAGGA-GAAUAGAUGCCUCCAAAGGAGU CGC’ (from 5’ to 3’).

For any characteristic sequences of the RNA secondary structure $S = S_1S_2 \dots S_i \dots S_N$, where N is the length of the signature sequence of RNA secondary structures. With the method of dinucleotide graphical representation, at first, the RNA characteristic sequence is converted into numeral sequence of quaternary. Four bases in sequences are respectively represented by four digits of quaternary. Let

$$U \rightarrow 0, C \rightarrow 1, G \rightarrow 2, A \rightarrow 3.$$

Thus, this paper has characteristic sequences of the RNA secondary structure converted into numeral sequence of quaternary. Then we have numeral sequences of quaternary transformed into dinucleotide numeral sequences using the concept of dinucleotide. From left to right every two adjacent quaternary base figures form a group, and then it calculated the decimal value of each group [22]. Finally, the dinucleotide sequence of numbers is shown in a graph. Thus, we get the dinucleotide graphical representation of RNA secondary structures.

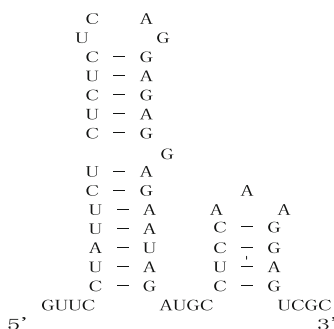


Fig. 1. The substructure of LRMV-3.

For example, the characteristic sequence of ‘LRMV-3’, this paper has it converted into numeral sequence of quaternary as follows:

$$S_q = '200110300101010101013223232232330323021101133322320121'$$

Grouping each adjacent pair of quaternions, then we calculate its decimal value of dinucleotide as follows:

$$S_d = '8, 0, 1, 5, 4, 3, 12, 0, 1, 4, 1, 4, 1, 4, 1, 4, 1, 7, 14, 10, 11, 14, 11, 14, , 10, 11, 14, 11, 15, 12, 3, 14, 11, 12, 2, 9, 5, 4, 1, 5' \quad '7, 15, 15, 14, 10, 11, 14, 8, 1, 6, 9'$$

Namely, the first three adjacent pairs of quaternions are ‘20, 00, 01’, so the corresponding decimal number are ‘8, 0, 1’

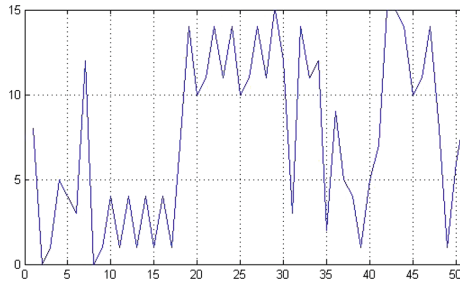


Fig. 2. Dinucleotide graphical representation of LRMV-3.

Finally, this paper shows the dinucleotide sequence of numbers in a graph. The Fig. 2 is the dinucleotide graphical representation of LRMV-3. In Fig. 2, the horizontal axis (x axis) indicates the length of the LRMV-3 minus 1, and the vertical axis (y axis) is S_d sequence.

The benefit of this representation is that the graphics and sequences are corresponding. Each RNA sequence corresponds to a unique curve. It can eliminate the degradation, and have a more comprehensive expression of biological information. In the previous graphical representation, most nucleotides were regarded as the individual bases without considering the contact between single bases. In biological sequence it has a certain link between the bases. The dinucleotide graphical representation regard adjacent base as a group. It takes into account the relationship between the neighboring bases to some extent. It also can easily restore the original characteristic sequence by the graphical representation of secondary structure. From the graphical representation, we can roughly determine the similarity degree of sequence.

3 The Numerical Characteristics of RNA Secondary Structures

After RNA characteristics sequence is converted into a numeral sequence, and represented in the graph, this paper gets a two-dimensional space curve of RNA sequence. Then this artical extracts feature vectors which are cross-correlation functions from graphical representation. At first, this paper selects the shortest sequence as the base

sequence. Then it calculated cross-correlation functions between every sequence and the base sequence. Suppose the two sequences are $x(n)y(n)$, the cross-correlation function between the two signals is the formula (1).

$$R(n) = \frac{\sum_{m=0}^{N-n} [x(m)y(m+n)]}{N} \quad (1)$$

The length of the sequence is equal in formula (1). N refers to the length of the sequence $x(n)$, $y(n)$ in formula (1). We calculate five cross-correlation function values to form a 5-dimensional vector ($n = 0, 1, 2, 3, 4$). Such every sequence would get their five dimension cross-correlation function with the base vector sequence. The five-dimensional vectors of sequences X , Y are as formula (2), (3).

$$X^5 = \{R_X(0), R_X(1), R_X(2), R_X(3), R_X(4)\} \quad (2)$$

$$Y^5 = \{R_Y(0), R_Y(1), R_Y(2), R_Y(3), R_Y(4)\} \quad (3)$$

Finally, the Euclidean distance (as formula (4)) of the vector end is calculated to compare their similarity degree.

$$d(X, Y) = \sqrt{\sum_{i=0}^4 [R_X(i) - R_Y(i)]^2} \quad (4)$$

In the cross-correlation function formula, the lengths of two sequences are equal. In reality, the length of each RNA sequences is unequal. Therefore, there will be problems in calculating cross-correlation function of sequence. In this paper, measure we have taken is to move short sequence to align long sequence in turn, and cut the extra bases. Then we calculate 5-dimensional cross-correlation function values between sequence of cutting extra base and short sequence. We take a minimum value of each dimension to form a new five-dimensional vector. We use the new five-dimensional vector to represent the sequence's vector of cross-correlation function. Thus, we have calculated each nucleotide of long sequence, and it does not result in the loss of sequence information. Because in the calculation of the cross-correlation function, each nucleotide of long sequence are not missing and are calculated.

4 The Result of 9 Kinds of RNA Viruses

In order to demonstrate the feasibility of this paper's proposed method, this paper uses nine different kinds of RNA viruses to analyze. First, this paper has the secondary structure of nine kinds of RNA viruses represented graphically, and then extract numeral characteristics. After the graphical representation, RNA sequences are converted into dinucleotide sequences. Our extracted numeral characteristic is the cross-correlation function of dinucleotide sequences. When extracting the cross-correlation

Obviously, the similarity of two identical sequences is 0 and 0 is the lowest value in similarity matrix. Therefore the more close to 0 Euclidean distance is, the more similar two kinds of RNA viruses is.

In Table 2, it can be found that AVII, CiLARV-3, LRMV-3, TSV-3, CVV-3 and EMV-3 are more similar with each other because the Euclidean distance between them is relatively small. They are regarded as same cluster. Thus the AIMV-3, APMV-3 and PDV-3 are out of the same cluster. In addition, AIMV-3, APMV-3 and PDV-3 also have a high degree of similarity. They are regarded as the other cluster. It is not only the result of the real life that is consistent with the results of this paper, but also the results of the references [16, 17, 19, 23] are consistent with this article. The data in Table 3 come from Reference [16], while the data in Table 4 come from Reference [19]. The larger differences of similarity matrix elements contribute to the following cluster analysis, which show the advantage of our method.

Table 3. RNA similarity matrix reported in [16].

Species	AIMV-3	CiLRV-3	TSV-3	CVV-3	APMV-3	LRMV-3	PDV-3	EMV-3	AVII
AIMV-3	0	0.5439	0.3790	0.4862	0.2901	0.5227	0.2042	0.5766	0.6607
CiLRV-3		0	0.2275	0.0699	0.5620	0.2465	0.6339	0.1002	0.1665
TSV-3			0	0.2166	0.3790	0.3083	0.4293	0.3144	0.3767
CVV-3				0	0.5156	0.1994	0.5937	0.1222	0.1806
APMV-3					0	0.4594	0.2042	0.6366	0.6607
LRMV-3						0	0.6002	0.2917	0.2359
PDV-3							0	0.7001	0.7638
EMV-3								0	0.1492
AVII									0

Obviously, the similarity of two identical sequences is 0 and 0 is the lowest value in similarity matrix. Therefore the more close to 0 Euclidean distance is, the more similar two kinds of RNA viruses is.

In Table 2, it can be found that AVII, CiLARV-3, LRMV-3, TSV-3, CVV-3 and EMV-3 are more similar with each other because the Euclidean distance between them is relatively small. They are regarded as same cluster. Thus the AIMV-3, APMV-3 and PDV-3 are out of the same cluster. In addition, AIMV-3, APMV-3 and PDV-3 also have a high degree of similarity. They are regarded as the other cluster. It is not only the result of the real life that is consistent with the results of this paper, but also the results of the references [16, 17, 19, 23] are consistent with this article. The data in Table 3 come from Reference [16], while the data in Table 4 come from Reference [19]. The larger differences of similarity matrix elements contribute to the following cluster analysis, which show the advantage of our method.

Table 4. RNA similarity matrix reported in [19].

Species	AIMV-3	CiLRV-3	TSV-3	CVV-3	APMV-3	LRMV-3	PDV-3	EMV-3	AVII
AIMV-3	0	0.3294	0.3467	0.4789	0.0296	0.5067	0.1160	0.5177	0.5320
CiLRV-3		0	0.0185	0.1523	0.3007	0.1782	0.2138	0.1890	0.2032
TSV-3			0	0.1341	0.3180	0.1603	0.2309	0.1712	0.1854
CVV-3				0	0.4504	0.0318	0.3630	0.0422	0.0560
APMV-3					0	0.4780	0.0874	0.4890	0.5033
LRMV-3						0	0.3907	0.0119	0.0258
PDV-3							0	0.4018	0.4161
EMV-3								0	0.0143
AVII									0

Sequence Clustering is intended to divide sequence data into a plurality of clusters, so that sequences in the same cluster are as similar as possible, and sequences in the different cluster are as much dissimilar as possible [24]. By Sequence Clustering, sequence of each family can be clustered into different subfamilies. The sequences in the same subfamily have relevant functions. In addition, there is a lot of data redundancy in the sequence database. These redundant data is often difficult to provide additional information. After clustering, we can consider only their representatives of sequences.

For convenience, this paper uses AVII and eight other viruses as the objects to analyze our data. From our data, AVII, CiLARV-3, CVV-3, LRMV-3, EMV-3 and TSV-3 are regarded as a cluster because they are quite similar, and other viruses are regarded as a cluster in that they are not similar to AVII. There are relatively large differences between our data. Thus our data is easier to be extracted than the data reported in [16, 19]. Then we analyze the data by the Sequence Clustering method.

Table 5. Data comparison in same class.

Data	Minimum	Maximum	Percentage
Our data	2.2500	46.455	95.16%
Reference [16]	0.1492	0.3767	60.39%
Reference [19]	0.0143	0.2032	92.96%

The minimum refers to minimum in their cluster in the similarity degree matrix in the Table 5, and the maximum refers to the maximum in their cluster in the similarity degree matrix. For example, AVII, CiLARV-3, CVV-3, LRMV-3, EMV-3 and TSV-3 can be seen as a cluster in Table 1. We can find that their corresponding values in the matrix are 4.2133, 2.2500, 32.6101, 10.1392 and 46.455, and the minimum is 2.2500

while maximum is 46.455. Formula 5 reflects the degree between the numbers together to a certain extent, so it can be used to express the quality of classification. Percentage is calculated as formula (5):

$$percentage = \frac{\max - \min}{\max} \times 100\% \quad (5)$$

The percentage in the Table 5 reflects the difference of the same cluster. It can be seen that the smaller the difference of the same cluster is, the smaller the percentage from the formula is. The smaller the percentage is, the closer genetic relationship of species is. In order to make it easier to identify difference of the same type of species, it hoped that the smaller the percentage is, the better it is. There is a relatively large difference between our data. The percentage in our data is worse than that the data reported in [16] in same cluster.

Table 6. Data comparison in different cluster.

Data	Same cluster	Out of same cluster	Percentage
Our data	23.55	284.35	88.02%
Reference [16]	0.2218	0.6951	68.09%
Reference [19]	0.0969	0.4832	79.97%

The data in second column in Table 6 refer to the average in same cluster, while the data in the third column refer to the average out of same cluster. For example, in Table 2, AVII, CiLARV-3, CVV-3, LRMV-3, EMV-3 and TSV-3 are in same cluster, and their corresponding value in the matrix are 4.2133, 2.2500, 32.6101, 10.1392 and 46.455. Their average is 19.134. While 187.979, 190.150, 101.055 are out of the same cluster and their average is 159.728. Percentage is calculated as formula (6):

$$percentage = \frac{\text{out of same cluster} - \text{same cluster}}{\text{out of same cluster}} \times 100\% \quad (6)$$

The percentage in the Table 6 reflects the difference of different cluster. It can be seen that the smaller the greater the gap between values is, the greater the percentage difference from the formula is. Between the same cluster and other clusters, the greater the difference is, the easier it will be distinguished. The greater the percentage is, the further genetic relationship of species is. The greater percentage difference is, the better our proposed algorithm is. Compared with the data reported in [16], the percentage in our data increases by 19.93% in different cluster. The percentage in our data improves 8.05% than the data in [19] in different cluster. Compared with that reported in [16] and [19], our data is improved in different clusters.

From the above analysis, it can be clearly concluded that our data have an advantage. It is relatively quick to identify our data out of the same clusters. On the one hand, our extracted numeral characteristic is the cross-correlation function. In signal processing, the cross-correlation function is used to indicate the degree of correlation

between the two signals, which is an important digital characteristic of signals. This paper applies it to sequence similarity analysis, and it can reflect the matching degree of sequence in different positions. This is the concept of sequence alignments. This paper selects five function values to compose a 5-dimensional vector, which avoids the contingency of one-dimensional figure. At the same time, in the progress of calculating different length sequences, we repeatedly calculate the 5-dimensional vectors, and select the minimum value to form new 5-dimensional vector. This can avoid the loss of information. On the other hand, although the cross-correlation function reflects the concept of sequence alignments, but time complexity of the cross-correlation function is much lower than that of the sequence alignments algorithm. Sequence alignment mostly used dynamic programming algorithm. The efficiency of dynamic programming algorithm increased exponentially with the growth of the number of sequences. Most of its time complexity is exponential form. While the cross-correlation function contains only simple addition and multiplication. Time complexity is a constant form. For example, time complexity of alignment-based algorithms in computational L/L or M/M matrix is at least n^3 , and the alignment-free sequence alignment algorithm time complexity is n . In the process of execution, it saves space and time. Our method is relatively simple in the process of calculation.

While it also has shortcomings, for example, the data is better than that using our method in the same cluster. We can see that our approach is not perfect.

5 Conclusions

This paper presents a graphical representation of dinucleotide sequences. It not only eliminates degradation of image representation to a certain extent, but also can easily achieve conversion between a graphical representation and the original sequence of RNA secondary structure. It takes into account the relationship between the neighboring bases to some extent. Secondly, we use the cross-correlation function of the signal to characterize similarity degree of the RNA sequence. Cross-correlation function can well reflect the degree of correlation between two signals. In the process of extracting characteristic features, the calculation of cross-correlation function formula improves the efficiency. Finally, we apply it to analyze the example of 9 kinds of virus and achieve relatively valuable results. Seen from the analysis results, our conclusion achieves relatively good results, and also verifies the rationality that we use the cross-correlation function to characterize similarity degree.

In addition, the paper still has some shortcomings. The data in reference is better than that using our method in the same cluster. So in the future, we will use the results from our recent works [25–28], and improve the method for RNA sequences similarities analysis.

Acknowledgement. This work is supported by the National Natural Science Foundation of China (Nos. 61425002, 61751203, 61772100, 61702070, 61672121, 61572093), Program for Changjiang Scholars and Innovative Research Team in University (No. IRT_15R07), the Program for Liaoning Innovative Research Team in University (No. LT2015002).

References

1. Luo, J.: *Fundamental Concepts of Bioinformation*. Peking University Press, Beijing (2002)
2. Smith, T.F., Waterman, M.S.: Identification of common molecular subsequences. *J. Mol. Biol.* **147**, 195–197 (1981)
3. Needleman, S.B., Wunsch, C.D.: A General method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.* **48**, 443–453 (1970)
4. Yao, Y., Dai, Q., Ling, L., Nan, X., He, P., Zhang, Y.: Similarity/dissimilarity studies of protein sequences based on a new 2D graphical representation. *J. Comput. Chem.* **31**, 1045–1052 (2010)
5. Feng, J., Wang, T.: A 3D graphical representation of RNA secondary structures based on chaos game representation. *Chem. Phys. Lett.* **454**, 355–361 (2008)
6. Tang, X., Zhou, P., Qiu, W.: On the similarity/dissimilarity of DNA sequences based on 4D graphical representation. *Chin. Sci. Bull.* **55**, 701–704 (2010)
7. Yu, C., Deng, M., Yau, S.T.: DNA sequence comparison by a novel probabilistic method. *Inf. Sci.* **181**, 1484–1492 (2011)
8. Zheng, X., Qin, Y., Wang, J.: A Poisson model of sequence comparison and its application to coronavirus phylogeny. *Math. Biosci.* **217**, 159–166 (2009)
9. Yang, X., Wang, T.: Linear regression model of short K-word: a similarity distance suitable for biological sequences with various lengths. *J. Theor. Biol.* **337**, 61–70 (2013)
10. Yang, X., Wang, T.: A novel statistical measure for sequence comparison on the basis of K-word counts. *J. Theor. Biol.* **318**, 91–100 (2013)
11. Yano, M., Kato, Y.: Using hidden Markov models to investigate G-quadruplex motifs in genomic sequences. *BMC Genom.* **15**, S15 (2014)
12. Wu, T., Hsieh, Y., Li, L.: Statistical measures of DNA sequence dissimilarity under Markov chain models of based composition. *Biometrics* **57**, 441–448 (2001)
13. Pham, T.D., Zuegg, J.: A probabilistic measure for alignment free sequence comparison. *Bioinformatics* **20**, 3455–3461 (2004)
14. Jeong, B.S., Bari, A.G., Reaz, M.R., Jeon, S., Lim, C.G., Choi, H.J.: Codon-based encoding for DNA sequence analysis. *Methods* **67**, 373–379 (2014)
15. He, Q., Bai, X., Liu, X., Xu, N., et al.: Protein and mRNA expression of CTGF, CYR61, VEGF-C and VEGFR-2 in bone marrow of leukemia patients and its correlation with clinical features. *Chin. Assoc. Pathophysiol.* **22**, 653–659 (2014)
16. Zhang, Y., Qiu, J., Su, L.: Comparing RNA secondary structures based on 2D graphical representation. *Chem. Phys. Lett.* **458**, 180–185 (2008)
17. Liu, L., Wang, T.: On 3D graphical representation of RNA secondary structures and their applications. *J. Math. Chem.* **42**, 595–602 (2007)
18. Yu, H., Huang, D.: Graphical representation for DNA sequences via joint diagonalization of matrix pencil. *IEEE J. Biomed. Health Inform.* **17**, 503–511 (2013)
19. Tian, F., Wang, S., Wang, J., Liu, X.: Similarity analysis of RNA secondary structure with symbolic dynamics. *J. Comput. Res. Dev.* **50**, 445–452 (2013)
20. Wang, S., Tian, F., Qiu, Y., Liu, X.: Bilateral similarity function: a novel and universal method for similarity analysis of biological sequences. *J. Theor. Biol.* **265**, 194–201 (2010)
21. Liu, Z., Liao, B., Zhu, W.: A new method to analyze the similarity based on dual nucleotides of the DNA sequence. *Match-Commun. Math. Comput. Chem.* **61**, 541–552 (2009)
22. Liu, Z., Liao, B., Zhu, W., Huang, G.: A 2D graphical representation of DNA sequence based on dual nucleotides and its application. *Int. J. Quantum Chem.* **109**, 948–958 (2009)
23. Bai, F., Li, D., Wang, T.: A new mapping rule for RNA secondary structures with its applications. *J. Math. Chem.* **43**, 932–942 (2008)

24. Li, W., Fu, L., Niu, B., Wu, S., Wooley, J.: Ultrafast clustering algorithms for metagenomic sequence analysis. *Brief. Bioinform.* **13**, 656–668 (2012)
25. Yang, J., et al.: Entropy-driven DNA logic circuits regulated by DNAzyme. *Nucl. Acids Res.* (2018). <https://doi.org/10.1093/nar/gky663>
26. Wang, B., et al.: Constructing DNA barcode sets based on particle swarm optimization. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **15**, 999–1002 (2018)
27. Pan, L., Wang, Z., Li, Y., Xu, F., Zhang, Q., Zhang, C.: Nicking enzyme-controlled toehold regulation for DNA logic circuits. *Nanoscale* **9**(46), 18223–18228 (2017)
28. Wang, B., Xie, Y., Zhou, S., Zheng, X., Zhou, C.: Correcting errors in image encryption based on DNA coding. *Molecules* (2018). <https://doi.org/10.3390/molecules23081878>



Refrigerant Capacity Detection of Dehumidifier Based on Time Series and Neural Networks

Gang Peng^{1,2,3}, Zuhuang Yang^{1,2(✉)}, and Min Wang^{1,2}

¹ School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China

penggang@hust.edu.cn, yangzh10th@163.com,
95681622@qq.com, wm526@163.com

² Key Laboratory of Image Processing and Intelligent Control of Education Ministry, Wuhan, China

³ Shenzhen Institute, Huazhong University of Science and Technology, Shenzhen 518060, China

Abstract. Refrigerant leakage is common in the use of dehumidifier, leading to decrease of dehumidification ability. To detect refrigerant capacity effectively and maintain it timely, a neural network with deep learning skills was proposed based on time series. As input of networks, the time series consists of the operating parameters of the dehumidifier at multiple time points. To determine the refrigerant capacity, the proposed method combines the outputs on the neural networks of all the time series examples in a single run of dehumidifier. As the result suggests, the proposed method is capable of detecting the refrigerant capacity with a low missing rate and high accuracy, which improves the maintenance mechanism of the dehumidifier and is of great significance to ensuring a comfortable air environment for user.

Keywords: Refrigerant capacity detection · Time series · Neural networks

1 Introduction

Dehumidifier is critical for maintaining air comfort and storage of equipment. Yet refrigerant is often leaked in the use of the dehumidifier because of the limited degree of sealing with various pipeline interfaces and other reasons. To maintain the performance of dehumidifier, it is important to timely detect and charge refrigerant capacity.

At present, there are few papers on fault detection and diagnosis in the field of dehumidifier. However, a large number of methods are involved, e.g., rule reasoning [1, 2], expert systems [3], mean clustering [4], neural networks [5–9], and ARX model [10–12]. [1] yields the inference rules for faults by comparing the parameters of normal and abnormal states of dehumidifier. [2] extracts fuzzy rules with training data and optimizes these rules using genetic algorithm. Yet there are less data for rule extraction, and the date is too ideal. Thus, the yielded rules in [1, 2] are not applicable. [3] employs the expert system to establish the knowledge base that incorporates the principle knowledge, experience knowledge and on-the-spot knowledge, and forms the fault tree

to make reasonable judgment. This method is applied in most dehumidifier products, whereas it has a high missing rate and an insufficient accuracy due to the complexity of the dehumidifier system and the limited experience of experts. [4] combines the data of normal state with 9 sorts of fault for clustering and get the centers of 10 cluster to classify the new samples by calculating the distance. Yet this method does not apply to massive data since different sorts of data will be mixed together, resulting in inefficient clustering. [5–7] build the relationship between failures and parameters of dehumidifier before the use of neural networks. This means the types of failures can be already determined by dehumidifier parameter indices, and neural networks are just another expression of this relationship. The built relationship based on a small number of condition is not objective enough, and the three papers do not investigate the mapping relationship under neural networks. [8] predicts the COP value of the dehumidifier using GRNN and diagnoses the fault in line with the value. However, it relies solely on the COP value, and the robustness is obviously weak. Thus in [9], two additional parameters have been introduced for fault classification. In fact, the calculation of COP value is subject to error so that the method has limited practical results. [10] builds the relationship between input and output of dehumidifier by implementing ARX, one of black model, and the case when the error between the actual output of the system and the ARX output exceeds the threshold is considered as the fault state. To better fit the nonlinear relationship, [11] introduces LS-SVM algorithm to optimize ARX model. Likewise, to optimize the parameters of LS-SVM, [12] introduces adaptive genetic algorithm based on [11]. Yet [12] does not improve much but more complicated. And as the author stated, LS-SVM applies more to small samples, whereas the state of the dehumidifier during the operation is complex and inevitably creates a large amount of data.

The above researches are almost based on small samples, which reach up to 1000. However, the actual situation will be much more complicated, which explains why the expert experience is limited. Aiming at the above noted problems, this paper proposed a neural networks method for refrigerant capacity detection of humidifier based on time series. Neural networks have an excellent ability to extract valid features from massive data and time series contains more information about humidifier operation. In comparison with the noted method, we employed a multi-layer networks with deep learning skills, and utilized time series as input to yield a wonderful result.

The rest of paper is organized as follows. Section 2 describes how the time series is sampled from the data. Section 3 describes the structure of our neural networks as well as some details. Finally, Sects. 4 and 5 present the test results and conclusions, respectively.

2 Sampling of Time Series

Under the small amount of data and ideal working conditions, the state values of dehumidifier at some certain moments are very valuable. Yet according to expert experience and our tests, refrigerant capacity is hard to determine under massive data. When the machine is running, the parameters and its statistical characteristics in a single time point of dehumidifier are sophisticated. These are likely to be very similar

even the same under various refrigerant capacities, which are divided into 9 levels from 20% to 100%, and 50% or less is considered as a malfunction. Accordingly, we consider the characteristics of the data over time as a feature, i.e., time series.

For time series, which time point and how many time points to choose should be considered. Using image recognition technology, we adopted the sliding windows as the data sampling method (Fig. 1).

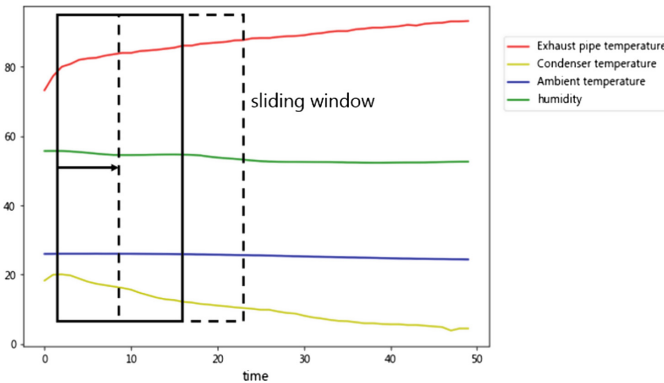


Fig. 1. Data over time and sliding window

In the figure above, we set a certain length of time to the window and sample a certain amount of data in the window at regular interval. For instance, the sampling time points will be 1, 3, 5, 7, 9 in the window when the window length is 10, and we sample 5 data with interval 2. Moreover, each time point data consists of all the useful operating parameters of dehumidifier. After one sampling is done, the window moves forward through a certain steps to sample the next example until to the end. There are numerous settings of the window resulting in various time series, maybe most of them are useless for our target. However, we can determine the time series we need after testing. This means we can find properties that really identify different levels of the refrigerant. In our proposed method, the length of window is 31, the interval equals to 6 (10 s between adjacent time points), and the step moves to 6, which has the optimal performance.

The data collection is uniformly limited within a certain period after the machine start-up in accordance with test result of sliding windows since the duration of each run of dehumidifier is different, and the detection does not need to be performed in real time. In such a way, it is only necessary to detect the refrigerant capacity within a certain time range every time the dehumidifier turns on, even if it is driven by the user for a long time. In the meantime, the numbers of examples acquired in each run of dehumidifier via sliding windows are equal because of the limited time, which does not lead to a serious imbalance in the proportion of examples at various refrigerant levels. In our method, we set the time range as 30 min.

Our research is conducted on the dehumidifier products of the project partners, and the major measurable parameters include exhaust pipe temperature (EPT), condenser

temperature (CT), ambient temperature as well as relative humidity. EPT and CT will generally return to ambient temperature after the dehumidifier stops working for a while. Yet before the cooling is completed, users may restart the machine again in practice, which makes the starting point of the data random to some extent at each run (Fig. 2).

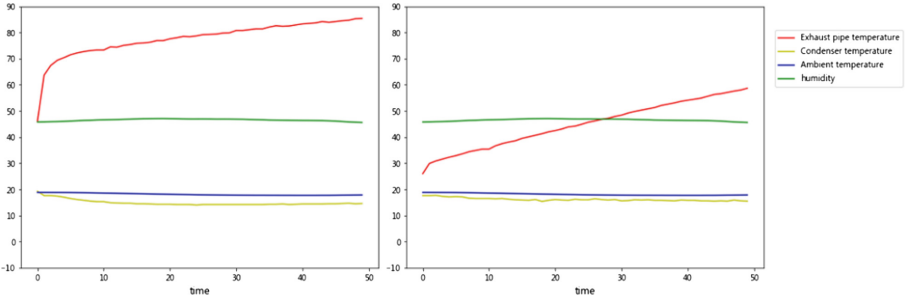


Fig. 2. The randomness of the starting point of EPT. The left and right part of figure are two runs of dehumidifier (only showing the first 50 time points) in the identical external conditions, yet their EPI parameter initial values are very different.

This phenomenon makes the example more complex, making the detection problem more difficult to solve. Accordingly, we substitute the following operations to the parameter of EPT and CT for each run of dehumidifier:

$$x_t \leftarrow x_t - x_0, t \in [0, T] \quad (1)$$

Where x_t denotes the value of EPT or CT at time point t , x_0 is initial value, and the T is the length of single run time of dehumidifier. As a result, the data will be translated to a new location with initial value 0, and the two parameters of the input of neural networks will no longer be numerical but trend.

3 Neural Networks Structure

Since the dehumidifier is a nonlinear system, yet there has been no accurate mathematical model to describe it, and it is easy to get massive operating data. Neural networks is the most suitable method among all intelligent fault diagnosis technologies, particularly for multidimensional input. Neural networks can approximate any continuous nonlinear function with arbitrary precision. Furthermore, it can learn from, summarize and promote the examples with a strong self-adaptability [13].

Besides the time series, the natural logarithm of each time point response to data also serves as network input, reducing the misjudgment between similar examples at various windows. Hence, the network has 30 input nodes (each example includes 6 time points with 5 parameters at each time point), and 9 at the output corresponding to

refrigerant capacity levels. To fit the complex nonlinearity, we finally implement a structure consisting of four hidden layers, i.e., 30-25-15-10. At the end of the neural networks, we install the softmax layer to convert the output to probability so that the selected loss function is cross-entropy.

For each hidden layer, the input distribution is constantly changing under the parameters update, impacting the training speed of the networks. The more the layers, the more serious the phenomenon will be. To solve this problem, we employ the skill known as batch normalization (BN) [14]. It performs normalization between each hidden layer and its activation function, to prevent the dispersion of the gradient while keeping the distribution unchanged. The role of BN in our method is shown in Fig. 3.

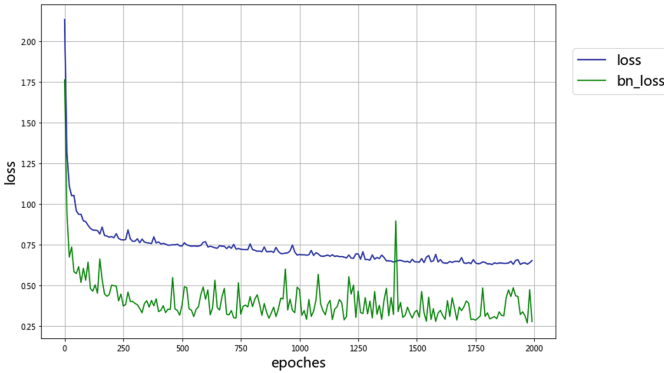


Fig. 3. The contrast between whether deployed BN.

Figure 3 shows that BN accelerates network convergence and improves network performance (By Adam optimization algorithm with learning rate 0.001).

4 Experiment Results and Analysis

In this section, we test the proposed method in the data generated by dehumidifiers, which are charged by different levels of refrigerant capacity and operating in various environments, e.g., laboratories, dormitories and office. Also, more than 160 work conditions are considered. The training data undergo stratified sampling in line with the ratio of 7:3, which fall into training set and validation set. The training set is employed to train the model, while the validation set is to test the performance of the model for optimal selection. The training result is shown in the figure below.

Figure 4 illustrates that the accuracy of 80% and 90% refrigerant capacity is extremely low, leading to a low overall accuracy. Therefore, the corresponding outputs are analyzed.

The error is serious but most of the predictions are within $\pm 10\%$ of the true label, as shown in the Fig. 5. Subsequently, we repeated the training many times, the results are similar and sometimes be worse. This means some levels accuracy increase while

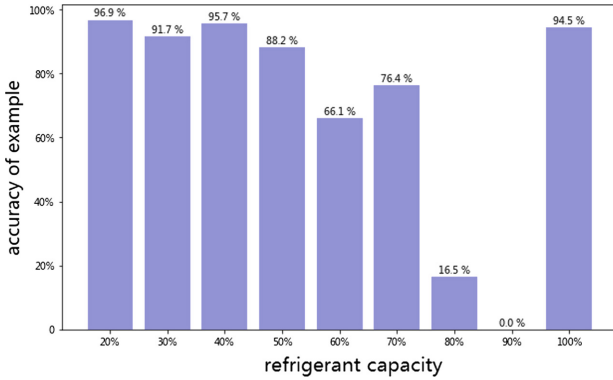


Fig. 4. The training set result with total accuracy 76%

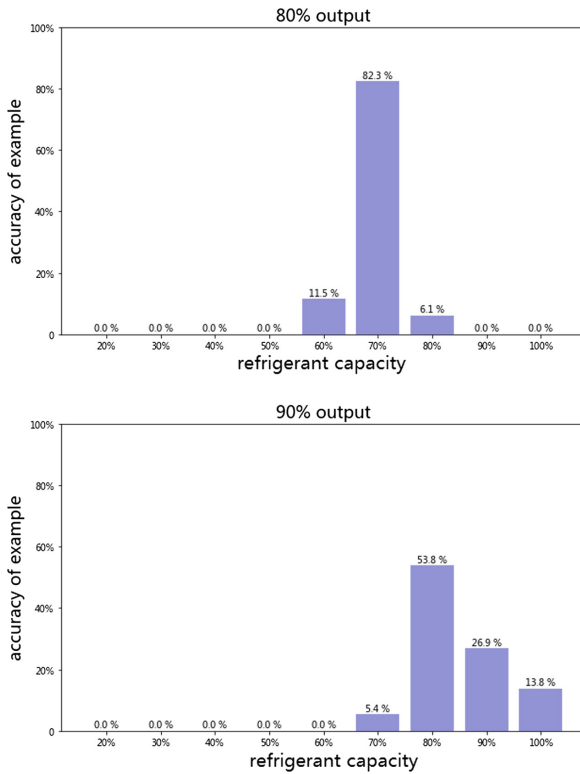


Fig. 5. The output of 80% and 90% refrigerant capacity

some others decline, and the accuracy distribution is similar to that in Fig. 4. However, the overall accuracy will be over 90% when the error of $\pm 10\%$ of refrigerant capacity is allowed. By comparing the visualization of examples and consulting the system

engineers of the project partner, it is concluded that: different levels of operational data may be theoretically very similar, especially adjacent levels, which may be attributed to the lack of available measurable parameters.

Accordingly, we adjust our performance measure of method. Given that the primary target is to determine whether the capacity is less than or equal to 50% which is considered to be faulty, we summarize all the output probabilities above 50% for each example. When the summated probability of an example is greater than 0.7 (normal) or less than 0.3 (fault), it will be recorded as a valid example. When the number of valid examples of one type, i.e., normal or fault, takes up more than 70% of the total examples in a run (25 samples are sampled by the sliding windows for each run), we will give judgments to the run. The result of our final model is illustrated as follows (Fig. 6).

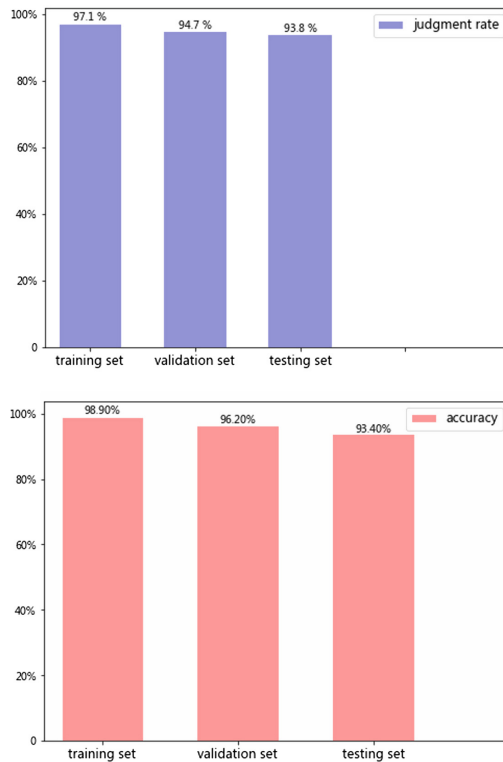


Fig. 6. The final result of proposed method.

As the results suggest, our proposed method has a very low missing rate and a high accuracy for those runs we have confidence to make judgments. In contrast, we employ the state of dehumidifier in single time point (after average) as networks input for testing, which is applied by the noted method. Moreover, the test result (we sample 18 samples for each run by interval 10) is shown as follow.

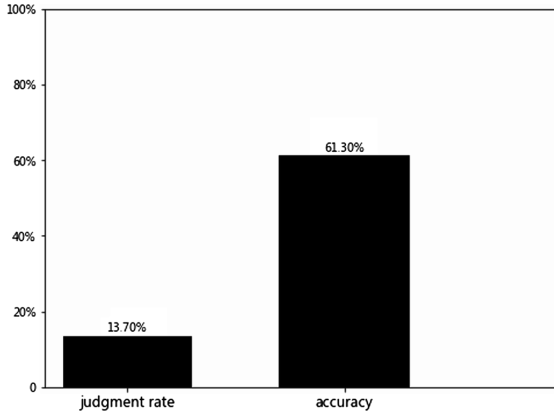


Fig. 7. The result of single time point.

Figure 7 shows that it is difficult to reliably distinguish fault and normal under large amounts of data with single time point. By comparing the above results, our proposed method in this paper is effective for detecting whether the refrigerant capacity is below 50%. The error originates from the similar data at different refrigerant capacities as well as the limitedness of measurable parameters. Besides, the results also prove that the dehumidifier system is hard to describe mathematically, and expert experience is difficult to gain.

5 Conclusions

In this paper, we propose a method of detecting the refrigerant capacity based on time series and neural networks, which can effectively determine whether the refrigerant capacity level is below the threshold. Compared with existing method, we consider the various conditions of refrigerant capacity more comprehensively and fully exploit the feature of time series to obtain the valuable nonlinear mapping relationship. As the test results suggest, the method with a low missing rate and a high accuracy only needs to perform the detection within the first certain period. Yet in the training and testing process, the data is overall generated from the same type of dehumidifier so that we should further improve the method's versatility. Furthermore, we will combine features of different sizes of sliding windows other than one in our future work.

Acknowledgments. This paper was supported by foundation research project No. JCYJ20150730103208405 of Shenzhen Science and Technology Innovation Committee, and open research project of State Key Laboratory of Air-conditioning Equipment and System Energy Conservation, China.

References

1. Ren, H., Liu, S., Gao, Y.: Decompression machine fault diagnosis based on S600 building automation system. *Refrig. Air Cond. Electr. Power* **28**(1), 11–13 (2007)
2. Gao, Y., Liu, S., Zhang, Z.: An improved fault identification method based on genetic fuzzy rules. *Coal Mine Mach.* **28**(12), 181–183 (2007)
3. Liang, J.: Study on the real-time fault diagnosis expert system of dehumidifier based on CLIPS. *Refrig. Air Cond.* **3**, 008 (2010)
4. He, B., Liu, S.: Faults diagnosis for dehumidifier based on genetic fuzzy C-means clustering algorithm. *Refrig. Air Cond. Electr. Power Mach.* **4**, 005 (2009)
5. Wang, X., Liu, S., Liu, X.: Application of artificial neural network in fault diagnosis of dehumidifier. *Mech. Electr. Eng. Technol.* **36**(7), 62–63 (2007)
6. Huang, H.: Fault diagnosis of dehumidifier based on RBFNN. *Refrig. Air Cond.* **4**, 019 (2011)
7. Zhang, Q., Wu, Y., Xu, J.: Fault diagnosis and life prediction of dehumidifier based on genetic neural network. *Environ. Eng.* **1**, 78–83 (2017)
8. He, W., Gao, Y.: COP prediction of dehumidifier based on GRNN and genetic algorithm and its application in fault diagnosis. *Refrig. Air Cond. (Beijing)* **9**(5), 17–20 (2009)
9. Huang, Z., Liu, H., Liu, S., et al.: Research on faults diagnosis of dehumidifier based on COP and improved PNN. *Refrig. Air Cond. (Sichuan)* **24**(5), 66–69 (2010)
10. Gao, Y., Liu, S., Zhang, Z.: Application of ARX model to fault diagnosis of dehumidifier. *Comput. Simul.* **25**(2), 332–335 (2008)
11. Liu, H., Liu, S., Gao, Y., et al.: Faults diagnosis for dehumidifier based on LS-SVM ARX model. *Refrig. Air Cond. Electr. Power* **31**(5), 47–51 (2010)
12. Gao, Y., Liu, S., Li, F., et al.: Fault detection and diagnosis method for cooling dehumidifier based on LS-SVM NARX model. *Int. J. Refrig* **61**, 69–81 (2016)
13. Zhu, D., Yu, S.: A summary of knowledge-based fault diagnosis methods. *J. Anhui Univ. Technol. (Nat. Sci.)* **19**(3), 197–204 (2002)
14. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. *arXiv preprint [arXiv:1502.03167](https://arxiv.org/abs/1502.03167)* (2015)



An Improved Artificial Bee Colony Algorithm and Its Taguchi Analysis

Yudong Ni¹, Yuanyuan Li^{2,3}, and Yindong Shen^{2,3}(✉)

¹ School of Mathematics, Hefei University of Technology, Hefei 230009, China

² School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China

yindong@hust.edu.cn

³ Key Laboratory of Education Ministry for Image Processing and Intelligent Control, Wuhan 430074, China

Abstract. The artificial bee colony (ABC) algorithm is one of well-known evolutionary algorithms, which has been successfully applied to many continuous or combinatorial optimization problems. To increase further its convergence speed and avoid being trapped in local optimum, this paper proposes an improved ABC algorithm (IABC), which aims to enhance diversification of search at each stage of the ABC algorithm. Firstly, a chaotic mapping rule is established by introducing a chaos operator into the initial position generation rules in order to ensure the ergodicity of initial positions. Then, an isometric contraction parallel search rule is devised, based on which a neighborhood search on initial positions is performed to enhance the convergence speed and the local search ability. Next, a parallel selection strategy is developed by using roulette and reverse roulette simultaneously, which allows selecting poor positions to escape from local optimum. Meanwhile, a global updating mechanism based on gravitational potential field is developed, which can guide the rejection and generation of positions to accelerate the convergence of the algorithm. The computational results show that the IABC can improve the convergence speed and solution quality without falling into the local optimum prematurely. Finally, a further analysis on the IABC is conducted using the Taguchi method, which focuses on the factor level setting related to the following key factors: chaotic mapping rules in the initial position generation rules, isometric contraction parallel search rules, parallel selection strategies and the update threshold in the global updating mechanism. The results display that the optimal combination of factor levels has been achieved in the IABC.

Keywords: Artificial bee colony · Chaotic map · Isometric contraction search
Reverse selection · Potential field · Taguchi analysis

1 Introduction

Artificial bee colony (ABC) algorithm belongs to the family of evolutionary algorithms, which have been developed by simulating the behaviors of species in their evolutionary process [1–3]. The ABC algorithm was first proposed by Karaboga in 2005 based on the intelligent foraging behaviour of honey bee swarm [4]. Since then, it

has been successfully applied to many continuous or combinatorial optimization problems due to its multi-role conversion and unique selection mechanism, which can ensure that the algorithm has a lower computational complexity and faster convergence rate. However, along with the expansion of the solution space dimension and the complexity of the problem such as multi-target search and real-time planning, the ABC algorithm reveals some defects, such as low convergence speed when handling unimodal problems [5], and easy to be trapped in local optimal [6]. The reasons include that, as well known, it generates a new scheme based on the current scheme, which is usually good at exploration but poor at exploitation. Therefore, accelerating the convergence and avoiding being trapped in local optimum become the two main goals of improving the ABC algorithm [7–16]. For example, Zhu and Kwong [7] incorporated the information of global best solution into the solution search equation to improve the exploitation, which aimed to avoid being trapped in local optimum, Karaboga and Kaya [8] proposed arithmetic crossover to gain the rapid convergence feature, and Alata [9] proposed the chaotic ABC algorithm, which focused on avoiding being trapped in local optimum, but needed more evaluations in chaotic search. This paper aims to achieve both of these goals simultaneously. The key factors governing the ABC's efficiency and performance are pointed out, based on which an improved ABC algorithm, called IABC, is developed. To investigate the IABC further, Taguchi analysis is employed to verify the optimal combination of key factor levels.

Taguchi method was invented by Genichi Taguchi in the late 1940s based on multi-factor experimental designs [17]. The orthogonal table is the basic tool for analysis of various factors, which can get the optimal combination of factor levels based on few trials [18]. Therefore, it has been widely applied in engineering [19]. This paper will extend the application of Taguchi method by employing it to analyze the rules and strategies devised in the IABC algorithm.

The remainder of the paper is structured as follows. Section 2 gives a description on the traditional ABC algorithm, in which four key factors governing the algorithm efficiency and convergence speed are pointed out. Section 3 presents the IABC algorithm, which improves the ABC algorithm on each key factor intensively. Section 4 shows the computational results of this improved algorithm and conducts further TAGUCHI orthogonal analysis. Section 5 contains some concluding remarks and possible future work.

2 Description of Artificial Bee Colony Algorithm

The ABC algorithm consists of three essential components: food sources, employed bees and unemployed bees. The last two components search for rich food sources, which is the first component, close to their hive. The number of employed bees is equal to the number of food sources around the hive. Employed bees go to their food source and come back to hive and dance on this area. Unemployed bees consist of two groups of bees: onlookers and scouts. Onlookers watch the dances of employed bees and choose food sources depending on dances. Scouts fly and choose the food sources randomly without using experience.

To apply ABC to solving optimization problems, a solution is presented as a food source and the artificial bees randomly discover a population of initial solutions and then iteratively improve them by employing the strategies: moving towards better solutions by means of a neighborhood search mechanism while abandoning poor solutions. The main steps of ABC include:

(1) Initialization

Suppose there is a set N_p of employed bees ($N_p = \{1, 2, \dots, N\}$) searching foods in search space with a set D_s of dimensions ($D_s = \{1, 2, \dots, D\}$), i.e. N initial positions of food sources are to be randomly generated in D -dimensional space, each of which is assigned to a employed bee. Any position $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ can be generated by using Formula (1), where $x_{\max j}$ and $x_{\min j}$ are the upper and lower bounds on dimension j of search space, and $rand$ denotes a random number in range (0, 1).

$$x_{ij} = x_{\min j} + rand(x_{\max j} - x_{\min j}) \quad (1)$$

(2) Updating food sources using neighborhood search

Each employed bee performs a neighborhood search around the hive from its current position X_i to obtain a new position $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ and updates X_i according to greedy mechanism. A move from X_i to V_i in the neighborhood search can be defined as Formula (2), where $j \in D_s$ and $k \in N_p$ are randomly chosen, and k has to be different from i .

$$v_{ij} = x_{ij} + rand(x_{kj} - x_{ij}) \quad (2)$$

(3) Selecting food sources using roulette

Each onlooker bee on the dance area chooses a food source from those found by employed bees and then updates the food source using the same way as employed bees did in Step (2). The food source selection is based on the roulette mechanism, in which the probability is employed by using Formula (3), where fit_i is the fitness value of the food source i .

$$P_i = fit_i / \sum_{i=1}^N fit_i \quad (3)$$

(4) Global updating

All the food sources are checked and those without update for ϵ times in succession are to be abandoned whilst the corresponding employed bees become scout bees. Each

of the scout bees randomly generates a new food source, which can be done using Formula (1) as in Step (1), meanwhile, they convert back into employed bees. Then, go to Step (2) until the termination condition is satisfied.

3 An Improved Artificial Bee Colony Algorithm

From the steps of the ABC algorithm described in the previous section, we can point out the following main factors, which govern the efficiency and convergence speed of the ABC algorithm: (1) initial position generation rule, (2) search rule for employed bees to update positions, (3) selection strategy for onlooker bees to select food sources, and (4) global updating mechanism to govern the rejection and generation of positions. Therefore, this section aims to develop an improved ABC algorithm, called IABC, which tries to enhance the ABC algorithm at each of the four factors intensively.

3.1 Initial Position Generation Rule Based on Chaotic Map

In the ABC algorithm, the diversity of initial solutions is hard to be guaranteed, although they are generated randomly. To ensure the ergodicity of initial solutions, chaotic map may be applied to generating initial solutions.

Chaos is a type of movement phenomenon, which exists widely in the nonlinear systems of nature. The process seems to be chaotic and disorderly, but it has the delicate internal structure. Taking advantages of the characteristics of chaotic variables, such as ergodicity, randomness and initial conditional sensitivity [20–22], we may improve the ABC by establishing a chaotic map based initial position generation rule, which constructs an one-to-one mapping on interval (0,1) to ensure the ergodicity of positions efficiently. Using this one-to-one mapping to generate a D -dimensional chaos operator, then introduce the chaos operator as the weight to generate a pair of symmetric positions in the search space, and take the better one as an initial position for employed bee. Depend on this rule to generate N initial positions in turn.

Suppose K is the threshold of chaos mapping, and $ch_i^{(K)} = (ch_{i1}^{(K)}, ch_{i2}^{(K)}, \dots, ch_{iD}^{(K)})$ is the i -th chaos operator. The initial position generation rule based on chaotic map can be devised as follows, in which the formulas (5) and (6) make the positions V_i and OV_i symmetric about the search space center which may enhances the ergodicity of initial positions.

Step 1: Generate a random vector $ch_i^{(0)} = (ch_{i1}^{(0)}, ch_{i2}^{(0)}, \dots, ch_{iD}^{(0)})$, $ch_{ij}^{(0)} \in (0, 1)$

Step 2: Chaos mapping iteratively by Formula (4) until getting the chaos operator $ch_i^{(K)} = (ch_{i1}^{(K)}, ch_{i2}^{(K)}, \dots, ch_{iD}^{(K)})$.

$$ch_{ij}^{(k+1)} = \cos\left(\frac{\pi}{2} \times ch_{ij}^{(k)}\right) \quad k = 1, 2, \dots, K \tag{4}$$

Step 3: Introduce the chaos operator $ch_i^{(K)}$ as the weight to generate position $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ by using Formula (5).

$$v_{ij} = (1 - ch_{ij}^{(K)})x_{\min j} + ch_{ij}^{(K)}x_{\max j} \quad (5)$$

Step 4: Generate another position $OV_i = (ov_{i1}, ov_{i2}, \dots, ov_{iD})$ by using Formula (6).

$$ov_{ij} = ch_{ij}^{(K)}x_{\min j} + (1 - ch_{ij}^{(K)})x_{\max j} \quad (6)$$

Step 5: Selected the better position from $\{V_i, OV_i\}$ as the initial position X_i for the i -th employed bee.

3.2 Isometric Contraction Parallel Search Rule

In the local search, each employed bee or onlooker bee blindly chooses a certain dimension component near the location of the food source to update position while using a random number in the range $[0, 1]$ to control the neighborhood. The blindness and randomness in the local search process reduce the search efficiency and convergence speed of the algorithm [23]. This section proposes an isometric contraction parallel search rule, in which an adaptively contractive neighborhood and a best-position guidance are devised.

Instead of the random coefficient $rand$ in formula (2), a new isometric contraction coefficient is devised as $\alpha = (G-t)/G$, where G denotes the maximum number of iterations, t denotes the current iteration. This coefficient can make the neighborhood contract isometrically and adaptively along with the iteration increases. Furthermore, instead of a random position x_{kj} in formula (2), the current optimal position $x_{best,j}$ is employed to enhance the intensification of the neighborhood search.

Given the i -th position $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ and suppose $X_{best} = (x_{best,1}, x_{best,2}, \dots, x_{best,D})$ is the current optimal position, a parallel search is conducted to obtain two new positions $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ and $OV_i = (ov_{i1}, ov_{i2}, \dots, ov_{iD})$ by using Formula (7), where $j \in D_s$ and $k \in N_p (k \neq i)$, are randomly chosen.

$$\begin{cases} v_{ij} = x_{ij} + \alpha(x_{best,j} - x_{ij}) \\ ov_{ij} = x_{ij} + \alpha(x_{k,j} - x_{ij}) \end{cases}, \alpha = (G - t) / G \quad (7)$$

Finally, the i -th position is updated with the best position in $\{V_i, OV_i, X_i\}$.

3.3 Selection Strategy Based on Parallel Mechanism

The selection strategy in the ABC algorithm is dependent on roulette mechanism, i.e. a food source with bigger fitness value has a higher probability to be selected by onlooker bees. This will lead to the rapid evolution of the evolutionary process to the high concentration of the location, and induce the ABC into premature convergence. Therefore, in order to avoid the algorithm being premature and falling into the local optimal as far as possible, a parallel selection strategy is constructed.

The parallel selection strategy is developed by using roulette and reverse roulette simultaneously. The reverse roulette works depending on the principle that the greater the reciprocal of the fitness value, the greater the probability that the position is selected [24]. This is helpful for onlooker bees to explore the positions with poor fitness values in order to maintain the diversity of the population.

The parallel selection strategy can be devised as follows.

Firstly, an onlooker bee chooses one position X_i from all current positions based on the roulette mechanism. The probability of roulette is given by using Formula (3).

Then, the onlooker bee chooses another position X_j from all current positions based on the reverse roulette mechanism. The probability of the reverse roulette is developed by using Formula (8).

$$Q_i = (1/fit_i) / \sum_{j=1}^N (1/fit_j) \tag{8}$$

Finally, the onlooker bee searches and updates the positions X_i and X_j respectively.

3.4 Global Updating Mechanism Based on Potential Field

In the global updating phase of ABC, new positions are replenished randomly. To increase the optimization efficiency, a new global updating mechanism is proposed based on potential field, in which the gravitational effect of the target position is introduced to generate new positions. The basic idea is that the position i will be discarded if it is not updated for consecutive ε times, meanwhile, the corresponding i -th employed bee becomes a scout bee. Assuming that the gravitational potential of the j -th space is $q(j)$, a scout bee generates a new position according to a reliable ratio, which relies on the gravitational effect of the target position, rather than generates a new position randomly as done in the ABC algorithms [25].

The improved global updating mechanism is devised as follows.

Step 1: Introducing the ratio of the j -th space for scout bees by using Formula (9).

$$Gra_j = q(j) / \sum_{j=1}^D q(j) \tag{9}$$

Step 2: Any position i is discarded if it is not updated for consecutive ε times, the corresponding employed bee (now served as a scout bee) needs to generate a new position $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ based on the above probability as described in Formula (10).

$$x_{ij} = x_{minj} + Gra_j(x_{maxj} - x_{minj}), j = 1, 2, \dots, D \tag{10}$$

3.5 Main Steps of IABC Algorithm

Based on the above explanation, the pseudo-code of IABC algorithm is given below:

01: Set the position size N , the maximum number of iterations G , and updated threshold ε , $limit_i$ is the non-improvement number of position X_i .

{Generating initial positions for employed bees based on chaotic map}

03: **for** $i=1$ to N **do**

04: **for** $j=1$ to D **do**

05: Randomly initialize variables $ch_{ij}^{(0)} \in (0,1)$

06: **for** $k=1$ to K **do**

07: $ch_{ij}^{(k+1)} = \cos\left(\frac{\pi}{2} \times ch_{ij}^{(k)}\right)$

08: **end for**

09: Produce the positions component v_{ij} and ov_{ij} by using Formulas (5) and (6)

10: **end for**

11: Select the better position from $\{V_i, OV_i\}$ as the initial position X_i

12: **end for**

13: **While** (the iteration threshold is not met, namely $g < G$) **do**

{Employed bees search a position by using the isometric contraction parallel rule.}

15: **for** $i=1$ to N **do**

16: Randomly choose $j \in D_s, k \in N_p (k \neq i)$

17: Generate new positions V_i and OV_i by using Formula (7)

18: Selected the best position from $\{V_i, OV_i, X_i\}$ to update the i -th position

19: **if** position X_i improve **then** $limit_i=0$

20: **else** $limit_i=limit_i+1$

21: **end for**

{Onlooker bees select positions based on parallel mechanism.}

23: Calculate the probabilities P_i and Q_i by using Formulas (3) and (8) respectively

24: $t=0, i=1, j=1$

25: **repeat**

26: **if** $random < P_i$ and $random < Q_j$ **then**

27: search and update position X_i and position X_j , and update $limit_i$ and $limit_j$

28: $t=t+1$

29: **end if**

30: **until** ($t=N$)

{Global updating based on potential field.}

32: Calculate the ratio of the j -th space for scout bees by using Formula (9)

33: **if** $limit_i > \varepsilon$ **then**

34: Replace X_i by using Formula (10)

35: $g=g+1$

36: **end if**

37: **end while** ($g=G$)

4 Computational Results and Taguchi Orthogonal Analysis

The IABC algorithm has been implemented by using MATLAB. To verify its performance, it has been tested on a number of complex functions in comparison with ABC algorithm. Eight complex functions, a half of which are uni-modal functions while another half are multi-modal functions, are selected from the document [1] as the test problems listed in Tables 1 and 2.

Table 1. Uni-modal functions used in experiments

Function	Expression	Search range	Optimal solution	Min
f_1	$f(x) = \sum_{i=1}^D x_i^2$	$[-100, 100]^D$	$(0, 0, \dots, 0)$	0
f_2	$f(x) = \sum_{i=1}^D [x_i + 0.5]^2$	$[-100, 100]^D$	$(0, 0, \dots, 0)$	0
f_3	$f(x) = \sum_{i=1}^D ix_i^2$	$[-100, 100]^D$	$(0, 0, \dots, 0)$	0
f_4	$f(x) = \sum_{i=1}^D (x_i - 1)^2 - \sum_{i=2}^D x_i x_{i-1}$	$[-100, 100]^D$	$(15, 28, 39, 48, 55, 60, 63, 64, 63, 60, 55, 48, 39, 28, 15)$	0

Table 2. Multi-modal functions used in experiments

Function	Expression	Search range	Optimal solution	Min
f_5	$f(x) = \sum_{i=1}^D [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$[-100, 100]^D$	$(0, 0, \dots, 0)$	0
f_6	$f(x) = -20 \exp(-0.2 \sqrt{\sum_{i=1}^D x_i^2 / D}) - \exp(\sum_{i=1}^D \cos 2\pi x_i / D) + 20 + e$	$[-100, 100]^D$	$(0, 0, \dots, 0)$	0
f_7	$f(x) = \sum_{i=1}^D (\sum_{k=1}^i x_k)^2$	$[-100, 100]^D$	$(0, 0, \dots, 0)$	0
f_8	$f(x) = \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos(\frac{x_i}{\sqrt{i}}) + 1$	$[-100, 100]^D$	$(0, 0, \dots, 0)$	0

In Table 1, each uni-modal function contains only one optimal solution without any other local extremum in the solution domain. They are served to test the convergence precision and convergence speed of the algorithm. In Table 2, all multi-modal functions contain multiple local extremum, which are more complex than uni-modal functions. They are served to test algorithm performance on getting global optimal solutions and the ability of avoiding premature.

For all the experiments presented in this paper, we set the population size $N = 50$, solution space dimension $D = 15$, renewal threshold $\varepsilon = 30$, the maximum number of iterations $G = 1000$, chaotic sequence length $K = 300$. Experiments are first carried out on ABC and the IABC algorithms respectively, and then, a further investigation on the combination of key factors is conducted by Taguchi orthogonal analysis.

4.1 Computational Results

Average results of 30 runs with different pseudo random number seeds are listed in Table 3, where $RPD = (ABC - IABC)/ABC * 100\%$, denotes the average relative percentage deviation of the IABC results over the ABC results.

It can be seen from Table 3: by using IABC, the best, the mean and the worst solutions are improved by 121.76%, 108.44% and 81.68% respectively, which demonstrates that the IABC has superior convergence precision; the RPD of the average convergence time (average CPU time) is 48.57% which indicates that the IABC converges faster; and the RPD of variance is 65.91% which shows that the IABC is quite stable.

To investigate further the IABC's performance on convergence intuitively, the evolutionary curves of the optimal values for eight test functions are displayed in Fig. 1.

It can be seen from Fig. 1 that the solid line continues to approach the optimal solution with a high rate of change before convergence, and the value of the solid line is better than the dotted line which indicates that the IABC is less likely to trap into local optimum. Therefore, the results in Table 3 and Fig. 1 show that the IABC improve the convergence speed and solution quality without falling into the local optimum prematurely.

4.2 TAGUCHI Orthogonal Analysis

Although the IABC's performance has been improved, we would like to investigate further on its level of factor combination, which is carried out by applying the Taguchi orthogonal method below.

An orthogonal table $L_n(j^i)$ is the basic tool of orthogonal analysis, where i is the number of factors, j is the number of levels for each factor and n is the number of level combination. To analyze the following four key factors: (A) the update threshold in the global updating mechanism, (B) chaotic mapping rules in the initial position generation rules, (C) search rules, and (D) selection strategies, we select three levels for each factor [26, 27] as displayed in Table 4. A level combination can be denoted by $A_1B_1C_3D_3$ as an example, where the digitals present the level numbers.

Since the function f_5 in Table 2 is a multimodal function and easy being trapped in local optima, it is selected to conduct the orthogonal test with an orthogonal table $L_9(3^4)$. The results are shown in Table 5, where R_k and R_m denote the extremum differences corresponding to the best results and iteration respectively.

Table 3. Average results of 30 runs with different pseudo random number seeds for ABC and the IABC

Function	Algorithm	Best	Worst	Mean	Variance	CPU Time/s
f_1	IABC	$3.46e-14$	$1.663e-08$	$9.945e-10$	$3.08e-14$	0.4206
	ABC	1.2869	8.2883	3.4883	4.1526	1.0727
f_2	IABC	$3.87e-13$	$1.589e-07$	$4.460e-09$	$1.076e-12$	0.4160
	ABC	1.4591	6.1626	3.4221	4.1609	1.0763
f_3	IABC	$1.68e-12$	$7.085e-08$	$5.377e-09$	$1.376e-08$	0.3958
	ABC	9.3437	46.4222	23.8385	8.9448	0.9995
f_4	IABC	-651.403	673.3760	-335.8408	342.5644	0.5703
	ABC	-295.527	$1.973e+03$	847.1913	561.9017	1.3603
f_5	IABC	6.4315	39.2470	17.3341	34.5678	0.4598
	ABC	74.2658	127.1254	103.5656	65.7906	1.1322
f_6	IABC	20.3739	20.7197	20.5858	$1.064e-08$	0.6500
	ABC	20.4688	20.8193	20.6532	$6.502e-02$	1.5134
f_7	IABC	1.8990	616.1780	222.8920	154.9182	0.8920
	ABC	$1.6e+03$	$5.006e+03$	$3.299e+03$	915.4248	2.0270
f_8	IABC	$1.50e-11$	0.0734	0.0168	0.0179	3.0718
	ABC	0.2576	0.7620	0.5615	0.1054	5.7011
Avg.	IABC	-77.8370	168.6993	-9.3765	66.5085	0.8595
	ABC	357.608	898.6475	537.7776	195.0682	1.6711
	RPD*	121.76%	81.68%	108.44%	65.91%	48.57%

Table 5 shows that the best values for A , B , C and D are $k_{1A}= 91.84$, $k_{1B}= 99.35$, $k_{3C}= 37.23$ and $k_{3D}= 94.04$ respectively. This means that the optimal level of factor combination is: $A_1B_1C_3D_3$. The extremum difference holds $R_{kc} > R_{kc} > R_{kc} > R_{kc}$, i.e. the factors affect the order of the best values: (main) $C \rightarrow A \rightarrow B \rightarrow D$ (minor). This indicates that the search rule is the main factor, the selection strategy is the minor factor. Similarly, the optimal level of factor combination on the iteration number is: $A_1B_3C_3D_3$; the factors that affect the iteration hold the order: (main) $D \rightarrow C \rightarrow A \rightarrow B$ (minor). This indicates that the selection strategy is the main factor while the chaotic mapping rule is the minor factor. It can be find that, A_1 , C_3 and D_3 are better for both of the two indicators, and B is a minor factor for iteration, so B_3 will be changed to B_1 . Finally, depend on the comprehensive balance, the optimal factor level is obtained as $A_1B_1C_3D_3$, which indeed is the four factor level combinations of IABC. The results display that the optimal combination of factor levels has been achieved in the IABC, and its optimization results are better.

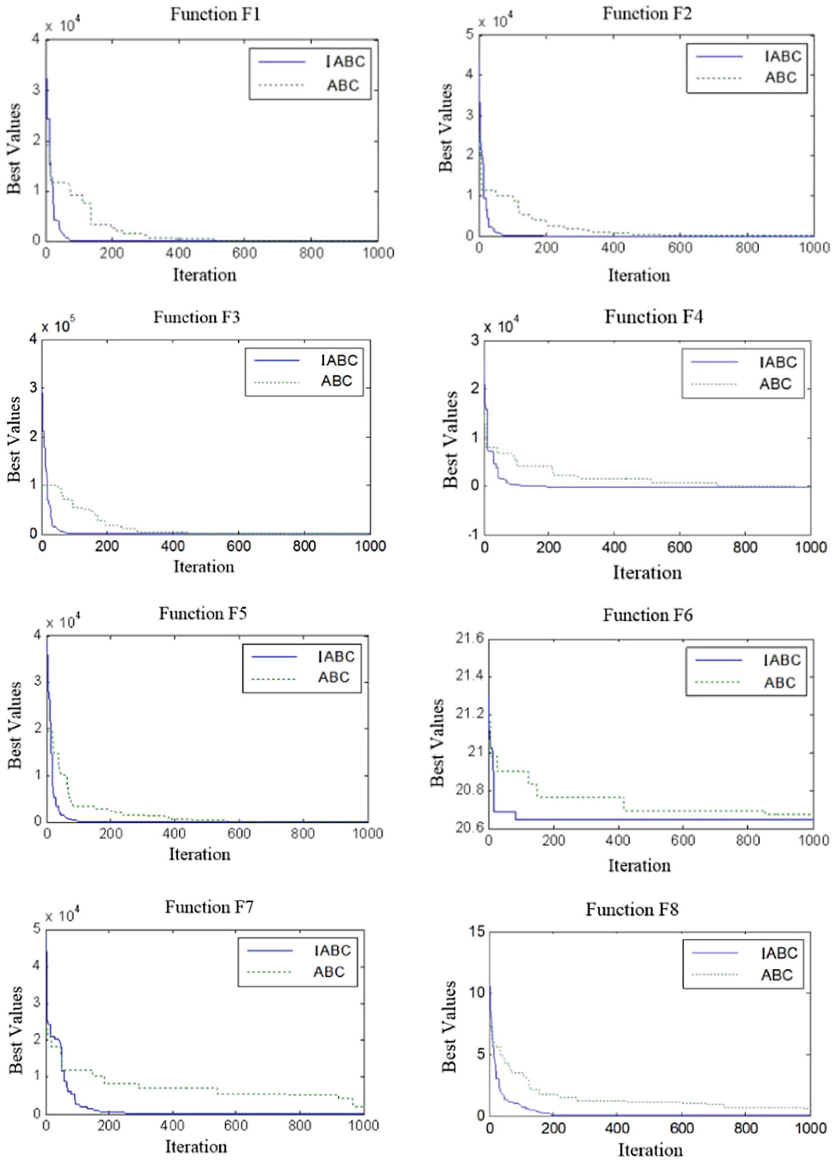


Fig. 1. Comparison of ABC and IABC on their convergence performance with the eight test functions

Table 4. Three levels for each of the four key factors

Level	Factors			D
	A	B	C	
1	30	$ch_{k+1,j} = \cos(\frac{\pi}{2} \times ch_{k,j})$	$v_{ij} = x_{ij} + rand(x_{k,j} - x_{ij})$	Roulette
2	50	$ch_{k+1,j} = \sin(\pi \times ch_{k,j})$	$v_{ij} = x_{ij} + \alpha(x_{best,j} - x_{ij})$	Reverse - roulette
3	70	$ch_{k+1,j} = \tan(\frac{\pi}{4} \times ch_{k,j})$	$\begin{cases} v_{ij} = x_{ij} + \alpha(x_{best,j} - x_{ij}) \\ ov_{ij} = x_{ij} + \alpha(x_{k,j} - x_{ij}) \end{cases}$	Parallel strategy

Table 5. Results of the Taguchi analysis on function f_5

No.		Factor				Results	
		A	B	C	D	Best(k)	Iteration(m)
1		1	1	1	1	77.46	1000
2		1	2	2	2	9.17	968
3		1	3	3	3	5.21	503
4		2	1	2	3	12.9	715
5		2	2	3	1	23.03	1000
6		2	3	1	2	115.12	1000
7		3	1	3	2	8.99	951
8		3	2	1	3	76.02	914
9		3	3	2	1	24.28	1000
Best	k_1	91.84	99.35	268.6	124.77	$R_k = \max\{\bar{k}_i\} - \min\{\bar{k}_i\}$	
	k_2	151.05	108.22	46.35	133.28		
	k_3	109.29	144.61	37.23	94.04		
	\bar{k}_1	30.61	33.12	89.53	41.59		
	\bar{k}_2	50.35	36.07	15.45	44.43		
	\bar{k}_3	36.43	48.2	12.41	31.35		
	R_k	19.74	15.08	77.12	13.08		
Iteration	m_1	2471	2666	2914	3000	$R_m = \max\{\bar{m}_i\} - \min\{\bar{m}_i\}$	
	m_2	2715	2882	2683	2919		
	m_3	2865	2503	2454	2132		
	\bar{m}_1	823.67	888.67	971.33	1000		
	\bar{m}_2	905	960.67	894.33	973		
	\bar{m}_3	955	834.33	818	710.67		
	R_m	131.33	126.34	153.33	289.33		

5 Conclusion

An improved ABC algorithm (IABC) has been developed in this paper with the purpose of increasing further the convergence speed and avoiding being trapped in local optimum. In the IABC, four key factors in the ABC have been enhanced respectively as follows. A chaotic mapping rule is established by introducing a chaos operator into the initial position generation rules so that the ergodicity of initial positions can be ensured; an isometric contraction parallel search rule is devised by parallelly introducing the current optimal position to guide the search and setting an isometric reduced parameter to control the neighborhood of the search so that the convergence speed and the local

search ability can be enhanced; a parallel selection strategy is developed by using the roulette and the reverse roulette simultaneously, which allows selecting poor positions to escape from local optimum. Meanwhile, the IABC develops a global updating mechanism based on gravitational potential field to guide the rejection and generation of positions so that the convergence of the algorithm is increased.

Experiments on eight test functions have demonstrated that the IABC algorithm is superior to the traditional ABC in terms of solution quality, convergence speed and iteration curve rate of change. To further analyze the level of factor combination, a Taguchi orthogonal method has been investigated and the computational results show that the proposed IABC algorithm has achieved the optimal combination of factor levels.

Acknowledgements. This research is supported by Natural Science Foundation of China (Grant No. 71571076 and 71171087) and by Major Program of National Social Science Foundation of China (Grant No. 13&ZD175).

References

1. Wu, H.S., Zhang, F.M., Wu, L.S.: New swarm intelligence algorithm-wolf pack algorithm. *J. Syst. Eng. Electron.* **35**(11), 2430–2438 (2013)
2. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948. IEEE (1995)
3. Dorigo, M., Maniezzo, V., Colomi, A.: Ant system: optimization by a colony of cooperating agents. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **26**(1), 29 (1996)
4. Karaboga, D.: An idea based on honey bee swarm for numerical optimization. Technical report-TR06, Erciyes University (2005)
5. Storn, R., Price, K.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **23**, 689–694 (2010)
6. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **39**, 459–471 (2017)
7. Zhu, G.P., Kwong, S.: Gbest-guided artificial bee colony algorithm for numerical function optimization. *Appl. Math. Comput.* **217**(7), 3166–3173 (2010)
8. Karaboga, D., Kaya, E.: An adaptive and hybrid artificial bee colony algorithm (aABC) for ANFIS training. *Appl. Soft Comput.* **49**, 423–436 (2016)
9. Alata, B.: Chaotic bee colony algorithms for global numerical optimization. *Expert Syst. Appl.* **37**, 5682–5687 (2010)
10. Zhang, X., Zhang, X., Yuen, S.Y., Ho, S.L., Fu, W.N.: An improved artificial bee colony algorithm for optimal design of electromagnetic devices. *IEEE Trans. Magn.* **49**(8), 4811–4816 (2013)
11. Li, J.Q., Pan, Q.K., Duan, P.Y.: An improved artificial bee colony algorithm for solving hybrid flexible flowshop with dynamic operation skipping. *IEEE Trans. Cybern.* **46**(6), 1311–1324 (2016)
12. Gao, W.F., Liu, S.Y., Huang, L.L.: A novel artificial bee colony algorithm based on modified search equation and orthogonal learning. *IEEE Trans. Cybern.* **43**(3), 1011 (2013)
13. Yang, J., Li, W.T., Shi, X.W., Xin, L., Yu, J.F.: A hybrid ABC-DE algorithm and its application for time-modulated arrays pattern synthesis. *IEEE Trans. Antennas Propag.* **61**(11), 5485–5495 (2013)

14. Li, Yu., Zhang, J., Zhou, D., Zhang, Q.: A segmented artificial bee colony algorithm based on synchronous learning factors. In: Nguyen, N.T., Trawiński, B., Fujita, H., Hong, T.-P. (eds.) ACIIDS 2016. LNCS (LNAI), vol. 9621, pp. 636–643. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49381-6_61
15. Kang, F., Li, J., Li, H., Ma, Z.: An improved artificial bee colony algorithm. In: International Workshop on Intelligent Systems and Applications, pp. 1–4. IEEE (2011)
16. Du, Z., Han, D., Liu, G., Jia, J., Du, Z., et al.: An improved artificial bee colony algorithm with elite-guided search equations. *Comput. Sci. Inf. Syst.* **14**, 27 (2017)
17. Zhang, J.Q.: CAE optimization analysis of injection process parameters for automobile CD bracket. *Eng. Plast. Appl.* **44**(07), 73–78 (2016)
18. Yang, W.H., Tarng, Y.S.: Design optimization of cutting parameters for turning operations based on the Taguchi method. *J. Mater. Process. Technol.* **84**(1–3), 122–129 (1998)
19. Bhatt, H.D., Vedula, R., Desu, S.B., Fralick, G.C.: Thin film TiC/TaC thermocouples. *Thin Solid Films* **342**(1–2), 214–220 (1999)
20. Ogryczak, W., Ruszczyński, A.: Dual stochastic dominance and related mean risk models. *SIAM J. Optim.* **13**(1), 60–78 (2002)
21. Rockafellar, R.T., Uryasev, S.: Conditional value-at-risk for general loss distribution. *J. Bank. Finan.* **26**(17), 1443–1471 (2002)
22. Gao, W., Liu, S.: A modified artificial bee colony algorithm. *Comput. Oper. Res.* **39**(3), 687–697 (2012)
23. Yu, H., Zeng, A.Z., Zhao, L.: Single or dual sourcing: decision-making in the presence of supply chain disruption risks. *Omega* **37**(4), 788–800 (2009)
24. Xiang, W.L., Ma, S.F.: Artificial bee colony based on reverse selection of roulette. *Appl. Res. Comput.* **30**(1), 86–89 (2013)
25. Liu, D.L., Chen, Y.Y.: A fragrance concentration based artificial bee algorithm and its application in robot path planning. *J. East China Univ. Sci. Technol. (Nat. Sci. Ed.)* **42**(3), 375–381 (2016)
26. Yu, H., Chung, C.Y., Wong, K.P.: Robust transmission network expansion planning method with Taguchi's orthogonal array testing. *IEEE Trans. Power Syst.* **26**(3), 1573–1580 (2011)
27. Mach, P., Zeman, P., Kotrčová, E., Barto, S.: Optimization of lead-free wave soldering process using Taguchi orthogonal arrays. In: Electronic System-Integration Technology Conference, pp. 1–4. IEEE (2010)



PLS-Based RBF Network Interpolation for Nonlinear FEM Analysis of Dropped Drum in Offshore Platform Operations

Hongwei Liu, Wenjun Zhang^(✉), Shuaichen Liu, and Yan Li

Navigation College, Dalian Maritime University, Dalian 116026, China
wenjunzhang@dmlu.edu.cn

Abstract. Offshore drilling platform plays an important role in the exploitation of offshore natural resources. Safety is the top priority in offshore platform operations. Among various risks, dropping objects is a major source of risk that threatens personal safety, platform structure and environment safety. In this paper, simulations are performed using finite element simulation software. As the custom objects in platform crane operations, the oil drum is used as the research object in the simulation of damage caused by dropping objects on the drilling platform deck structure at different contact angles. Through the analysis of the simulation test results, the relationship between the angle of the dropping object and the energy impact of the deck is obtained. As the result of the impact and the contact angle is a highly nonlinear mapping, the radial basis function neural network based on partial least squares is implemented for interpolation purposes. The approach of PLS-RBF (Partial Least Square-Radial Basis Function) method takes advantage of the RBF network and PLS regression method can obtain high generalization accuracy for nonlinear system mappings. The results are compared with other approaches to illustrate its effectiveness.

Keywords: Dropped object · Impact angle · ANSYS LS-DYNA
Partial least squares · Radial basis function network

1 Introduction

In the process of mining marine resources, the safety of offshore operations cannot be ignored [1]. According to DORIS (Dropped Objects Register of Incidents & Statistics), dropped objects are among the top ten causes of fatality and serious injury in oil and gas industry [2]. As the number of lifting increases, the occurrence of dropping objects is difficult to avoid. In offshore operations, the consequences of dropping objects vary according to the location of the impact, and the consequences are different [3]. When a collision occurs below sea level, it may cause damage to subsea equipment or submarine pipelines [4]. The collision between the ship and the platform may cause overall bending and partial depression of the platform structure, resulting in reduced bearing capacity, affecting the safety, durability and normal use of the platform structure [5].

The research methods of collision problems of offshore platforms mainly include empirical formula method/simplified the analytical method/energy method/the experimental method and numerical analysis method. Among them, the experimental method

is the most accurate [2]. So far, most of the test methods are only simulating ship collisions. For the collision damage research of offshore platforms, less experimental research can obtain accurate and reliable results, but it is a too extremely expensive destructive test to implement; Empirical and analytical methods usually have a strict scope of application; Numerical analysis method is used to discretize the collision structure into a finite element model. The explicit nonlinear finite element software can be used to simulate the collision process to obtain accurate numerical analysis results. It is an effective and fast tool to reduce the amount of experimental work [6].

This paper uses finite element method to study the influence of oil drum drop on drilling platform. The platform of ANSYS LS-DYNA is implemented to simulate the impact energy and the damage degree of the upper deck structure of the drilling platform at a different angle under the same height [7].

The simulation using the FEM (Finite Element Modeling) method is a tedious process with longtime consumption, so it is impossible to simulate all the possible situations. Therefore, it is reasonable to interpolate impact result within the result caused by the FEM method. As the result of the impact and the contact angle is a highly nonlinear mapping, the custom linear approach cannot get interpolation result with high accuracy, so the radial basis function neural network based on partial least squares is implemented for interpolation purposes and the results are compared with other approaches. The approach of PLS-RBF method takes advantage of RBF network and PLS regression method can obtain high generalization accuracy for nonlinear system mappings.

2 Introduction to the Finite Element Method

The numerical analysis method discretize the collision structure into the finite element model [8]. The explicit nonlinear finite element software can obtain accurate numerical analysis results for the collision process simulation. It cannot only calculate the structural damage deformation and collision force of the collision zone, but also combine it [9]. The analysis and calculation of the external mechanism can simulate the collision phenomenon and can partially replace the collision test of the model to realize the “virtual collision”, which provides an effective and fast tool for reducing the experimental workload [10].

The basic idea of the finite element method is to discretize the continuous structure into a finite number of units, and set a finite number of nodes in each unit, and regard the continuum as a collection of a group of units connected only at the nodes [11]. The distribution law of the fixed field function is used to establish the finite element equation for solving the unknown of the node by using the variation principle in mathematics so that the infinite degree of freedom problem in a continuous domain is transformed into the finite degree of freedom problem in the discrete domain. The FEM for solving the general procedure is shown in the Fig. 1 [12].

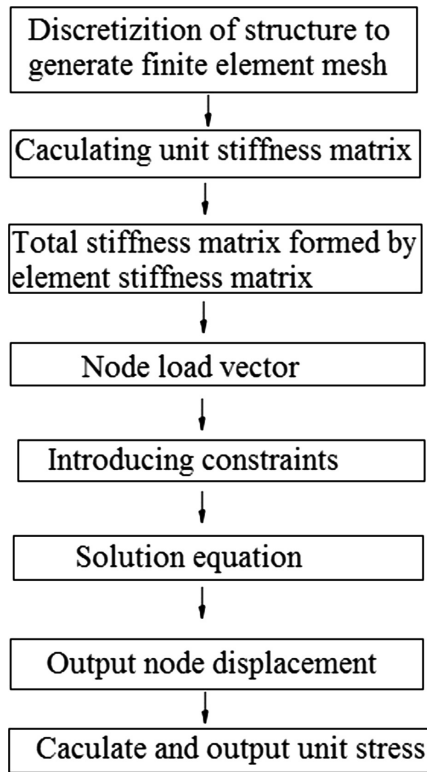


Fig. 1. Finite element method for solving the general program

3 RBF Network Based on Partial Least Squares Approach

BP network is a multilayer feedforward neural network composed of the input layer, hidden layer, and output layer. Figure 2 shows the topology of a typical three-layer BP network. A full interconnection between layers, there is no interconnection between the same layer, the hidden layer can have one or more layers. There are two kinds of signals circulating between layers: One is the working signal (indicated by the solid line), which is the signal that propagates forward after applying the input signal until the actual output is produced at the output, which is a function of the input and the weight. The other is the error signal (indicated by the dotted line). The difference between the actual output of the network and the expected output is the error. It starts with the output and propagates backward layer by layer.

BP network consists of forwarding calculation process and error backpropagation process. In the forward calculation process, the input amount is calculated layer by layer from the input layer and passed to the output layer. The state of each layer of neurons only affects the state of the next layer of neurons. If the output layer cannot obtain the desired output, the error is reversed to the propagation process, and the error signal returns along the original connection path and the weights and thresholds of each

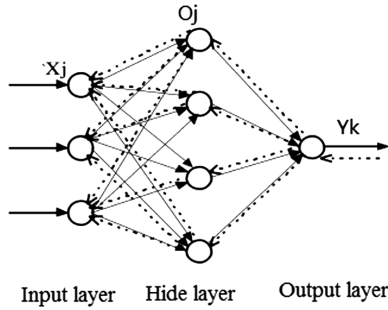


Fig. 2. Architecture of RBF network

layer of the network are adjusted successively until reaching the input layer, and then the calculation is repeated.

4 Numerical Analysis

4.1 Establishment of the Finite Element Model

Refer to real drilling platform parameters, built the finite element model of the deck and oil drums in the geometry of ANSYS [9]. The dropped object strikes a part of the position of the platform deck. Taking part of the platform deck as a research object, the deck dimension is $1000\text{ cm} \times 8000\text{ cm}$, the thickness is 3.0 cm . The width of the transverse section is 3.1 m . The elastic modulus of all components is $E = 206\text{ Gpa}$, Poisson's ratio is $\mu = 0.3$, The density is $\rho = 7850\text{ kg/m}^3$; The height of the oil drum is 93 cm , OD is 58 cm , ID is 57.8 cm , and fell from 500 cm , initial vertical velocity is 5 m/s , gravitational acceleration is 9.81 m/s^2 and study the endurance of deck. The ANSYS model at the angle of 0° is shown in Fig. 3. And the finite model of dropped oil drum at the angle of 90° is shown in Fig. 4. The working conditions settings of the simulation are shown in Table 1.

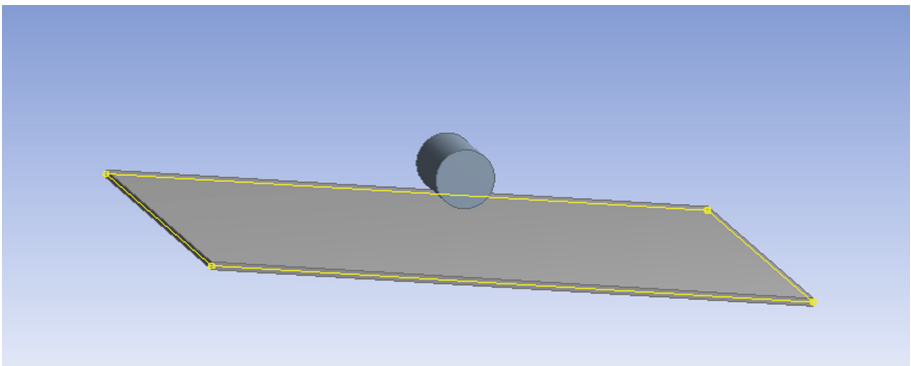


Fig. 3. The finite model of dropped oil drum at an angle of 0°

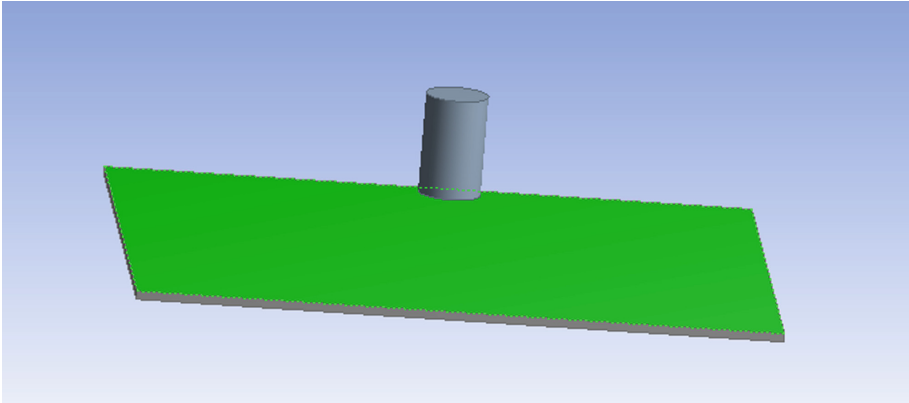


Fig. 4. The finite model of dropped oil drum at the angle of 90°

4.2 ANSYS LS-DYNA Simulation Results

In order to measure the consequences of dropping objects on the deck impact, the results of total deformation, equivalent stress and equivalent elastic strain were considered in the simulation. The simulation results obtained have a significant effect on the impact energy of the deck depending on the impact angle [15]. The stress/strain and degree of depression of the deck material vary with the angle of impact. The dynamic analysis of the finite element software ANSYS LS-DYNA shows the stress distribution and deformation of the deck.

The stress distribution cloud at an impact angle of 45° and 90° through ANSYS LS-DYNA Post-processing results are shown in Figs. 5 and 6.

4.3 Consequence Analysis of the Dropped Drums

The results produced by simulating the dropping objects are shown in Table 2. It can be seen from Table 2 that when the oil drum drops at an angle of 0° , the equivalent deformation to the deck is the largest, and the equivalent stress is also relatively large. At this time, the deck structure is seriously damaged. When the drum impacts the deck at an angle of 45° , the equivalent strain and equivalent stress produced by the deck is minimal.

The equivalent deformation of the oil drum to the deck at 0° impact on the deck is the largest. The momentum and energy curves of the drum when it hits the deck at 0° are shown in the Figs. 7 and 8.

It can be seen from the above experimental results that the impact damage of the collision is mainly concentrated in the contact area of the collision, and the energy of the dropping object is mostly absorbed by the deformation of the deck. In the case of other factors being equal, the impact of the oil drum at different angles has a significant impact on the equivalent deformation of the deck. The energy impact is generated when

Table 1. Working conditions settings of the simulation

Condition	Shape	Dimension (mm)	Height of drop (m)	Impact velocity (m/s)	Mass (Kg)	Density (Kg/m ³)	Elastic modulus	Angle (°)
Drum-0	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	0
Drum-05	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	5
Drum-10	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	10
Drum-15	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	15
Drum-20	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	20
Drum-25	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	25
Drum-30	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	30
Drum-35	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	35
Drum-40	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	40
Drum-45	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	45
Drum-50	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	50
Drum-55	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	55
Drum-60	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	60
Drum-65	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	65
Drum-70	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	70
Drum-75	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	75
Drum-80	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	80
Drum-85	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	85
Drum-90	Drum	OD = 580 ID = 578	5	5	180	7850	2.10E+11	90

the angle of the drop is 45° is the smallest, with 45° as the boundary. When the angle is gradually increased to 90° or decreased to 0° , the impact energy generated by the dropping object on the deck is increasing.

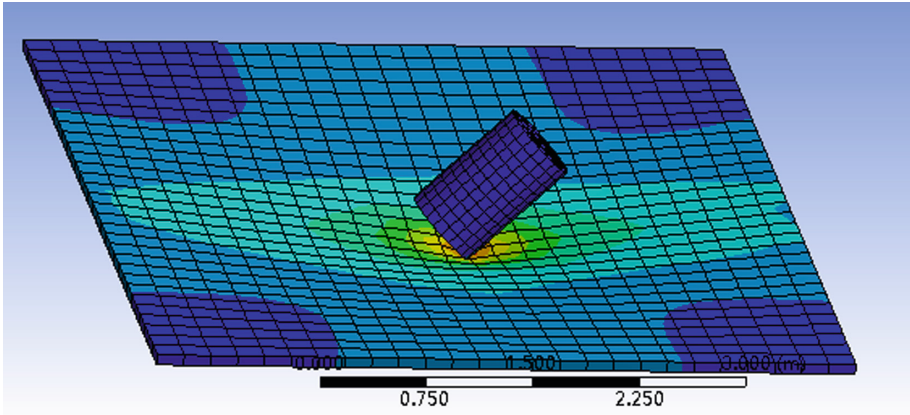


Fig. 5. The stress distribution at an impact angle of 45°

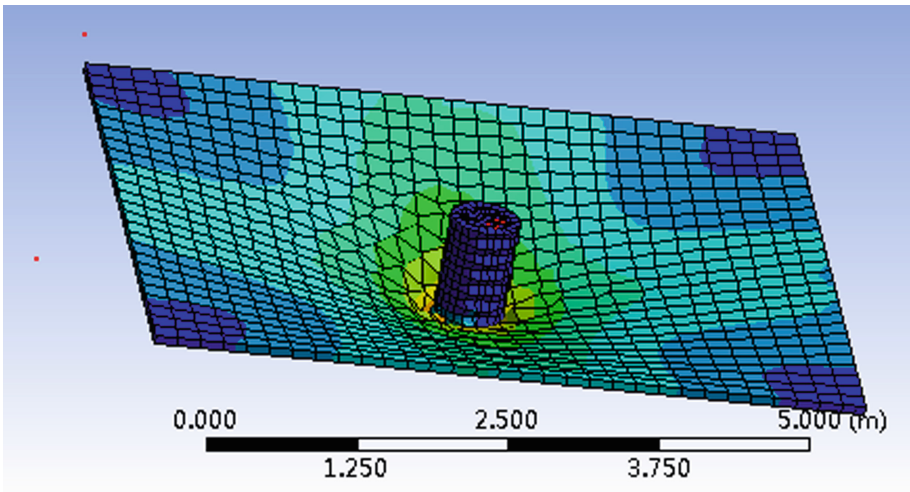


Fig. 6. The stress distribution at an impact angle of 90°

4.4 Nonlinear Mapping of the FEM Results Using PLS-RBF Network

In this study, the relation between the impact and the contact angle is a highly nonlinear process, conventional linear approach cannot get precise interpolation result, so the radial basis function neural network based on partial least squares is implemented for interpolation purposes and the results are compared with custom polynomial interpolation method.

The simulation results in the previous section demonstrate that the result of the collision force is related to the impact angle. The polynomial fitting method was used to fit the simulation and the results are shown in Fig. 9. It is noticed from Fig. 9 that the

Table 2. Equivalent deformation and equivalent stress of deck under different angles

Angle (°)	Equivalent elastic strain (m)	Equivalent (Pa)
0	0.14	2.22E+10
5	0.124	2.25E+10
10	0.104	1.92E+10
15	0.0533	8.56E+09
20	0.0788	1.63E+10
25	0.0314	5.27E+09
30	0.0287	4.85E+09
35	0.0233	4.16E+09
40	0.076	1.36E+10
45	0.023	4.06E+09
50	0.044	7.61E+09
55	0.025	4.55E+09
60	0.036	6.29E+09
65	0.047	7.92E+09
70	0.062	1.04E+10
75	0.1102	2.09E+10
80	0.10941	1.76E+10
85	0.12463	2.20E+10
90	0.10494	1.82E+10

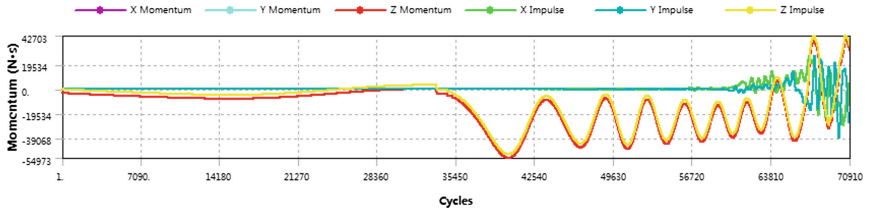


Fig. 7. The momentum curve at an impact angle of 0°

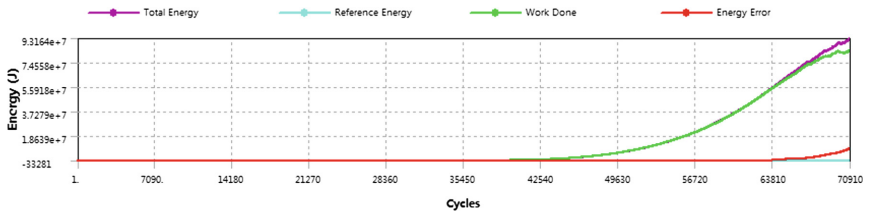


Fig. 8. The energy curve at an impact angle of 0°

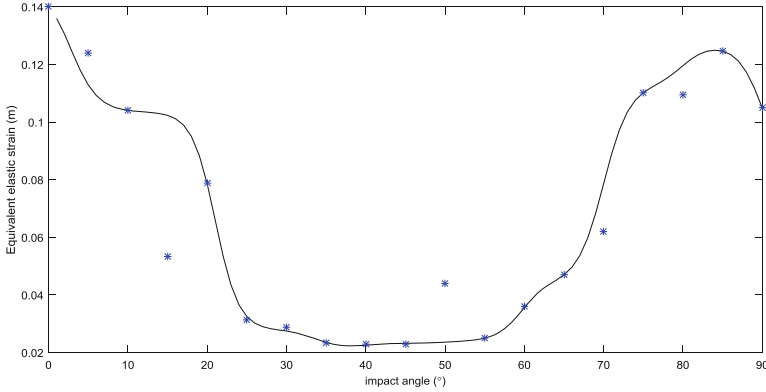


Fig. 9. Nonlinear mapping result achieved by PLS-RBF method

maximum impact force has a nonlinear relationship with the impact angle. The fitting curve can well reflect the data distribution. The fitting degree is 95.43%.

For comparison purpose, the custom polynomial method is also adopted to mapping the same mapping between the contact angle and the impact result. It is also noted that there are some field data in the simulation result which present challenges for the mapping. Neural networks possess high nonlinear mapping capability [16], are often used for interpolation. In this study, the PLS-RBF network can track the holistic change trends of the curve and neglects the field data, which shows the good representing ability of PLS-RBF network. The fitting results by using polynomial fitting method are shown in Fig. 10.

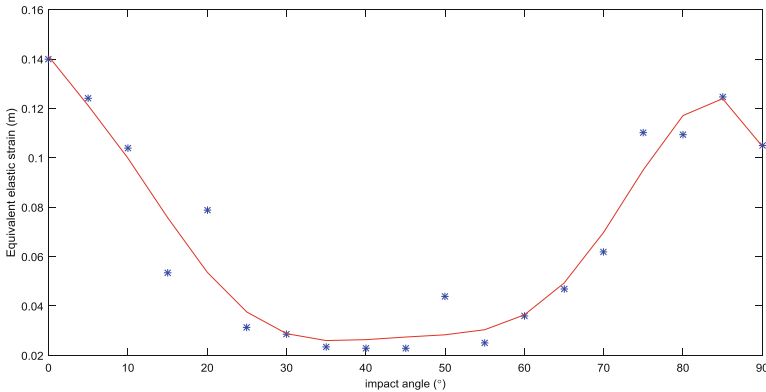


Fig. 10. Mapping result achieved by polynomial method

It is shown from the figures that the approach of PLS-RBF method takes advantage of RBF network and PLS regression method, can obtain higher generalization accuracy for FEM nonlinear system mappings than custom polynomial method.

After the training, several other simulations are conducted and the results are used to verify the fitting result of the fitting method (Table 3).

Table 3. Comparison of fitting results

Impact angle	Simulated value	Generalization by PLS-RBF	Generalization by polynomial
17	0.0954	0.0976	0.0682
27	0.0305	0.0294	0.0341
77	0.1123	0.1109	0.1096

According to the verification simulation results, the fitting result achieved by PLS-RBF network shows higher accuracy than that of the polynomial method, which reflect the higher nonlinear representation ability for the complex relationship between the maximum equivalent stress and the impact angle.

5 Conclusion

In this paper, the finite element simulation software ANSYS LS-DYNA is used to simulate the damage caused by the oil drum object dropping at different angles to the deck of the drilling platform, and a simple simulation result is obtained. It is found that the drop angle is also one of the important factors determining the damage result of offshore platform deck. The impact results show that the equivalent stress and the equivalent elastic deformation change nonlinearly when the oil drum is at different angles, the other conditions are the same. Using the non-linear mapping of the FEM results of the PLS-RBF network, higher generalization accuracy and wider adaptability are obtained.

This paper assesses the damage caused by falling objects in offshore activities and aims to reduce the risk or damage to the safety of personnel or equipment during the offshore operations. In order to obtain more reasonable and accurate simulation results, it needs detailed modeling and necessary external influence factors. The nonlinear mapping using the neural network is also to be improved to satisfy the high-dimension mapping caused thereby.

Acknowledgement. This work is supported by grant from the 7th Generation Ultra-Deep-water Drilling Rig Innovation Project, the Liaoning Natural Science Foundation of China, and the Natural Science Foundation of China under Grant 51609132 [13].

References

1. Kenny, J.P.: Protection of offshore Installations Against Impact offshore Technology Report. OTI-88535 (1991)
2. Arabzadeh, H., Zeinoddini, M.: Dynamic response of pressurized submarine pipelines subjected to transverse impact loads. *Procedia Eng.* **14**(2259), 648–655 (2011)

3. Fujii, Y., et al.: Some factors affecting the frequency of accidents in marine traffic. *J. Navig.* **27**(2), 239–247 (1974)
4. Yan, S., Tian, Y.: Analysis of pipeline damage to impact load by dropped objects. *Trans. Tianjin Univ.* **12**, 138–141 (2006)
5. Abosbaia, A.S., Mahdi, E., Hamouda, A.M.S., et al.: Energy absorption capability of laterally loaded segmented composite tubes. *Compos. Struct.* **70**(3), 356–373 (2005)
6. Thapa, P., Khan, F.: Dropped object effect in offshore subsea structures and pipeline approach. Technical report (2016)
7. Kawsar, M.R.U., Youssef, S.A., Faisal, M., et al.: Assessment of dropped object risk on corroded subsea pipeline. *Ocean Eng.* **106**, 329–340 (2015)
8. Awotahegn, M.B.: Experimental investigation of accidental drops of drill pipes and containers during offshore operations. University of Stavanger, Norway (2015)
9. Sun, L.P., Ma, G., Nie, C.Y., et al.: The simulation of dropped objects on the offshore structure. *Adv. Mater. Res.* **339**, 553–556 (2011)
10. Amdahl, J., Eberg, E.: Ship collision with offshore structures. In: Proceedings of 2nd European Conference on Structural Dynamics, Trondheim, Norway, pp. 495–504 (1993)
11. Pedersen, P.T., Jensen, J.J.: Ship impact analysis for bottom supported offshore structures. *Adv. Mar. Struct.* 276–295 (1991)
12. Gu, Y., Wang, Z.L.: An inertia equivalent model for numerical simulation of ship-ship collisions. In: 2nd International Conference on Collision and Grounding of Ships ICCGS, Copenhagen, Denmark, pp. 155–160 (2001)
13. Yin, J.C., Wang, N., Perakis, A.: A real-time sequential ship roll prediction scheme based on adaptive sliding data window. *IEEE Trans. Syst. Man Cybern.: Syst.* **99**, 1–11 (2017)



Logic Circuit Design of Sixteen-Input Encoder by DNA Strand Displacement

Yanfeng Wang^{1,2}, Aolong Lv^{1,2}, Chun Huang^{1,2},
and Junwei Sun^{1,2} (✉)

¹ Henan Key Lab of Information-Based Electrical Appliances,
Zhengzhou University of Light Industry, Zhengzhou 450002, China
junweisun@yeah.net

² School of Electrical Information Engineering,
Zhengzhou University of Light Industry, Zhengzhou 450002, China

Abstract. As a very powerful tool, DNA strand displacement technology is widely used in molecular calculations, which has received more and more attention from people. It has achieved rapid development in recent years. In this paper, a sixteen-input encoder based on DNA strand displacement is designed, including dual-rail circuits and seesaw circuits based on DNA strand displacement. Visual DSD is a visualization software with compiling and emulation capabilities. All reactions of encoding and simulation are performed in the visualization software Visual DSD. By changing the input signal, the accurate output can be obtained, the function of sixteen-input encoder in digital circuit is realized. With Visual DSD software, more complex logic operations are implemented based on DNA strand displacement. This kind of research based on dual-rail circuits by DNA strand displacement have great prospects for the development and practical application of biological information processing and molecular computing.

Keywords: DNA strand displacement · Sixteen-input encoder
Visual DSD · Molecular computing

1 Introduction

In the development of traditional electronic technology, integrated circuits are increasingly unable to meet people's needs [1]. Due to its powerful parallel computing power and massive storage capacity, DNA molecular computing have attracted the attention of researchers [2, 3]. In recent years, molecular computing has achieved rapid development. With the rise of molecular computing, the most new method of DNA strand replacement technology has been greatly developed [4]. It has realized the construction of biological logic gates to a large number of biological logic circuits, targeted therapy for diseases, biological nano-computers, and so on [5, 6]. Similarly, it is fast, accurate, and convenient that DNA strand replacement can solve complex mathematical problems, such as the Hamilton path [7]. Therefore, exploration of DNA strand replacement technology has become an important way to study biological computing.

The encoder is a device that compiles or converts a signal or data into a form that can be used for communication, transmission, and storage. According to the working principle, the encoder can be divided into two types: incremental and absolute [8]. The incremental encoder converts the displacement into a periodic electrical signal, which is then converted into a counting pulse. Each position of the absolute encoder corresponds to a certain digital code, so its indication is only measured. The encoder has the advantages of small size, precision, high resolution and long life, and is widely used at home and abroad, especially the sixteen-input encoder [9].

Compared with the existing work, there are some advantages, which can make our study more interesting and attractive than the previous logic circuit. First, the method based on DNA strand displacement is first applied to the logic circuit of the sixteen-input encoder. Secondly, compared with the four-input encoder and the eight-input encoder, the sixteen-input encoder implements more functions and is more widely used. The function of this circuit is more powerful in studying complex circuits. Finally, the research of sixteen-input encoder provides convenience and method for the research of more input encoders in the future.

In this paper, the rest of the content is as follows: the Sect. 2 is the reaction mechanism of DNA strand replacement. The Sect. 3 is the logic design of the sixteen-input encoder, including the dual-rail circuit and the seesaw circuit. The Sect. 4 is the Visual DSD simulation of the sixteen-input encoder. At last, the Sect. 5 is the conclusion of sixteen-input encoders based on dual-rail circuit and DNA strand displacement.

2 DNA Strand Displacement

The DNA strand displacement reaction relies on the weak traction between molecules to obtain power [10, 11]. The development of DNA strand displacement technology is based on DNA self-assembly technology. In the case where both single-stranded and double-stranded strands exist, the DNA strand displacement reaction can be completed, and the theoretical basis is the principle of base complementary pairing (A is paired with T, G is paired with C) [12]. The number of bases affects the traction between the molecules, which in turn affects the rate of reaction. The reaction mechanism of DNA strand replacement is shown in Fig. 1. First, the base “g” on the DNA single strand $\langle c g d \rangle$ is base-paired with the base sequence “g*” on the DNA double strand $\langle g^* d^* g^* \rangle$, and then the base “d” on the DNA single strand $\langle a t b \rangle$ is base-paired with the base sequence “d*” on the DNA double strand $\langle g^* d^* g^* \rangle$, and replace the base “d” which has been paired with “d*” on the DNA double strand $\langle g^* g^* g^* \rangle$ [13]. Finally, the single strand $\langle d g n \rangle$ detaches from the complex and becomes a separate part. The whole reaction can be seen as a process in which the single strand $\langle c g d \rangle$ replaces the single strand $\langle d g n \rangle$, that is, the single strand $\langle c g d \rangle$ is an input strand, and the single strand $\langle d g n \rangle$ is an output strand [14, 15].

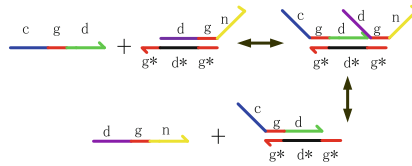


Fig. 1. DNA strand displacement reaction mechanism.

3 Logic Circuit of Sixteen-Input Encoder

3.1 Digital Logic Circuit

Logical operations have two different input signals, which are “0” and “1” respectively. When there is a signal input, it is “1”, otherwise, it’s “0” [16]. In digital logic circuit, logic operation includes three basic operation modes, which are respectively AND gate, OR gate, and NON-gate [17]. Sixteen-input encoder has four outputs, they are “Y₀”, “Y₁”, “Y₂”, “Y₃”. When one of the sixteen input signals is “1”, the remaining fifteen are “0”, it can produce corresponding output result. There are total 16 output results. Since the “Y₀”, “Y₁”, “Y₂”, “Y₃” logic circuit are the same, we only show the logic circuit of “Y₀”. Of course, here we show the module circuit of “Y₀”, modules 2 and 3, 4, 5 are the same modules (Fig. 2).

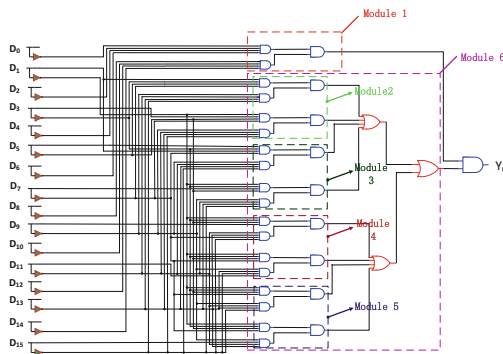


Fig. 2. Digital logic circuit of sixteen-input encoder of “Y₀”.

3.2 Dual-Rail Logic Circuit

The input and output signals of the DNA logic gate are single-stranded DNA molecules, and their logic values are different according to the level of their own concentration [18, 19]. A DNA non-gate needs to be able to distinguish between an uncalculated high input signal and a low input signal. When the upper input signal reaches a certain high concentration and requires sufficient time instead of the gate calculation speed too fast, the non-gate output signal may damage to the underlying calculations. In view of the difficulties in the construction of DNA non-gates, in order

to construct a semi-subtractor biological circuit, it is necessary to apply to the dual-rail logic [20]. Namely a circuit of AND, OR, NOT, NAND and NOR gates are transformed into an identical dual-rail circuit that only included AND and OR gates [21]. The original input and output logical values are represented by two different DNA strands. That is, for each input “X”, two inputs “ X_0 ” and “ X_1 ” can be used instead to represent logic “0” and logical “1” states of “X” respectively. If “ X_0 ” is on, then the logical value of “X” is “0”. If “ X_1 ” is on, then the logical value of “X” is “1”. Normally, one of “ X_0 ” and “ X_1 ” is “0” and the other is “1”. If “ X_0 ” and “ X_1 ” are both OFF, there is no input signal. If both “ X_0 ” and “ X_1 ” are ON, this circuit does not exist [21, 22].

In order to more clearly show the dual-rail logic diagram of output Y_0 , we only show the dual-rail logic circuit of module 1 and module 2 of “ Y_0 ” here. We assume the output of module 1 as “W”. The dual-rail circuit come into being output signal “ W_0 ” and “ W_1 ” represent OFF and ON, respectively. Similarly, we define the output of module 2 as “ A_0 ” and “ B_0 ”. If “ A_0^0 ” is ON, then the logical value of “ A_0 ” is “0”. If “ A_0^1 ” is ON, then the logical value of “ A_0 ” is “1”. If “ B_0^0 ” is ON, then the logical value of “ B_0 ” is “0”. If “ B_0^1 ” is ON, then the logical value of “ B_0 ” is “1” (Fig. 3).

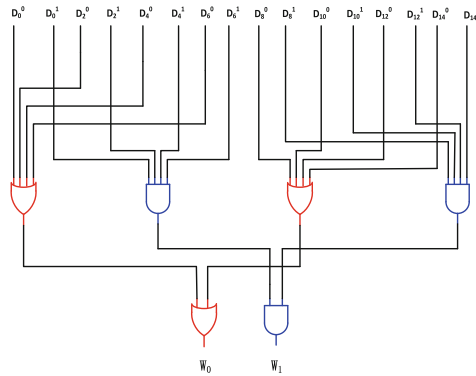


Fig. 3. Dual-rail circuit of module 1 of “ Y_0 ” of sixteen-input encoder.

3.3 Seesaw Cascade Circuit

During the DNA strand displacement reaction, when the input chain concentration is greater than the threshold chain concentration, the input chain first participates in the reaction with the threshold chain [19]. When the threshold chain reaction is completed, the remaining input chains participate in the reaction with the gate complex, thereby releasing the output chain $\langle S_6 \text{ T } S_5 \rangle$, the released output chain can be used as the output of the next gate [23, 24]. When the input chain concentration is less than the threshold chain concentration, the input chain does not react with the gate complex, the output signal chain is not released. When the output signal is released, the DNA single strand $\langle T^*S_5^*T^* \rangle$ will be generated. If it is not eliminated, it will react in the reverse direction. Therefore, the fuel chain $\langle S_7 \text{ T } S_5 \rangle$ is added to react with it, causing the

reaction to be positive. Direction is carried out to achieve amplification of the output signal [25] (Figs. 4 and 5).

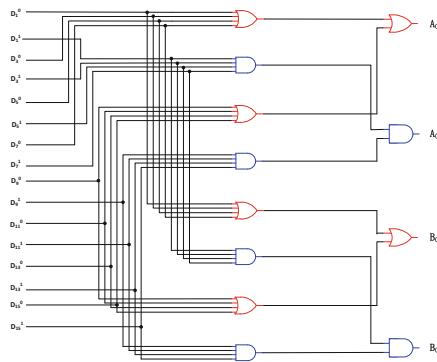


Fig. 4. Dual-rail circuit of module 2 of “Y0” of sixteen-input encoder.

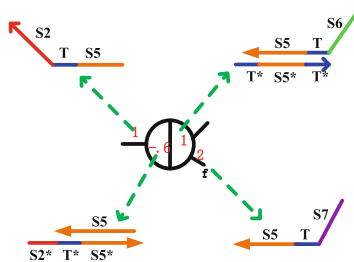


Fig. 5. DNA gate level.

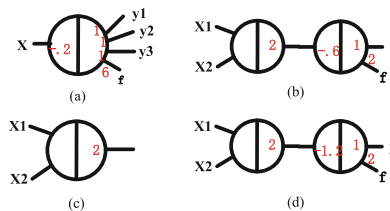


Fig. 6. (a) Amplifying gate: the amplifying gate of the seesaw cascade circuit. (b) “OR” gate: the seesaw cascade circuit of the “OR” gate with two inputs and one output (c) Integrated gate: the integrated gate of the seesaw cascade circuit. (d) “AND” gate: the seesaw cascade circuit of the “AND” gate with two inputs and one output.

Figure 6(a) is an input and three output amplifying gates (single input and multiple output can be realized). Figure 6(b) is the seesaw cascade circuit of the “OR” gate. Its gate threshold (indicated by the) is “-0.6” [26]. When the logical value of the chain

“X1” or the chain “X2” is “1”, the logical value of the generated chain “y” is “1”; if the logical values of the added chains x1 and x2 are both “0”, the logical value of the generated chain “y” is “0”. Figure 6(c) is an integrated gate with two inputs and one output (multiple input and single output can be realized), Fig. 6(d) is the seesaw cascade circuit of the “AND” gate, whose second-level gate has a threshold of “-1.2” [27]. When the logical values of the join chains “X1” and “X2” are both “1”, the logical value of “y” is only “1”; if the logical value of the added chains x1 and x2 has a value of “0”, the logical value of the generated chain “y” is “0” [28, 29] (Figs. 7 and 8).

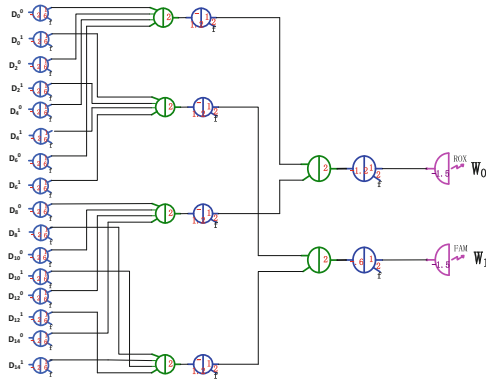


Fig. 7. Seesaw cascade circuit of the module 1 of “Y0” of sixteen-input encoder

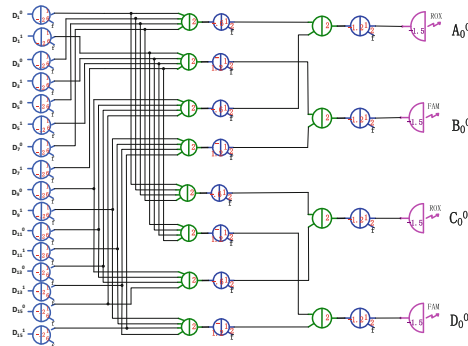


Fig. 8. Seesaw cascade circuit of the module 2 of “Y0” of sixteen-input encoder

4 Simulation with Visual DSD

To help solve these challenges and verify the rationality of the DNA strand replacement design circuit, the researchers designed DSD software, which can be used for programming languages, and can also simulate computing devices composed of DNA [30, 31]. DSD consists of two parts, one is the coding part, and the other is the figure

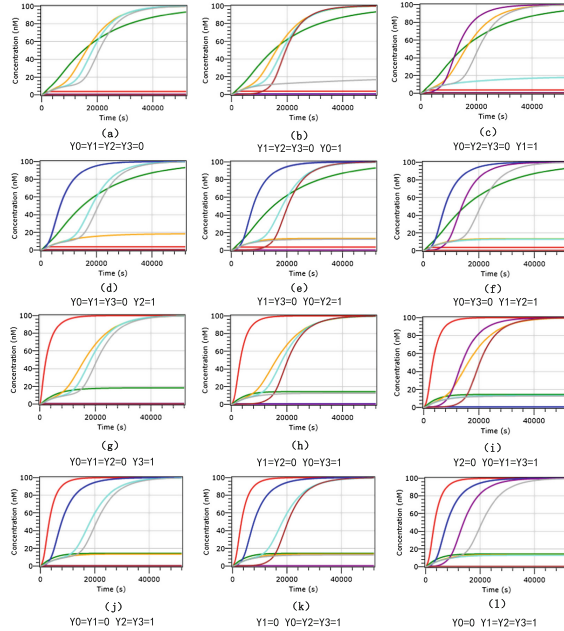


Fig. 9. Simulation results of sixteen-input encoder (Color figure online)

display part. In the coding section, some parameters need to be set, including reactant time, reactant concentration, separation time, and so on. In the figure display section, there are usually a plurality of curves, which are trajectories that change with time, the abscissa is time, and the ordinate is density. In addition, DSD can also build the state vector of the system [30] (Fig. 9).

Sixteen-input encoder has four outputs, they are “Y0”, “Y1”, “Y2” “Y3”. There are special 9 kinds simulation results of all 16 kinds simulation results. In 9 kinds simulation figures, gray line represents the logical value of “Y0” is “0”, magenta line represents the logical value of “Y0” is “1”, bright line green represents the logical value of “Y1” is “0”, purple represents the logical value of “Y1” is “1”, yellow line represents the logic of “Y2”. The value is “0”, blue line represents the logical value of “Y2” is “1”, green line represents the logical value of “Y3” is “0”, and red line represents the logical value of “Y3” is “1”. The simulation time is 52000 s. We put the threshold value of OFF is “0.1x”, and the threshold value of ON is “0.9x”. The ordinate indicates the change of the concentration of the reactants.

5 Conclusion

In this paper, the reaction mechanism of DNA strand displacement is first introduced. Secondly, the logic circuit design of the sixteen-input encoder is realized by the dual-rail circuit and the seesaw circuit. Finally, through the DSD simulation, the logical operation result of the sixteen-input encoder is correctly displayed.

The sixteen-input encoders can be extended to more input encoders, and the sixteen-input encoder play a more prominent role in practical applications. Research on sixteen-input encoders based on DNA strand displacement have a positive effect on research and applications of molecular computation and information processing. Due to limited experimental conditions, in the case of multi-input encoders, more in-depth research is needed for logic circuits based on DNA strand displacement.

Acknowledgment. The work is supported by the State Key Program of National Natural Science of China (Grant No. 61632002), the National Key R&D Program of China for International S&T Cooperation Projects (No. 2017YFE0103900), the National Natural Science of China (Grant Nos. 61603348, 61775198, 61603347, 61572446, 61472372), Science and Technology Innovation Talents Henan Province (Grant No. 174200510012), Research Program of Henan Province (Grant Nos. 172102210066, 17A120005, 182102210160), Youth Talent Lifting Project of Henan Province and the Science Foundation of for Doctorate Research of Zhengzhou University of Light Industry (Grant No. 2014BSJJ044).

References

1. Bray, D.: Protein molecules as computational elements in living cells. *Nature* **376**, 307–312 (1995)
2. Wang, Z., Wu, Y., Tian, G., Wang, Y., Cui, G.: The application research on multi-digit logic operation based on DNA strand displacement. *J. Comput. Theor. Nanosci.* **12**(7), 1252–1257 (2015)
3. Ezzine, Z.: DNA computing: applications and challenges. *Nanotechnology* **17**(2), R27 (2005)
4. Gupta, P.K.: Single-molecule DNA sequencing technologies for future genomics research. *Trends Biotechnol.* **26**(11), 602–611 (2008)
5. Bui, H., Garg, S., Miao, V., Song, T., Mokhtar, R., Reif, J.: Design and analysis of linear cascade DNA hybridization chain reactions using DNA hairpins. *New J. Phys.* **19**(1), 015006 (2017)
6. Qian, L., Winfree, E.: A simple DNA gate motif for synthesizing large-scale circuits. *J. Roy. Soc. Interface* rsif20100729 (2011)
7. Li, W., Yang, Y., Yan, H., Liu, Y.: Three-input majority logic gate and multiple input logic circuit based on DNA strand displacement. *Nano Lett.* **13**(6), 2980–2988 (2013)
8. Shi, Z., Chen, P., Chen, D.: High resolution digital velocity detection and dynamic position detection methods. *J. Tsinghua Univ. (Sci. & Tech.)* **44**(8), 1021–1024 (2004)
9. Song, G., Qin, Y.X., Zhang, K.: Approach and realization to improve the measuring accuracy with low resolution encoder. *J. Shanghai Jiao Tong Univ.* **36**(8), 1169–1172 (2002)
10. Chen, Y.J., et al.: Programmable chemical controllers made from DNA. *Nat. Nanotechnol.* **8** (10), 755–762 (2013)
11. Wang, Y., Sun, J., Zhang, X., Cui, G.: Full adder and full subtractor operations by DNA self-assembly. *Adv. Sci. Lett.* **4**(2), 383–390 (2011)
12. Topal, M.D., Fresco, J.R.: Complementary base pairing and the origin of substitution mutations. *Nature* **263**(5575), 285 (1976)
13. Song, T., Garg, S., Mokhtar, R., Bui, H., Reif, J.: Analog computation by DNA strand displacement circuits. *ACS Synth. Biol.* **5**(8), 898–912 (2016)
14. Genot, A.J., Fujii, T., Rondelez, Y.: Computing with competition in biochemical networks. *Phys. Rev. Lett.* **109**(20), 208102 (2016)

15. Yang, X., Tang, Y., Traynor, S.M., Li, F.: Regulation of DNA strand displacement using an allosteric DNA toehold. *J. Am. Chem. Soc.* **138**(42), 14076–14082 (2016)
16. Qian, L., Winfree, E.: Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**, 1196–1201 (2016)
17. Zhang, X., Zhang, W., Zhao, T., Wang, Y., Cui, G.: Design of logic circuits based on combinatorial displacement of DNA strands. *J. Comput. Theor. Nanosci.* **12**(7), 1161–1164 (2015)
18. Lakin, M.R., Stefanovic, D.: Supervised learning in adaptive DNA strand displacement networks. *ACS Synth. Biol.* **5**(8), 885–897 (2016)
19. Wang, Y., Cui, G., Zhang, X., et al.: Logical NAND and nor operations using algorithmic self-assembly of DNA molecules. *Phys. Procedia* **33**, 954–961 (2012)
20. Qian, L., Winfree, E., Bruck, J.: Neural network computation with DNA strand displacement cascades. *Nature* **475**(7356), 368–372 (2011)
21. Zhang, C., Yang, J., Xu, J.: Circular DNA logic gates with strand displacement. *Langmuir* **26**(3), 1416–1419 (2009)
22. Wang, Z., Cai, Z., Sun, Z., Ai, J., Wang, Y., Cui, G.: Research of molecule logic circuit based on DNA strand displacement reaction. *J. Comput. Theor. Nanosci.* **13**(10), 7684–7691 (2016)
23. Qian, L., Winfree, E.: Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**(6034), 1196–1201 (2011)
24. Sawlekar, R., Montefusco, F., Kulkarni, V.V., Bates, D.G.: Implementing nonlinear feedback controllers using DNA strand displacement reactions. *IEEE Trans. Nano Biosci.* **15** (5), 443–454 (2016)
25. Goodman, R.P., Heilemann, M., Doose, S., Erben, C.M., Kapanidis, A.N., Turberfield, A.J.: Reconfigurable, braced, three-dimensional DNA nanostructures. *Nature Nanotechnol.* **3**(2), 93–96 (2008)
26. Yang, J., Dong, C., Dong, Y., Liu, S., Pan, L., Zhang, C.: Logic nanoparticle beacon triggered by the binding-induced effect of multiple inputs. *ACS Appl. Mater. Interfaces* **6** (16), 14486–14492 (2014)
27. Sun, J., Wu, Y., Cui, G., Wang, Y.: Finite-time real combination synchronization of three complex-variable chaotic systems with unknown parameters via sliding mode control. *Nonlinear Dyn.* **88**(3), 1677–1690 (2017)
28. Wang, Y., Tian, G., Hou, H., Ye, M., Cui, G.: Simple logic computation based on the DNA strand displacement. *J. Comput. Theor. Nanosci.* **11**(9), 1975–1982 (2014)
29. Phillips, A., Cardelli, L.: A programming language for composable DNA circuits. *J. Roy. Soc. Interface* **6**(Suppl. 4), S419–S436 (2014)
30. Lakin, M.R., Youssef, S., Polo, F., Emmott, S., Phillips, A.: Visual DSD: a design and analysis tool for DNA strand displacement systems. *Bioinformatics* **27**(22), 3211–3213 (2011)
31. Visual DSD. <https://www.microsoft.com/en-us/research/project/programming-dna-circuits>. Accessed 12 May 2018



PLS-RBF Neural Network for Nonlinear FEM Analysis of Dropped Container in Offshore Platform Operations

Zehua Li, Wenjun Zhang^(✉), and Haibo Xie

Navigation College, Dalian Maritime University,
Dalian 116026, Liaoning, China
wenjunzhang@dlmu.edu.cn

Abstract. Accidents caused by accidental load not only lead to casualties and major economic losses, but also cause serious pollution and damage to the surrounding environment and marine ecology. The study of its safety of offshore platform in complex and extreme environments is a research spotlight in ocean engineering area. In addition to the normal work load and environmental load, the ocean platform is also threatened by accidental load, such as the collision of the ship and the impact of the upper part of the platform. It is necessary to evaluate the risk that the platform encountered. However, there is little statistics on the risk of offshore platforms in the industry of marine engineering, which has caused difficulties in quantitative calculation of risk. In addition, the mechanical mechanism of the marine structure injury caused by the collision itself is complex and it is not feasible to conduct large number of experimental simulations, so the numerical simulation using nonlinear finite element method (FEM) is implemented in this study. To achieve accurate impact result under arbitrary situations, nonlinear interpolation is made by means of PLS-RBF network. Simulation results demonstrate the feasibility and effectiveness of the nonlinear PLS-RBF mapping for nonlinear finite analysis.

Keywords: Drop object · Container · LS-DYNA, PLS-RBF network

1 Introduction

Along with the development of oil and gas resources industry, oil and gas exploration and development is carried out from the land to the ocean gradually, and offshore drilling platforms and drilling ships as well as other offshore facilities are particularly important in the development of marine facilities. Among them, the risk from offshore facilities has been widely concerned. The using of risk assessment for identifying safety risks and guiding safety design attracted many concerns in areas of ocean engineering and energy. In offshore operations, damage by dropping objects from crane activity and other operations is a significant hazard for the offshore platform structure integrity.

A statistic from the Drop Objects Register of Incidents & Statistics shows that accidents caused by falling objects are one of the top 10 causes of death and serious injury in the oil and gas industry [1]. With the improvement of the modern economy and society, the world's consumption has increased rapidly in the past decades. Offshore platforms are considered basic facilities for exploitation of marine resources.

Along with the rapid development of ocean oil engineering, the activity of crane lifting and supply boat loading-unloading has so high frequency that drops from boat and crane are inevitable in the processes of production and operation [2]. The accident drop will cause damages to offshore structures, and the offshore platforms are considered as the major sources of risks to human, economics and environment. The falling of objects from lifting equipment and other superstructures on the platform deck frequently happens and presents threat to the structure of the platform, the safety of human onboard the platform, and the environmental safety of the ocean.

The dropped objects will damage the platform and even the equipment on the platform, so it will have unfavorable influences on the offshore production structure safe and serviceable life, even result in large number of personal death and unpredictable economy losses. According to an investigation in the offshore industry, the quantity of dropped objects of accidents ranks the first among all kinds of offshore structures, that is to say, dropped objects pose a great threaten to the safety of offshore structures [3]. As a major accident, more and more owners of offshore platforms concern the falling objects of platform, and put forward the safety analysis research activities of falling objects. In recent years, several studies have proposed relevant analytical method in the field of offshore risk assessment. Some items of the potential to cause injury, death, or equipment/environmental damage, that falls down or over from its previous position or operations.

Dropped objects may be further classified as static or dynamic. Some dropped object whose failure may be attributed to apply forces (e.g., from the impact of equipment, machinery, or other moving items, severe weather, or manual handling) [4]. As the practical experiment is not feasible under most circumstances, the nonlinear finite element numerical simulation method is usually used to examine the performance of the impacted ocean platform. Through simulating and analyzing of the impact process of falling objects and platforms, the general rules of collision force, structural damage deformation and energy conversion are obtained from the impact of falling objects on the ocean platform.

However, it is also not feasible to conduct the simulations under all circumstances. To achieve the effect of a specific circumstance, tedious work is needed for the nonlinear finite element simulation such as the construction of grid model of a falling object and the platform or pipelines to be impacted. To avoid this problem, a nonlinear mapping method is presented to achieve the accurate fitting of the nonlinear finite element method. The existing simulation results are utilized to train a neural network and the information are stored within the neurons and the connection parameters. When certain effects of falling object impact are needed, the generalization process is performed based on the established neural network.

2 Statistics of Platform Accidents

In the case of finite element analysis, it is necessary to analyze the potential factors of the platform or the drilling ship, identify the risk source of the risk of the object, and determine the name, type, location (plane location and height) of the risk source, and design the experimental scheme on this basis.

As the impact of object is complex and the effect is relevant to various factors such as the impact angle, the falling height, impact speed, the weight of the falling objects, the nature of the falling object and the nature of the platform or pipelines, which make it a high-dimension of nonlinear interpolation problem. Therefore, a partial least square (PLS) method is adopted by combining with the radial basis function (RBF) network. By combining nonlinear representation capability of RBF network and the high-dimension information processing ability of the partial least squares method, the resulted PLS-RBF network has better generalization capability than conventional RBF networks.

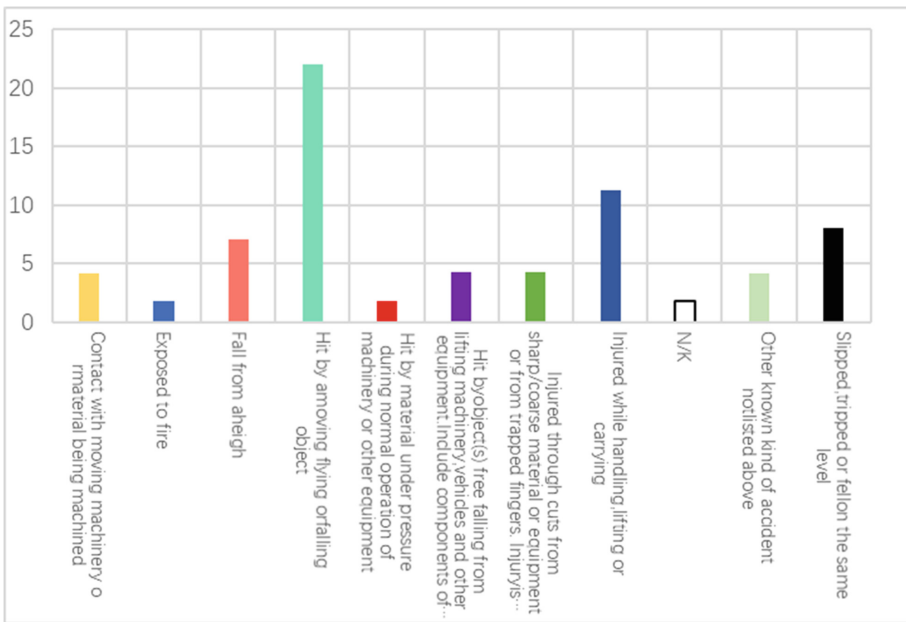


Fig. 1. The statistics of risk source

And objects hoisted from the ocean platform each year are classified according to their shape and weight as showed in Table 1 [5].

The kind of accident can easily be grouped into a number of condensed categories, which are generally parallel to the actual response provided. Figure 1 shows the kind of accident, using these condensed categories for the 67 incidents considered. Hit by moving, flying or falling object is the dominant kind of accident with over 20 cases. Falling off a height, injured while handling lifting or carrying and Slipped, tripped or fell on the same level also show some prominence. As can be seen from Table 1, the box container has the highest number of hoisting time [6]. Therefore, in the risk source of the falling object, the simulation experiment was conceived with the case design.

Table 1. The crane lifting objects statistics according to DNV-RP-F107

Number	Type	Weight (t)	Typical object	Offset angle (°) *1	Annual number of lifting*2	Total lifting ratio (%)
1	Flat pattern	<2	Drilling shaft/casing, scaffolding	15	700	17.50%
2		2–8	Drilling shaft/casing	9	50	1.25%
3		>8	Drill pipe, crane, derrick	5	5	0.12%
4	Square shape	<2	Light container (food, parts), crane, crane pulley	10	500	12.50%
5		2–8	Medium-sized container (food, parts), crane, crane pulley	5	2500	62.40%
6		>8	Heavy container (spare parts), bucket	3	250	6.24%
7	Square shape	>>8	Large items (such as blowout preventer), pipeline drum, etc.	2	0	0%

3 PLS-RBF Networks

RBF networks containing input, hidden and output layers. The input layer serves only as input distributor, each unit in the hidden layer represent the radial function, whose dimensionality being the same ones dimensionality of input data. The input value of each output unit is a weighted sum of all the outputs of hidden units. The architecture of a RBF network is shown in Fig. 2.

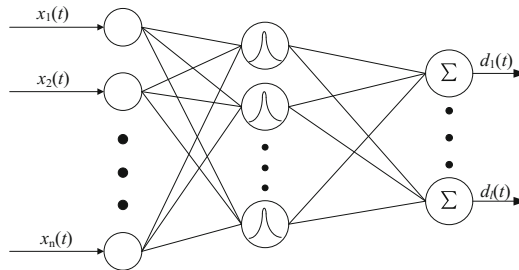


Fig. 2. Architecture of RBF network

Suppose that the input vector x consists of l dependent variables, and the data set available for modeling consists of m observations. So the dimension of the input matrix X is $m \times n$, and that of the output matrix Y is $m \times l$.

In this paper, Gaussian functions are used as RBFs to carry out nonlinear transformation of X and from activation matrix A :

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & & a_{2m} \\ & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mm} \end{pmatrix} \in R^{m \times m} \quad (1)$$

The elements of A are defined as:

$$a_{ij} = \exp\left(-\|c_j - x_i\|^2 / \sigma_j^2\right), j = 1, 2, \dots, m. \quad (2)$$

where x_i is a vector consisting of the values of variables taken from the i -th observation, a_{ij} is the element of A at the i -th row and j -th column, $\|\cdot\|$ is a norm that denotes the Euclidean distance here, c_j and δ_j are the center and width of the j -th RBF center, respectively. The parameter c_j is set by

$$c_j = x_j, j = 1, 2, \dots, m. \quad (3)$$

Thus, the resulting symmetrical matrix A has ones on its diagonal, we notice that the dimension of A is $m \times m$, which is independent of the number of variables in the input data set, but is determined by the number of observations in the set.

Then, the PLS procedure will be applied to the matrices A and Y . A and Y are projected on the low-dimensional score matrix T , respectively, so the linear PLS model is set up as:

$$Y = TR + F = AWR + F, \quad (4)$$

where $T \in R^{m \times n_T}$ is the low-dimensional score matrix of A , $R \in R^{n_T \times l}$ represents the regression coefficient matrix, $W \in R^{n_T \times l}$ is the transformation matrix of A and F is the residual matrix with the dimension of $m \times l$. Because of the fact that T is a linear combination of Gaussian functions that will maximize the covariance between A and Y , n_T plays an important role in the RBF-PLS network. When n_T is determined, the RBF-PLS network can be obtained and used for prediction or other purpose.

Therefore, by combining PLS algorithm with RBF network, the nonlinear relation between A and Y is transformed to the problem in linear algebra. After training, the RBF-PLS network can be used to make predictions new observations:

$$Y_p = A_p WR, \quad (5)$$

where A_p is the activation matrix of X_p which is used for prediction, Y_p is the resulting dependent matrix.

4 Simulation Based on LS-DYNA Finite Element Method

According to the results of finite element simulation, polynomial fitting and neural network analysis are carried out respectively, and the following conclusions can be brought. The fitting precision of neural network is higher than that of polynomial.

The finite element is the discrete elements that together represent the actual continuous domain. It regards the solution domain as consisting of many small interconnect subdomains called finite element, and assumes an appropriate approximate solution for each element, and then deduces the universal conditions of solving the domain, so as to obtain the solution of the problem. It has been demonstrated that the finite element method has high accuracy and can adapt to various complex shapes. The traditional energy method cannot describe the concrete deformation of the collision deformation region and the collision force. In order to solve a series of problems, a finite element method can be utilized to realize the process change in time [7].

Considering the process of the impact, explicit dynamic numerical simulation is utilized to analyze the nonlinear material response. This study uses ls-dyna to predict the consequence of the impact. The primary step is hazard identification and consequence analysis through numerical simulation. The main proposed method is to determine impact consequence through numerical simulation with explicit dynamic analysis software to assess whether the safety are controlled within the acceptance or not [8]. The finite element method for settling the general procedure is shown in Fig. 3.

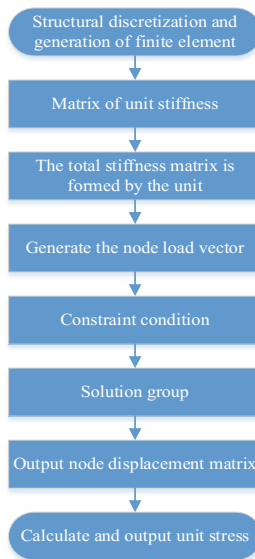


Fig. 3. Process of FEM analysis

Table 2. Experimental scheme

Operating condition	Mass (t)	Shape	Height of fall (m)	Impact location	Density (kg/m ³)	Impact velocity (m/s)	Elastic modulus	Poisson's ratio
Box1	10	Box	3	Deck	7850	7.67	2.10E+11	0.3
Box2	10	Box	5	Deck	7850	9.90	2.10E+11	0.3
Box3	10	Box	8	Deck	7850	12.53	2.10E+11	0.3
Box4	10	Box	10	Deck	7850	14.01	2.10E+11	0.3
Box5	10	Box	15	Deck	7850	17.16	2.10E+11	0.3
Box6	10	Box	20	Deck	7850	19.81	2.10E+11	0.3
Box7	20	Box	3	Deck	7850	7.67	2.10E+11	0.3
Box8	20	Box	5	Deck	7850	9.90	2.10E+11	0.3
Box9	20	Box	8	Deck	7850	12.53	2.10E+11	0.3
Box10	20	Box	10	Deck	7850	14.01	2.10E+11	0.3
Box11	20	Box	15	Deck	7850	17.16	2.10E+11	0.3
Box12	20	Box	20	Desk	7850	17.16	2.10E+11	0.3

4.1 Finite Element Model

On the basis of drilling platform, based on the real offshore platform deck, a typical finite element model with stiffeners was built-in the geometry of ANSYS [9]. Dropped object strikes a part of the deck. Part of the simulated deck, as the impact area, is x meters of x meters in size and $3e-02$ m in thickness. The 20-foot container was used for the landing, with a length of 6.058 m, a width of 2.438 m, a height of 2.591 m, a volume of 38.268 m^3 and a maximum load of 20.32t. All components of elastic modulus $E = 206 \text{ Gpa}$, Poisson's ratio = 0.3, (including density is $\rho = 7850 \text{ kg/m}^3$). The container falls from different heights with an initial vertical velocity of 0 and a gravitational acceleration of 9.81 m/s^2 . The durability of the deck is examined. Based on the above conditions, the experimental scheme is shown in Table 2.

Based on the known parameters, falling objects and deck models are designed in workbench, the force field is applied, and then meshing is carried out (Figs. 4 and 5).

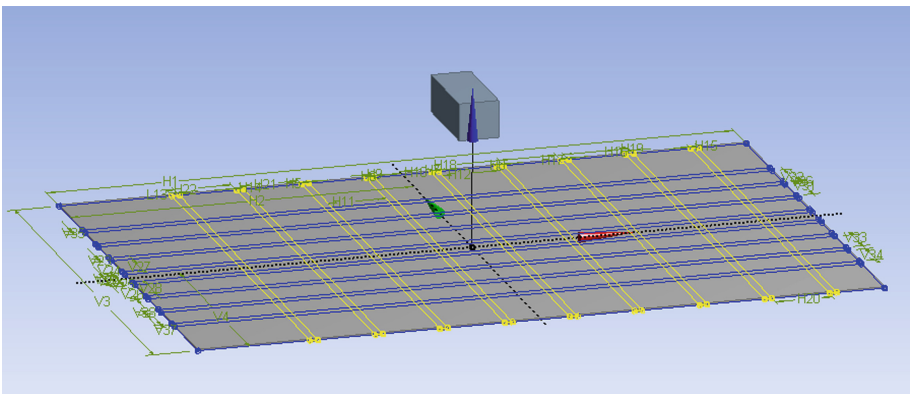


Fig. 4. Layout of container and deck

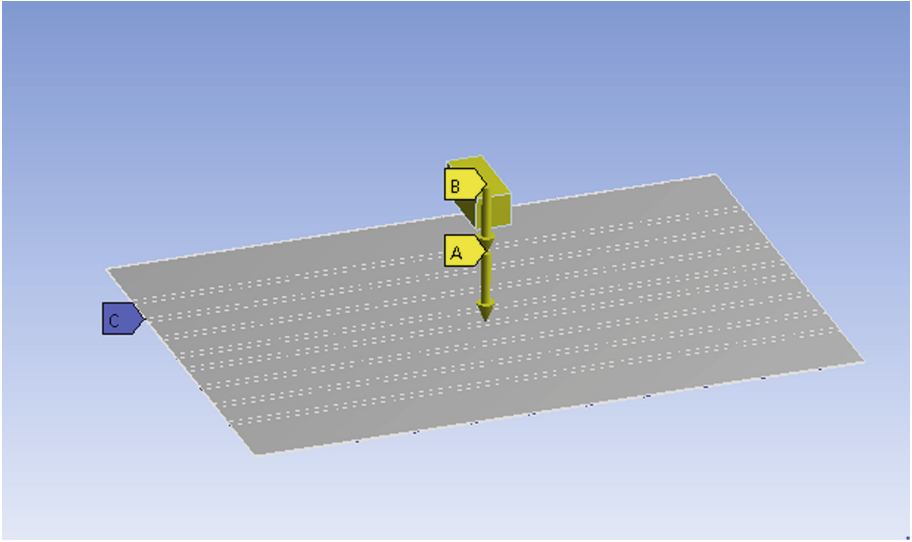


Fig. 5. Applying field force

4.2 Explicit Dynamic Analysis of Impact Problem

The interaction of the platform deck and dropped object is a typical impact problem, which is a complicated subject because of the high nonlinearity involved. Therefore, the ideal elastic-plastic model can be adopted, and the concrete parameters are selected according to the actual material characteristics of the platform structure [10]. The contact between the object and the platform deck is placed at adaptive contact. The most popular and effective collision problem is FE method. The control equation of dynamic problem can be given by:

$$[M]\ddot{x}(t) + [C]\dot{x}(t) + [K]x(t) = F(t), \quad (6)$$

where $[M]$, $[C]$ and $[K]$ are the mass, damping and stiffness matrices respectively. $X(t)$, the displacement, is velocity and acceleration of time t ; $F(t)$ is the external load. The implicit and explicit methods are presently two main solution methods which are widely applied to solve this equation. The explicit dynamic method is considered to have more potential in solving large Permanent deformation and highly nonlinearity problem. ANSYS/LS-DYNA is a powerful tool in solving these problems with an explicit dynamic solver, which is passed in this paper.

4.3 Numerical Simulation Results and Analysis

To evaluate the consequence of the impact on the platform deck material and its behavior, the result for the total deformation, equivalent stress and equivalent elastic strain are considered. Through explicit dynamic analysis, the deformation diagram and force diagram of the platform deck are obtained [11]. The structure of the falling object

collision platform is simulated and the energy conversion relationship during the collision is analyzed.

The results show that the collision of the object and the platform is in the vertical direction, the motion of the object by the action of gravity acceleration, because the elasticity of the material, makes the impact have repeatability. After the object is bounced back, the object will collide again due to gravity. The energy that its repeated impact is mostly absorbed by the first impact structure is not too damaging to the structure [12]. At the same time the initial kinetic energy of the falling object is mainly converted to the internal energy and kinetic energy of the platform. The total energy of the system is basically constant. Impact damage is mainly concentrated in the contact area [13]. The stress distribution on the deck and the degree of depression caused by impact can be obtained by finite element simulation experiment, as showed in the figure below (Figs. 6 and 7).

Maximum stress and deck sag during collision, under different conditions were calculated, as shown in Figs. 8 and 9.

As can be observed in the direction of the scatter diagram, as the falling height increases, the impact velocity increases, and the load duration decreases slightly with the increase of velocity. The peak of the impact force is proportional to the fall height, and the higher the fall height, the more dangerous the impact.

Take box 10 for example, Figs. 10 and 11 shows that the impact damage is localized and basically concentrated in the collision contact area. During the impact process, a considerable amount of energy is converted into kinetic energy of the platform. On the one hand, due to the impact energy of the object relative to the offshore platform is relatively small, on the other hand, the overall platform has good elastic deformation can reduce the impact of falling objects in the structural damage.

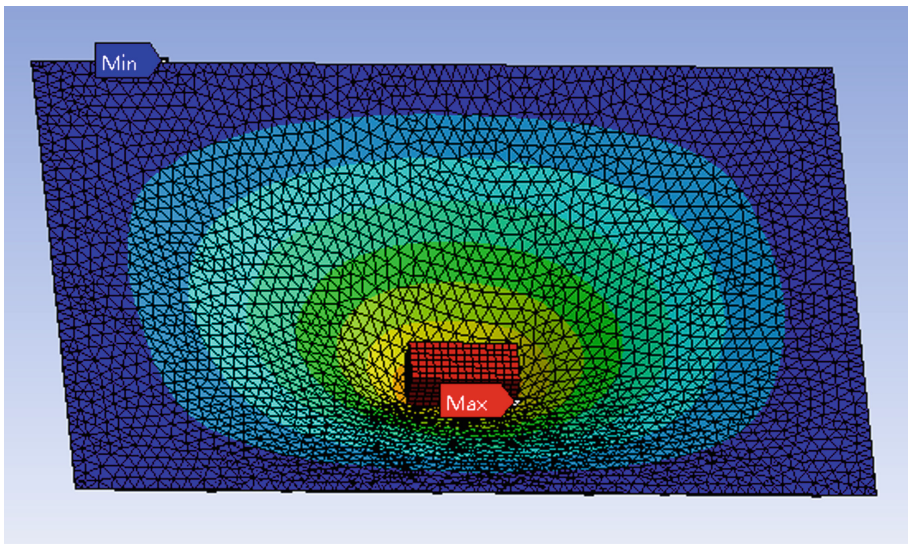


Fig. 6. Stress profile

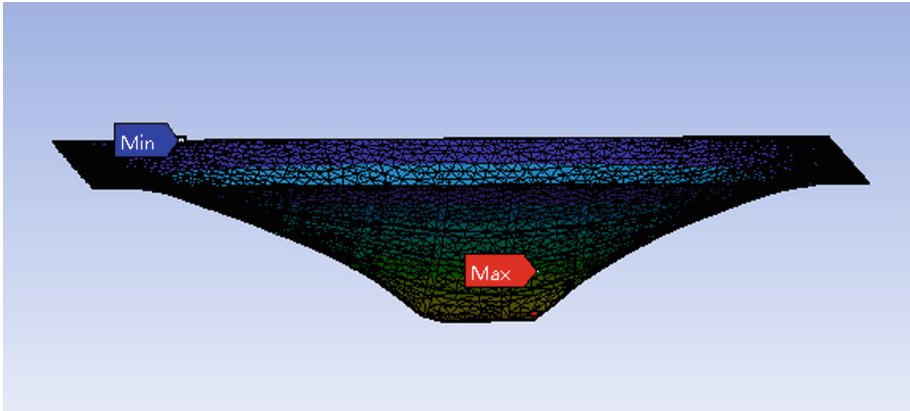


Fig. 7. Deck deformation

5 Numerical Analysis of Dropped Risers

According to the results of finite element simulation, polynomial fitting and PLS-RBF network analysis are carried out respectively, and the following conclusions are compared in this section. The identification result by using PLS-RBF network is shown in Figs. 12 and 13.

It is noticed in Fig. 12 that the PLS-RBF network fit the simulation results well with small identification errors. For comparison purpose, the conventional polynomial method is also implemented, and the identification result by using the polynomial method is shown in Fig. 13.

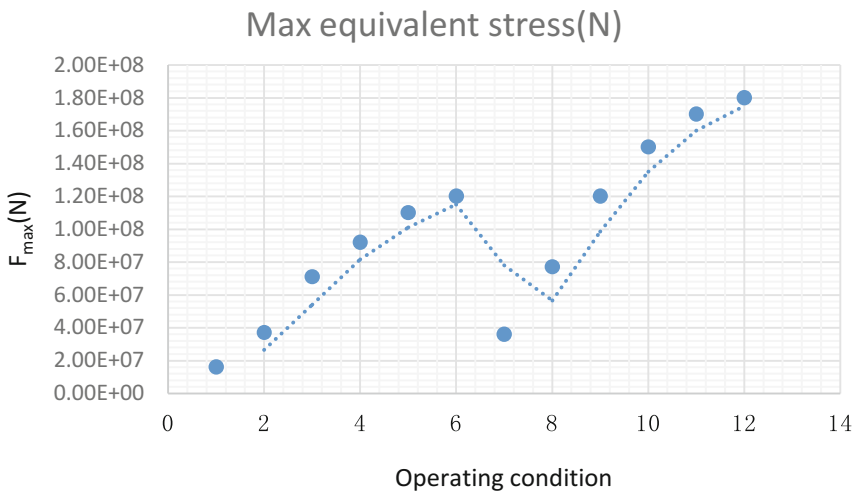


Fig. 8. Max equivalent stress

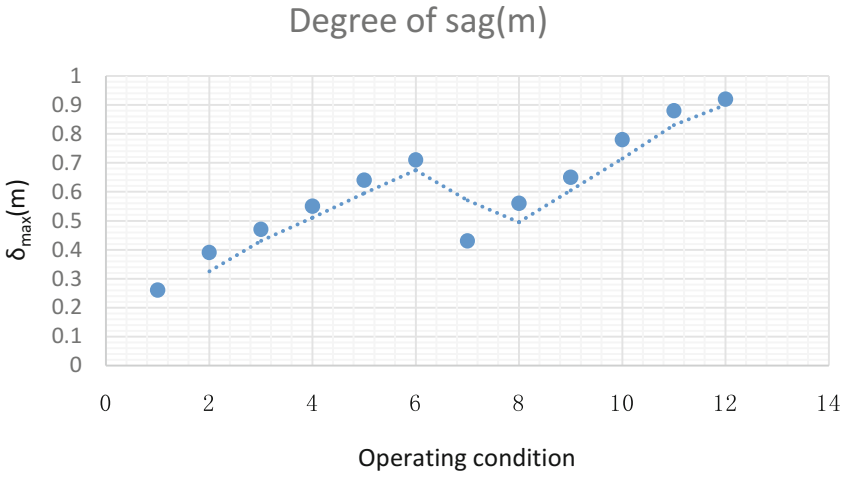


Fig. 9. Concavity results

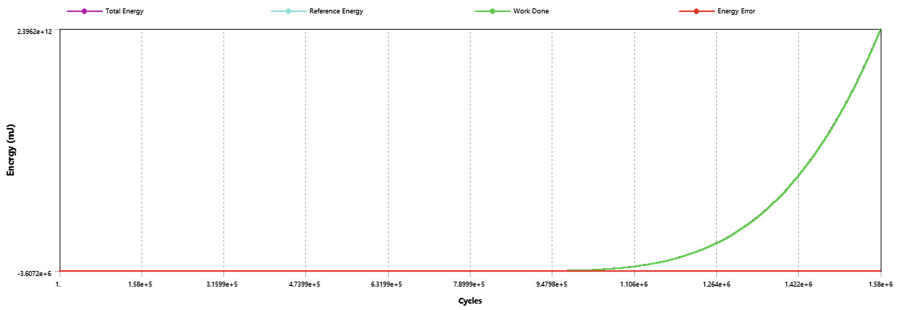


Fig. 10. Energy conservation (box 10)

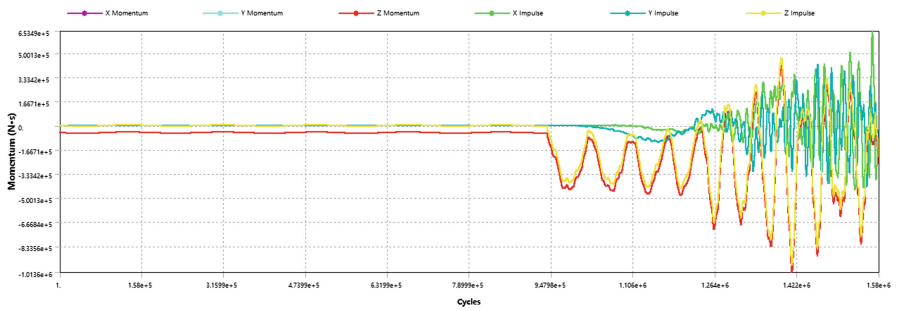


Fig. 11. Momentum summary (box 10)

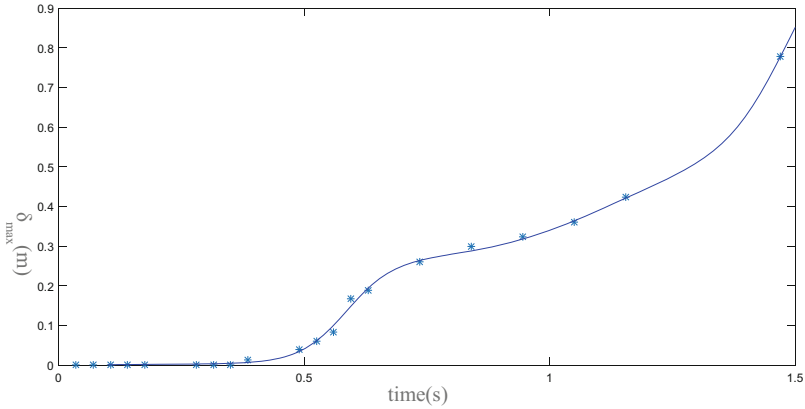


Fig. 12. PLS-RBF network fitting results (box 10)

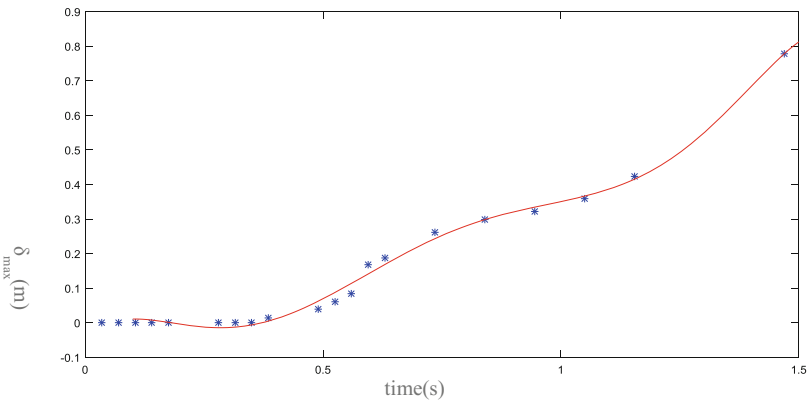


Fig. 13. Polynomial fitting results (box 10)

It can be noticed in Fig. 13 that the polynomial method can give accurate identification method either. To make comparison, the index of root of the mean square error is implemented in this study.

The generalization capability is the most important ability for evaluating the capability of neural networks [14]. In this study, the generalization capabilities of the PLS-RBF network and the polynomial method are compared. The time of 0.15, 0.7, 1.3, and 1.4 are set as inputs to the PLS-RBF network and the polynomial model, and the outputs are achieved and shown in Table 3. The points used to test the sample simulation were randomly selected to test the accuracy of the interpolation.

The root mean square error (RMSE) by using the PLS-RBF network is 0.0032, while the RMSE by using the polynomial method is 0.0284. That is, fitting generalization accuracy of PLS-RBF network is higher than that of polynomial fitting.

Table 3. Comparison of generalization capability

Methods/time	0.15	0.16	0.7	1.3	1.4
FEM simulation	0.0019	0.0019	0.2497	0.5135	0.6326
PLS-RBF network	0.0018	0.0019	0.2501	0.5107	0.6279
Polynomial method	0.0062	0.0044	0.2202	0.5485	0.6838

6 Conclusions

In the process of energy transformation, the kinetic energy loss, collision force and other variables are the dynamic changes of the impact condition, and the finite element method provides a powerful tool for analyzing the contact and collision problems. By using the finite element method to analyze the dynamic response of the upper structure of the container and the upper structure of the deck equipment, the following conclusion is obtained: The database of the impact of falling objects facilitate the practical application of nonlinear FEM; and the nonlinear mapping by using the PLS-RBF network can construct a model which possesses high accuracy which satisfy the practical need of accuracy. The risk assessment based on the neural network-based falling object impact database would be our further research efforts.

Acknowledgement. This work is supported by grant from the 7th Generation Ultra-Deep-water Drilling Rig Innovation Project, the Liaoning Natural Science Foundation of China, and the Natural Science Foundation of China under Grant 51609132.

References

1. Kawsar, M.R.U., Youssef, S.A., Faisal, M., et al.: Assessment of dropped object risk on corroded subsea pipeline. *Ocean Eng.* **106**, 329–340 (2015)
2. Le, C.H.: Risk assessment of offshore platform and peripheral installation due to collision damage. Tianjin University, Tianjin (2010)
3. Liu, C.: Finite element analysis of the notched legs of ocean platform structure. *J. Mech. Strength* **24**(4), 543–546 (2002)
4. Sun, L.P., Ma, G., Nie, C.Y., et al.: The simulation of dropped objects on the offshore structure. *Adv. Mater. Res.* **339**, 553–556 (2011)
5. Veritas, D.N.: Risk assessment of pipeline protection. Recommended practice No. DNV-RP-F107, 45 p. DNV, Oslo, Norway (2010)
6. Qin, T., Liu, C., Duan, M., et al.: Finite element analysis of cracked members of ocean platform structures. *Ocean Eng.* **18**(3), 15–19 (2000)
7. Oluwole, L., Odunfa, A.: Investigating the effect of ocean waves on gravity based offshore platform using finite element analysis software ANSYS. *Int. J. Sci. Eng. Res.* **6**(8), 24–33 (2015)
8. Song, Y., Wang, J.: Finite element method for design of reinforced concrete offshore platforms. *China Ocean Eng.* **7**(1), 27–36 (1992)
9. Nie, B.: Design and finite element analysis of water weights bag for ocean platform. *Petro-Chem. Equip.* **5**, 010 (2012)

10. Chen, Z.J., Yuan, J.H., Zhao, Y.: Impact experiment study of ship building steel at 450 MPa level and constitutive model of Cowper-Symonds. *J. Ship Mech.* **11**(6), 933–941 (2007)
11. Yin, J., Bi, G., Dong, F.: A partial least squares regression method for growing radial basis function networks. In: 2008 Chinese Control and Decision Conference 2008, pp. 2562–2565. IEEE (2008)
12. Malekzhehtab, H., Golafshani, A.A.: Damage detection in an offshore jacket platform using genetic algorithm based finite element model updating with noisy modal data. *Procedia Eng.* **54**, 480–490 (2013)
13. Wang, Y.Y.: Research on computation of hydrodynamic performance for oceanic floating platform in deep-water. *J. Dalian Univ. Technol.* **51**(6), 837–845 (2011)
14. Yin, J.C., Wang, N., Perakis, A.: A real-time sequential ship roll prediction scheme based on adaptive sliding data window. *IEEE Trans. Syst. Man Cybern.: Syst.* **99**, 1–11 (2017)



A Multiobjective Genetic Algorithm Based Dynamic Bus Vehicle Scheduling Approach

Hongyi Shi^{1,2(✉)}, Chunlu Wang^{1,2}, Xingquan Zuo^{1,2},
and Xinchao Zhao³

¹ School of Computer Science, Beijing University of Posts
and Telecommunications, Beijing, China
shihongyi@bupt.edu.cn

² Key Laboratory of Trustworthy Distributed Computing and Service,
Ministry of Education, Beijing, China

³ School of Science, Beijing University of Posts and Telecommunications,
Beijing, China

Abstract. Bus vehicle scheduling is very vital for bus companies to reduce operation cost and guarantee quality of service. Urban roads are easily blocked due to bad weather, such that it is significant to study the bus vehicle scheduling problem under traffic congestion caused by bad weather. In this paper, a dynamic bus vehicle scheduling approach is proposed, which consists of two parts: (1) generate a set of candidate vehicle blocks once the road is blocked; (2) adopt the non-dominated sorting genetic algorithm combined with a departure time adjusting process to select a subset of vehicle blocks from the candidate blocks set to form a vehicle scheduling scheme. Experiments show that our approach can significantly improve quality of service compared to the manual vehicle scheduling scheme.

Keywords: Bus vehicle scheduling · Dynamic vehicle scheduling
Multi-objective genetic algorithm · Urban bus scheduling

1 Introduction

In recent years, traffic congestion occurs frequently in big cities due to bad weather. The bus vehicles may be blocked due to the traffic congestion, such that vehicles cannot be scheduled according to the original scheduling scheme. In this case, the scheduling scheme must be dynamically adjusted to ensure the service quality.

Traditional vehicle scheduling methods consider the travel time of vehicles as a fixed one, and thus cannot deal with the uncertain travel time caused by traffic congestion. It is very significant to study the dynamic bus vehicle scheduling problem (DBVSP) to ensure the service quality and save operational cost. DBVSP is to assign vehicles to cover all the remaining trips in the bus timetable when traffic congestion occurs, i.e., make the departure times of vehicles coincide with all remaining times in the timetable.

There have existed numerous approaches for uncertainty bus vehicle scheduling problems [1–14]. For example, Sun *et al.* [1] proposed a hybrid cooperative

co-evolution algorithm combining a genetic algorithm and a cooperative co-evolution particle swarm optimization with the parameter self-adaptive mechanism to solve a uncertain bus vehicle scheduling problem. Petit *et al.* [2] proposed a discrete-time infinite-horizon approximate dynamic programming approach combined with an alternative bus substitution strategy for the dynamic bus vehicle scheduling problems. Zhu *et al.* [3] proposed a mathematical model for bus scheduling problem based on passenger-flow. Dynamic programming is presented to find its optimal solution.

In this paper, we propose a new dynamic bus vehicle scheduling approach based on our previous work [4]. In literature [4], we proposed a genetic algorithm-based scheduling approach to automatically generate a scheduling scheme for bus vehicles. It can only solve the static vehicle scheduling problem and cannot handle the uncertain travel time of trips caused by traffic congestion or bad weather. The proposed approach in this paper is an extension of the approach in [4]. It generates a new vehicle scheduling scheme based on the current scheduling environment when the traffic congestion occurs. First, it collects the current scheduling scheme and the time when the traffic congestion occurs, and then generates the candidate blocks set based on those trips that have not been finished. Secondly, it uses a multiobjective genetic algorithm (NSGA-II) to regenerate a new vehicle scheduling scheme based on the candidate blocks set.

2 Dynamic Bus Vehicle Scheduling Problem

Generally, there are two control points (CPs), CP1 and CP2, in most of bus lines. At each CP, drivers can have a rest and assume that the rest time is R . Each CP has limited space and can only accommodate limited number of vehicles, such that the maximum waiting time for the vehicle within a CP is stipulated. The time spent in a CP should not exceed the sum of the rest time R and the maximum waiting time W . A trip represents the travel of a vehicle from a CP to the other. Travel time refers to the time taken by a vehicle to perform a trip. The timetable contains a large number start times, which refer to the departure time of a trip. The start times in the two CPs may not be the same.

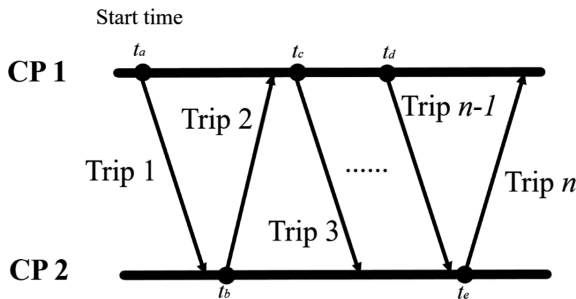


Fig. 1. The vehicle block.

As shown in Fig. 1, the vehicle block is defined as a collection of all the successive trips performed by a vehicle within a day. The vehicle block shown in Fig. 1 contains n trips assigned to a vehicle in one day. Generally, there are two types of vehicle blocks, namely long block and short block. The former requires one driver and the latter requires two. The operating time of a short block equal the maximum working time of a driver, Tw . The operating time of a long block is double the maximum working hour of a driver, $2Tw$. The vehicle scheduling problem is to arrange the vehicles to make their trips cover all the start times in the timetable.

When bad weather happens, some roads may be in serious congestion, resulting in longer travel time for trips. Assume that the travel time of all trips on certain road becomes longer during the period of bad weather. In this case, some start times in the timetable have already been covered by those executed trips, but all start times after the time when bad climate occurs have not been covered yet. Due to the longer travel time, those uncovered start times cannot be covered according to the original vehicle scheduling scheme. The purpose of dynamic vehicle scheduling is to generate a new vehicle scheduling scheme when traffic congestion happens, to cover all remaining start times that are not covered.

3 Dynamic Bus Vehicle Scheduling Approach

The proposed genetic algorithm based dynamic scheduling approach (GADSA) includes two steps: (1) generate a set of candidate vehicle blocks; and (2) choose some block subsets from the set of candidate blocks to construct the final solutions.

3.1 Generate a Set of Candidate Blocks

We consider all the remaining uncovered start times as a new timetable T . The initial start times in the timetable refer to the start times of the first trip of a vehicle block. Assume that the set of start times in $CP1$ is $T1$ while the set for $CP2$ is $T2$. The timetable, denoted by $T = T1 \cup T2$, contain n start times. All initial start times are expressed as $S = \{s_1, s_2, s_3, \dots, s_t\}, S \subset T$.

First, we calculate the current position of vehicles on work, and the time when they will arrive. As shown in the Fig. 2, those vehicles will arrive the destination CP later than their planed arriving time due to the bad weather. Thus, all start times to be covered by those vehicles cannot be covered. For each of those vehicles, when it arrives, an uncovered start time is chosen as its initial start time s_i . From s_i , generate the set of candidate blocks using the depth first search as follows.

Trip 1 is the first trip from s_i and its arriving time is t_a . Consider the start time $t \in [t_a + R, t_a + R + W]$ in the timetable, for instance t_b and t_c . Each of t_b and t_c can be selected as the departure time of the next trip. For example, trips 2 or 3 can be selected as the next trip. Each such selection constructs a possible vehicle block. Assume that we choose trip 3 as the next trip, we still have two start times t_d and t_e to be selected as the departure time of the next trip. Repeat above procedure to generate a set of blocks $B_{s_i} = \{b_{s_i}^1, b_{s_i}^2, \dots, b_{s_i}^{n_i}\}$, where $b_{s_i}^j (j = 1, 2, \dots, n_i)$ represents the j th trip in B_{s_i} departed

from s_i , and n_i represents the number of blocks whose first trip start from s_i . The whole block set is $B = \bigcup_{s_i \in S} B_{s_i} = \{b_1, b_1, \dots, b_{nb}\}$, where the total number of candidate blocks is $nb = \sum_{i=1}^t n_i$.

During constructing a block, the total work time T_e of a vehicle is compared with T_w (for short block) or $2T_w$ (for long block). Once the work time exceeds T_w ($2T_w$), the construction procedure ends. As suggested in [4], W is set to be $(t_b - t_a)\chi$, where $\chi \geq 1$ is a control factor to control the number of candidate blocks in B .

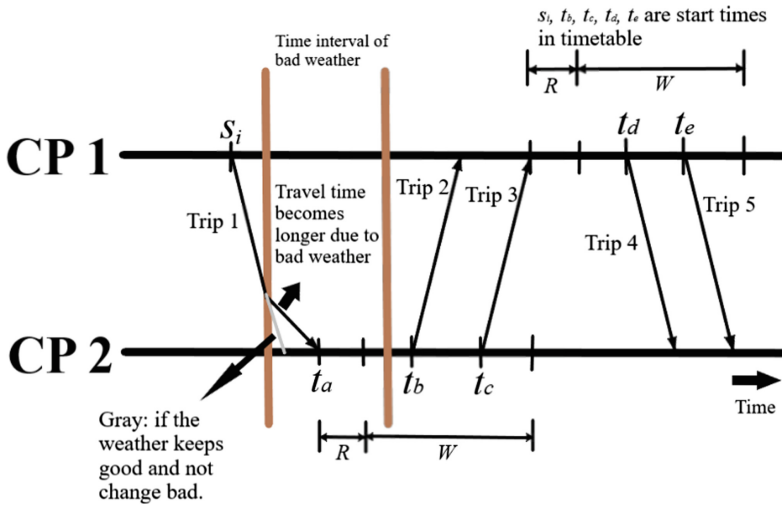


Fig. 2. Generate a block set when bad weather happens.

After generating the candidate blocks, there must exist some uncovered start times in the period of bad weather. To cover those start times, extra vehicles must be used. In this case, above depth first search is also used to generate the sets of candidate blocks for each of extra vehicles. Those generated blocks are added into the block set B .

3.2 Select Block Subsets to Form Pareto Solutions

After generating the blocks set B , a blocks subset $B^* \subset B$ is selected to cover all start times in T , with the objectives of minimizing the number of vehicles and drivers. There are a large number of blocks in B , thus that selecting a blocks subset B^* from B is a combinatorial optimization problem.

Besides the number of vehicles and drivers, to ensure the quality of solutions, we adopt another objective: the number of uncovered start times in the timetable. The solution coding in [4] is used to express a solution. Each gene in the code represents a block in the candidate block set. Instead of the binary tournament selection used in the original NSGA-II [5], we use the roulette selection [6]. A departure time adjustment

procedure is applied to each individual to improve its quality. The steps of NSGA-II are as follows:

Step 1: Let the number of generations, $gen = 0$. Generate N individuals by the initialization method to form the population of the first generation $P(gen)$. Set the archive set to be empty.

Step 2: Use a departure time adjustment procedure (DTAP) [4] to improve each individual in $P(gen)$, and then calculate its three objective function values.

Step 3: Individuals in $P(gen)$ perform crossover and mutation operations to obtain the population $Q(gen)$.

Step 4: Apply DTAP to each individual in $Q(gen)$ and then calculate its objective function values.

Step 5: Combine $P(gen)$ and $Q(gen)$ to construct a new population $R(gen)$.

Step 6: Use the roulette method to select N individuals from $R(gen)$ to form the population of next generation $P(gen + 1)$, and update the archive set.

Step 7: Let $gen = gen + 1$. The algorithm stops if it reaches the given number of generations; otherwise return to Step 2.

Initializing the population by a heuristic procedure is helpful to make the algorithm converge faster and obtain high quality solutions. Each solution in the initial population is initialized by the method in [4]. The initialization method in [4] only allows the first trip of each vehicle departure from CP1. In this paper, we extend the method in [4] to allow the first trip of a vehicle departure from CP1 or CP2.

Different from the original NSGA-II, an archive set is used to keep excellent solutions found so far. Suppose that the individual with the smallest number of uncovered start times in the population has x uncovered start times. Then, in each generation, all those individuals that have less than $(x + 10)$ uncovered start times are added into the archive set.

When the algorithm stops, the individuals with rank value 1 in the archive set construct the set of final Pareto solutions. Note that the archive set is not updated till 40 generations, since it is not possible to generate excellent individuals within a small number of generations.

3.3 Crossover and Mutation Operations

Each gene in the individual represents a vehicle block starting from a specific start time in timetable. Each individual in $P(gen)$ performs the single-point crossover operation with another randomly selected one according to a given probability P_c , resulting in two different new individuals. All generated new individuals are added to the population $Q(gen)$.

The mutation operation is to replace a randomly chosen gene with another vehicle block starting from the start time represented by the gene. Each individual in $P(gen)$ performs the mutation operation according to the probability P_m . If a gene mutates, it has a probability of 0.2 to be set to 0, which means that there is not vehicle starting from the start time represented by the gene, and has a probability of 0.8 to be set to a random vehicle block. All new individuals produced by the mutation operation are added into $Q(gen)$.

3.4 Calculate the Fitness Value

Each solution is evaluated by three objective functions. An individual X represents a set of vehicle blocks, $B_X = \{b_{X_1}, b_{X_2}, \dots, b_{X_{m_X}}\}$, where m_X is the number of used vehicles. m_X is the first objective function value, Obj_1 . Using the types of blocks (long or short block), the number of drivers can be calculated (the second objective function value, Obj_2). By applying DTAP to the individual, the departure times of some trips in B_X are adjusted, such that we can calculate the number of uncovered start times, which is the third objective function value, Obj_3 .

Utilizing the three objective function values, the fast nondominated ranking method in [5] is used to rank all solutions in the population and each individual is assigned a nondominated level represented by its ranking value. Individuals with the same ranking value indicate that they have the same nondominated hierarchy in the target space. Assume that an individual X has a ranking value i and r is the largest ranking value. Then, the fitness value of X is calculated by

$$FitValue(X) = r - i \quad (1)$$

If two individuals have the same ranking value, their fitness values are the same.

4 Experimental Results

The proposed approach (GADSA) is applied to a real-world bus line in Nanjing city, China, to verify its effectiveness.

The bus line information is shown in Table 1. During the peak and flat peak periods, the bus travel time is 35 min, while the bus travel time during the low peak period is 32 min. The peak periods are 6:30–9:00 and 17:00–20:00, the flat peak periods are 9:00–17:00 and 20:00–21:00, and the low peak periods are 4:30–6:30 and 21:00–01:05 (next day). The driver's minimum rest time is 8 min. A driver's maximum working time is not allowed to exceed 8 h. All vehicles begin their first trips from CP1 in this bus line.

Table 1. Information of the bus line.

Parameter	Value
Number of CPs	2
Total distance	10.7 km
Number of stations	17
The first start time	4:30 AM
The last start time	1.05 AM, next day
Number of start times for CP1	397
Number of start times for CP2	397
The maximum working time of the driver T_w	8 h
Maximum waiting time	4 min

Assume that the bad weather happens at 11:30 AM and lasts for 1 h. The travel time of all trips in this period becomes longer. The travel time of trips in this period equal the normal travel time multiplied by an abnormal coefficient which depends on the degree of traffic congestion. In this paper, we set the abnormal coefficient to be 1.2.

4.1 Algorithm Parameters

For GADSA, the code length of an individual is the number of start times that can be chosen as initial start times, namely 160. To ensure the population diversity, 40% of the initial individuals are generated by the initialization method in [4] and the remaining 60% of individuals are generated by the random method. The minimum rest time R is set as 8 min and the controlling factor is set to be 1.2 according to the setting in [4]. Parameters of the genetic algorithm are set to be $N = 800$, $P_c = 0.7$ and $P_m = 0.01$ after brief experiments. The maximum number of generations is given by 80.

4.2 Experimental Results

This approach is coded in Microsoft Visual C++ and run on a computer with Windows 10 operating system, 2.8 GHz i7 CPU, and 16G RAM.

First, the original vehicle schedule scheme is generated by the method in [4]. When the bad weather happens, GADSA regenerates a vehicle scheduling scheme. The set of candidate blocks generated by the depth first search in Sect. 3.1 contains 360 blocks (298 long blocks and 162 short blocks). The time used for generating the candidate blocks set is less than 1 s. The time of NSGA-II finding the final Pareto solutions is about 5 min.

Table 2. Pareto solutions generated by GADSA.

No. of solution	Obj_1	Obj_2	Obj_3	Time(s)
1	47	88	24	391
2	47	85	25	
3	47	84	26	
4	48	88	22	
5	50	89	19	
6	49	87	20	
7	46	82	30	
8	46	83	29	
9	47	82	28	
10	48	86	24	
11	48	84	25	
12	47	83	27	

Table 2 shows Pareto solutions generated by GADSA. We can see that the number of vehicles in those Pareto solutions is in the range [46, 50], the number of drivers fall within [82, 89] and the number of uncovered start times is in the range of [19, 30].

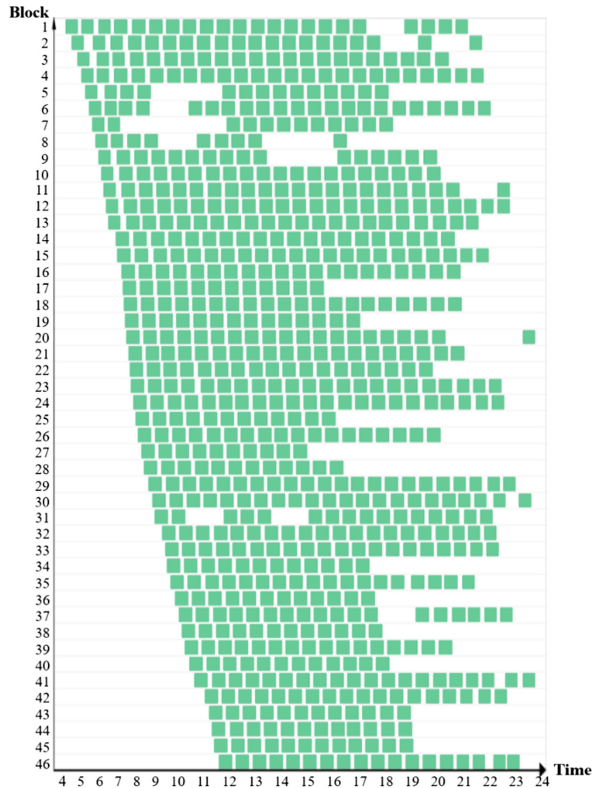


Fig. 3. The original vehicle scheduling scheme.

Figure 3 shows the original scheduling scheme generated by the method in [4]. The total number of used vehicles is 46. Almost all of the start times after 11:30 in the timetable cannot be covered when the bad weather happens at 11:30. There are 461 start times uncovered.

Figure 4 shows the 8th solution in Table 2. It has 46 vehicles (no extra vehicles are used). As shown in Table 2, the solution has 29 uncovered start times, which is only 1/10 of the number of uncovered start times in the original scheduling scheme shown in Fig. 3. It means that our method can greatly improve the service quality.

Our approach can generate a vehicle scheduling scheme with extra vehicles, based on practical requirements. If 2 extra vehicles are used to deal with the bad weather (the 4th solution in Table 2), then only 22 start times are not covered. The 4th solution can reduce 7 uncovered start times compared to the 8th solution that do not use extra vehicles. The comparison results are presented in Table 3.

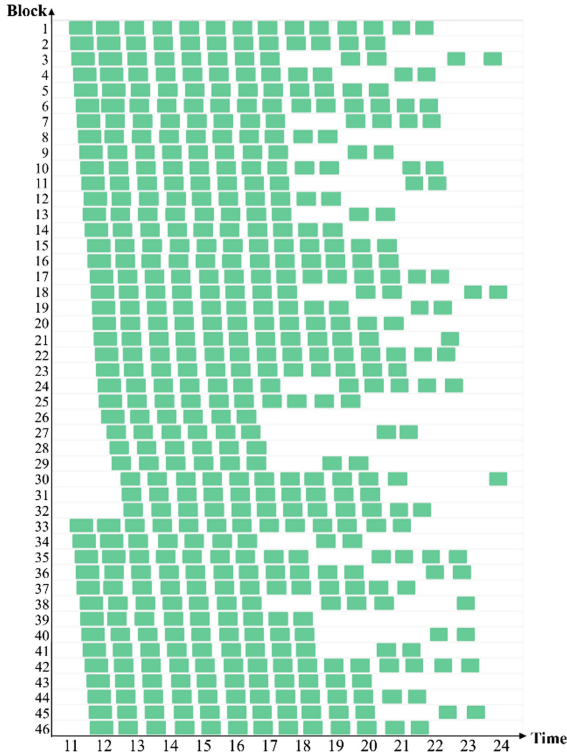


Fig. 4. The new generated scheduling solution (the 8th solution in Table 2).

Table 3. Comparison results

	Used vehicles	Number of uncovered start times
Original scheduling scheme	46	461
Solution 8	46	29
Solution 4	48	22

5 Conclusions

In this paper, we propose a genetic algorithm based dynamic bus vehicle scheduling approach. It can dynamically generate a scheduling scheme when traffic congestion caused by bad weather happens. First, all start times in the timetable that are not covered form a new timetable. Then, a candidate blocks set is generated by the depth first search. Finally, the nondominated sorting genetic algorithm combined with a departure time adjust procedure is used to generate the set of final Pareto solutions.

Experiments show that our approach can significantly decrease the number of uncovered start times in the timetable when traffic congestion occurs and produce a set of satisfactory vehicle scheduling solutions.

Acknowledgment. This work was supported by National Natural Science Foundation of China under Grant 61873040, 61374204, and 61375066.

References

1. Sun, L., Lin, L., Li, H., Gen, M.: Hybrid cooperative co-evolution algorithm for uncertain vehicle scheduling. *IEEE Access (Early Access)* **PP**, 1 (2018)
2. Petit, A., Ouyang, Y., Lei, C.: Dynamic bus substitution strategy for bunching intervention. *Transp. Res. Part B: Methodol.* **115**, 1–16 (2018)
3. Zhu, W., Li, R.: Research on dynamic timetables of bus scheduling based on dynamic programming. In: *Proceedings of the 33rd Chinese Control Conference*, pp. 8930–8934. IEEE, Nanjing (2014)
4. Zuo, X., Chen, C., Tan, W., Zhou, M.: Vehicle scheduling of an urban bus line via an improved multiobjective genetic algorithm. *IEEE Trans. Intell. Transp. Syst.* **16**(2), 1030–1041 (2015)
5. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **6**(2), 182–197 (2002)
6. Goldberg, D.: *Genetic Algorithm in Search. Optimization and Machine Learning*. Addison-Wesley, Boston (1989)
7. Lin, Y., Pan, S., Jia, L., Zuo, N.: A bi-level multi-objective programming model for bus crew and vehicle scheduling. In: *World Congress on Intelligent Control and Automation*, pp. 2328–2333. IEEE, Jinan (2010)
8. Yin, P., Chuang, Y., Lyu, S., Chen, C.: Collaborative vehicle routing and scheduling with cross-docks under uncertainty. In: *2015 IEEE Conference on Collaboration and Internet Computing*, pp. 106–112. IEEE, Hangzhou (2015)
9. Tan, D., Wang, J., Liu, H., Wang, X.: The optimization of bus scheduling based on genetic algorithm. In: *International Conference on Transportation, Mechanical, and Electrical Engineering*, pp. 1530–1533. IEEE, Changchun (2011)
10. Li, J., Hu, J., Zhang, Y.: Optimal combinations and variable departure intervals for micro bus system. *Tsinghua Sci. Technol.* **22**(3), 282–292 (2017)
11. Kwan, R., Wren, A., Kwan, A.: Hybrid genetic algorithms for scheduling bus and train drivers. In: *International Congress on Evolutionary Computation*, pp. 285–292. IEEE, La Jolla (2000)
12. Baghoussi, Y., Mendes-Moreira, J., Emmerich, M.: Updating a robust optimization model for improving bus schedules. In: *International Conference on Communication Systems and Networks*, pp. 619–624. IEEE, Bengaluru (2018)
13. Song, Y., Ma, J., Guan, W., Liu, T., Chen S.: A multi-objective model for regional bus timetable based on NSGA-II. In: *2012 IEEE International Conference on Computer Science and Automation Engineering*, pp. 185–188. IEEE, Zhangjiajie (2012)
14. Lin, K., Hashimoto, M., Li, Y.: Near-future traffic evaluation based navigation for automated driving vehicles considering traffic uncertainties. In: *2018 19th International Symposium on Quality Electronic Design*, pp. 425–431. IEEE, Santa Clara (2018)



Research on the Addition, Subtraction, Multiplication and Division Complex Logical Operations Based on the DNA Strand Displacement

Chun Huang^{1,2}, Yanfeng Wang², and Qinglei Zhou¹✉

¹ School of Information Engineering, Zhengzhou University,
Zhengzhou 450001, China

huangchunzzuli@yeah.net

² College of Electrical and Electronic Engineering,
Zhengzhou University of Light Industry, Zhengzhou 450002, China

Abstract. In recent years, the development of biological computers has become faster and faster. In order to adapt to the development of science and technology, some novel logic circuits designs are necessary. In this paper, a complex molecular combinational logic circuit is designed to perform large-scale biological information processing by compound logic operation of addition, subtraction, multiplication and division. The design of compound logic operations of addition, subtraction, multiplication and division based on DNA strand has four inputs signals and two output signals, and the dual-track design theory is used to perform the logic circuit operation, and the corresponding output result can be obtained in timely and accurately. The whole reaction progress of logic circuit operations can be programmed and simulated by the Visual DSD software. According to the simulation results of the Visual DSD software, the method of DNA strand displacement is feasible to achieve more complex logic computations. This investigation for the addition, subtraction, multiplication and division complex logical operations of a binary number may have a great prospect for the development and application in the biological information processing, medicine diagnosis, molecular computing, and so on.

Keywords: DNA strand displacement
Addition subtraction multiplication and division · Dual-rail circuit
Visual DSD

1 Introduction

In the current era of computers, DNA computing is a new research direction, which is included in the computer science and molecular biology [1–3]. Many scientists have demonstrated that DNA computing has the superior capabilities for processing and delivering information [4]. Therefore, with the progress of DNA nanotechnology [5, 6], the dynamic DNA nanotechnology has also been made a new difference, which could take the place of traditional silicon materials, and some simple logic computing functions have been realized, but there is still a large gap between theory and practical

application. Because the biochemical reaction requires more stringent conditions, such as temperature, rate of combination, and so on. It is difficult to control the reaction process and the low success rate [7, 8]. Therefore, only when these problems are solved can the DNA computer have a leap forward. With the appearance of DNA strand displacement, this situation has been changed greatly. Because the DNA strand displacement occurred at room temperature does not require any enzymes due to its simplicity, scalability, and modularity for large-scale applications [9–11].

Acted as the computing tool, DNA has solved many problems, such as Hamilton path, maximal clique problem [12–15]. DNA strand displacement technology is acted as a kind of dynamic DNA nanotechnology in the biological computing field [16]. Due to a series of characteristics of spontaneity, sensitivity and accuracy, DNA strand displacement technology has also been widely used in nano-machines, molecular logic circuits, nano-medicine and other fields [17, 18]. In recent years, the biological computer has been concerned widely by many scientists, who come from different fields, and molecular logic circuit is an important part of biological computer [19, 20]. Hence, the method of logic circuit design plays an important role in the biological computer.

DNA computing has processed a lot of molecule operations, such as self-assembly, fluorescence labeling, strand displacement, probe machine, and so on. Based on the strand displacement cascade reaction, the dynamical connection of the adjacent logic modules has been achieved, which makes it possible for the researchers to construct complex logic circuits [21, 22]. Moreover, with the advantages of high-capacity information accumulation, high performance parallel computing, programming and simulating, DNA strand displacement technology has been acquired an in-depth study in the fields of molecular computing, nano-machine, diagnosis and remedy of the disease. DNA strand displacement technology has a great potential in solving math problem, managing the nano-machine and discussing the computation science [23]. In addition, the construction of the biochemistry logic circuits based on DNA strand displacement has a significant research meaning by mastering the design procedures. This strategy for the addition, subtraction, multiplication and division complex logical operations of a binary number based on DNA strand displacement has a great application prospect in the fields of intelligent stimulus response materials, information processing, medicine diagnosis, molecular computing, biosensors and so on [24, 25].

In this paper, the multi-functional complex logic circuit of four logical operations of addition, subtraction, multiplication and division is constructed by using the mechanism of biological chain reaction, which enhances the ability of processing data in biological computers. The method of DNA strand displacement has great research significance in the field of mathematics and could be applied to molecular computers in the future.

Compared with the traditional circuit design methods, in this paper, the addition, subtraction, multiplication and division complex logical operations of one-bit binary number design based on DNA strand displacement has more accurate in the biological computer, and the strategy based on DNA strand displacement is firstly applied in the design of the addition, subtraction, multiplication and division complex logical operations [21]. Second, the design of the addition, subtraction, multiplication and division complex logical operations circuit with a high safety factor is a special case of intelligent testing, which does not require any chemicals, and the addition, subtraction,

multiplication and division complex logical operations logic designed in this paper has a high value in the biological computer digital logic circuit field [23]. Third, compared with the single logical operation, the results of the design of the complex logical operations of one-bit binary number design based on DNA strand displacement are displayed and implemented by using the dual-rail method, eliminating a lot of unnecessary links and saving a lot of manpower, material, etc. [25]. In addition, the dual-rail circuits based on DNA strand displacement can also reflect the blood group pairing nano logic circuit toward a large-scale direction.

In this study, the remaining content is given as follows: the mechanism about the DNA strand displacement is shown in the Sect. 2. The Sect. 3 is the design of the addition, subtraction, multiplication and division complex logical operations of one-bit binary number logic circuit, and then transformed from the digital logic circuit to the seesaw circuit. The results of simulation are presented in the Sect. 4. Finally, the conclusion for the addition, subtraction, multiplication and division complex logical operations of one-bit binary number circuit on the basis of DNA strand displacement by the dual-rail circuits are given in the Sect. 5.

2 The Reaction of DNA Strand Displacement

DNA strand displacement technique is originated from DNA self-assembly technique, in which DNA single strand can come into being the course of orderly multi-dimensional assembly spontaneously on account of base complementary pair rules [6–8]. In the DNA strand displacement reaction, once the initial DNA species are mixed together, the DNA strand displacement system can proceed autonomously. Consequently, DNA strand displacement reaction is a dynamic process. Ultimately, a target single strand can be displaced through an invading single strand from the sophisticated DNA duplex. The specific reaction process of DNA strand displacement is shown in Fig. 1.

Due to the toehold domain “T” hybridizing with the exposed complement toehold domain “T*” along the DNA duplex, a single strand $\langle T X Y \rangle$ is bound to the DNA duplex $\{T^*\} [X Y T] \langle Z \rangle$ in the first reaction from Fig. 1. The intermediate complex is produced on the right of the reaction. Owing to the strand $\langle T X Y \rangle$ is only fixed by a short toehold, the reaction is reversible. In the second reaction of Fig. 1, the overhanging strand $\langle X Y \rangle$ matches the bottom strand $\langle X^* Y^* \rangle$ of the DNA duplex along the double-stranded backbone because of the principle of the Watson-Crick base complementary pairings, the intermediate complex is generated on the right of the reaction. The intermediate complex is attached through a short toehold, the reaction is reversible. In the third reaction of the Fig. 1, the leftmost strand spontaneously unbinds on account of the low binding strength of the toehold. On the right of the reaction, the single strand $\langle X Y T Z \rangle$ can be bound to the DNA duplex $[T X Y] \{T^*\}$ owing to the toehold domain “T” is complemented with the toehold domain “T*”. Therefore, the reaction is also reversible.

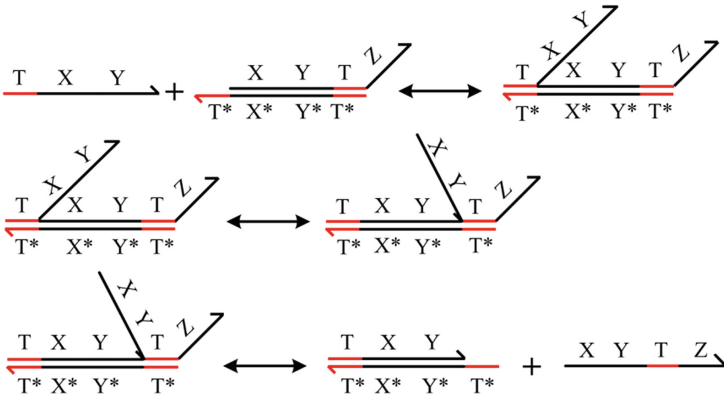


Fig. 1. The reaction mechanism of the DNA strand displacement. The “T” represents a short toehold domain and the toehold domain “T*” is the complementary pairing of the domain “T”. The strand $\langle T X Y \rangle$ is the input strand. The strand $\{T^*\} [X Y T] \langle Z \rangle$ is the complex DNA duplex. The strand $\langle X Y T Z \rangle$ is the output strand.

3 Complex Logical Operations of One-Bit Binary Number

3.1 Digital Logic Circuit

In this section, the logic operation is assumed by two kinds of states “0” and “1”. In the digital electronic circuits, there are three basic logical operations of logic algorithm, which are named with logic AND, logic OR and logic NOT. If the values of two input signals are “0”, then the value of logic OR gate is “0”, otherwise the value of logic OR gate is “1”. If the binary numbers of two inputs are all “1”, then the value of logic AND gate is “1”, otherwise the value of logic AND is “0”. The logic NOT implements that output states are the inverse of the input states. The four different binary numbers are computed by using the addition, subtraction, multiplication and division complex logical operations of one-bit binary number circuit (in Fig. 2) without the low-level borrow-bit.

In the digital circuit of the addition, subtraction, multiplication and division complex logical operations of a binary number circuit, according to the different logic input values, different logical operations are performed, and the diversification of the arithmetic functions greatly improves the data processing speed of the biological computer. The truth tables of the addition, subtraction, multiplication and division complex logical operations are shown in Table 1(a-d).

In this logic circuit about the addition, subtraction, multiplication and division complex logical operations of one-bit binary number circuit, which based on DNA strand displacement, four input signals are A, B, C and D, one output signal is Y. According to the different input signals, the output signal may be identical. In these input signals, four binary numbers are used to represent the addition, subtraction, multiplication and division complex logical operations of one-bit binary number signal. When the input signal $C_1C_0 = 00$, the logic circuit performs an addition logic

Table 1. The truth table of complex logical operations of one-bit binary number.

(a)		
Addition	Inputs	Results
$C_1 C_0$	A B	L M
0 0	0 0	0 0
0 0	0 1	1 0
0 0	1 0	1 0
0 0	1 1	0 1

(b)		
Subtraction	Inputs	Results
$C_1 C_0$	A B	L M
0 1	0 0	0 0
0 1	0 1	1 1
0 1	1 0	1 0
0 1	1 1	0 0

(c)		
Multiplication	Inputs	Results
$C_1 C_0$	A B	L M
1 0	0 0	0 0
1 0	0 1	0 0
1 0	1 0	0 0
1 0	1 1	1 0

(d)		
Division	Inputs	Results
$C_1 C_0$	C D	L M
1 1	0 0	0 1
1 1	0 1	0 0
1 1	1 0	0 1
1 1	1 1	1 0

operation. When $C_1C_0 = 01$, the logic circuit performs a subtraction logic operation. When $C_1C_0 = 10$, the logic circuit performs multiplication logic operation, when $C_1C_0 = 11$, the logic circuit performs the division multiplication logic operation, and the output signals are represented by L and M. When $L = 1$, it indicates that the output result is generated, and when $L = 0$, it indicates that there is no output result; if the output signal $M = 1$, which means that there is a borrowing operation when performing the logical operations, otherwise there is no borrowing operation. According to the logic value of the input signal AB, the value of the output result is also different, so that when performing complex logic circuit operations, a large number of logic circuit operations are involved.

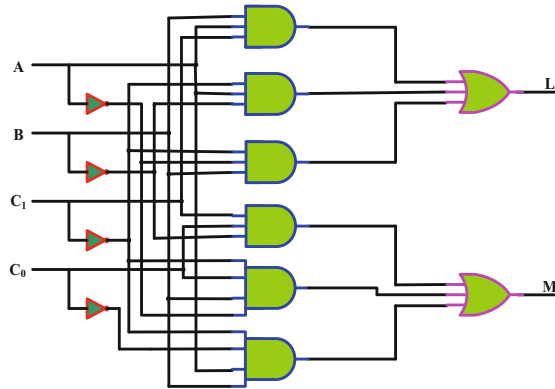


Fig. 2. Digital logic circuit of the complex logical operations of one-bit binary number.

3.2 Dual-Rail Logic Circuit

In this part, to avoid erroneous output signal generation, a two-track logic circuit design idea is proposed, and the design idea of the dual-rail logic circuit can play a big role in the whole process of composite logic operation. Through the design method of dual-rail logic circuit, more large-scale complex circuit operation results can be calculated quickly and accurately, moreover, this method can bring a lot of convenience to the development of biological computers. The two-layer circuit contains of eight AND logic gates and eight OR logic gates.

In the dual-rail logic circuits, each of original input signals is converted into two input signals, one of which can be expressed as logic “ON” or “OFF”. For example, if the input signal A can not be participated in the reaction, then the A^0 and A^1 will be shown logic “OFF” and logic “ON” in the dual-rail logic circuit, respectively. In the dual-rail logic circuit, every AND, OR and NOT logical function should be implemented by a series of OR logic gate ($W_1, W_3, W_5, W_7, W_9, W_{11}$) and a series of AND logic gate ($W_2, W_4, W_6, W_8, W_{10}, W_{12}$). The dual-rail logic circuit of the addition, subtraction, multiplication and division complex logical operations of one-bit binary number circuit based on DNA strand displacement is shown in Fig. 3.

For example, when the input signals $C_1C_0 = 00$, then $C_1^1 = 0, C_0^0 = 1, C_1^0 = 0, C_1^1 = 1$, and an addition logic operation is performed, if input signals $AB = 01$, then $A^0 = 1, B^1 = 1, A^1 = 0, B^0 = 0$. From top to bottom, the OR gate W_1 is operated by $A^0 = 1, B^0 = 0, C_1^0 = 1$ and the output result is $P_1 = [(A^0 = 1) \vee (B^0 = 0) \vee (C_1^0 = 1)] = 1$; The AND gate W_2 is operated by $A^1 = 0, B^1 = 1, C_1^1 = 0$, and the output result is represented by $P_2 = [(A^1 = 0) \wedge (B^1 = 1) \wedge (C_1^1 = 0)] = 0$; The OR gate W_3 is operated by $A^0 = 1, B^1 = 1, C_1^1 = 0$, and the output result is $P_3 = [(A^0 = 1) \vee (B^1 = 1) \vee (C_1^1 = 0)] = 1$; The AND gate W_4 is operated by $A^1 = 0, B^0 = 0, C_1^0 = 1$, the output of which is represented by $P_4 = [(A^1 = 0) \wedge (B^0 = 0) \wedge (C_1^0 = 1)] = 0$; The OR gate W_5 is operated by $A^1 = 0, B^0 = 0, C_1^1 = 0$, and the output result is $P_5 = [(A^1 = 0) \vee (B^0 = 0) \vee (C_1^1 = 0)] = 0$; The sixth AND gate W_6 is

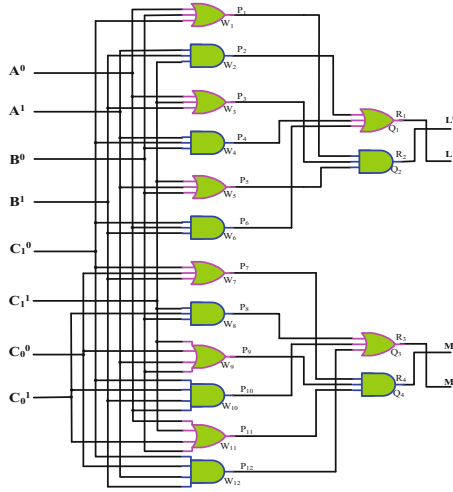


Fig. 3. The dual-rail logic circuit of the complex logical operations of one-bit binary number.

operated by $A^0 = 1, B^1 = 1, C_1^0 = 1$, the output of which is represented by $P_6 = [(A^0 = 1) \wedge (B^1 = 1) \wedge (C_1^0 = 1)] = 1$; The seventh OR gate W_7 is operated by $A^1 = 0, B^1 = 1$, and the output result is $P_7 = [(B^1 = 1) \vee (A^1 = 0)] = 1$; The eighth AND gate W_8 is operated by $B^0 = 0, C_1^1 = 0, C_0^0 = 0$, the output of which is represented by $P_8 = [(B^0 = 0) \wedge (C_1^1 = 0) \wedge (C_0^0 = 0)] = 0$. The ninth OR gate W_9 is operated by $A^1 = 0, B^0 = 0, C_1^1 = 0, C_0^0 = 1$ and the output result is $P_9 = [(A^1 = 0) \vee (B^0 = 0) \vee (C_1^1 = 0) \vee (C_0^0 = 1)] = 1$; The tenth AND gate W_{10} is operated by $B^0 = 0, C_1^1 = 1, C_0^1 = 0, B^1 = 1, A^0 = 1$, the output of which is represented by $P_{10} = [(B^0 = 0) \wedge (C_1^1 = 0) \wedge (C_0^1 = 1) \wedge (A^0 = 1)] = 0$. The eleventh OR gate W_{11} is operated by $A^0 = 1, B^0 = 0, C_1^1 = 0, C_0^0 = 0$, and the output result is $P_{11} = [(A^0 = 1) \vee (B^0 = 0) \vee (C_1^1 = 0) \vee (C_0^0 = 0)] = 1$; The twelfth AND gate W_{12} is operated by $C_1^1 = 1, B^1 = 1, A^1 = 0$, the output of which is represented by $P_{12} = [(B^1 = 1) \wedge (A^1 = 0) \wedge (C_1^1 = 1)] = 0$.

Then, the outputs results P_1, P_3 and P_5 are performed the AND operation through the AND gate Q_2 , the output result is represented by R_1 , that is $R_1 = [(P_1 = 1) \wedge (P_3 = 1) \wedge (P_5 = 0)] = 0$, so the final output result is $L^0 = 0$; The outputs results P_2, P_4 and P_6 are carried on the operation by the next stage OR gate Q_1 , and the output result is indicated by R_2 , that is, $R_2 = [(P_2 = 0) \vee (P_4 = 0) \vee (P_6 = 1)] = 1$, the final output result is $L^1 = 1$. The outputs results P_7, P_9 and P_{11} are performed the AND operation through the AND gate Q_4 , the output result is represented by R_3 , that is $R_3 = [(P_7 = 1) \wedge (P_9 = 1) \wedge (P_{11} = 1)] = 1$, so the final output result is $M^0 = 1$; The outputs results P_8, P_{10} and P_{12} are carried on the operation by the next stage OR gate Q_3 , and the output result is indicated by R_4 , that is, $R_4 = [(P_8 = 0) \vee (P_{10} = 0) \vee (P_{12} = 0)] = 0$, the final output result is $M^1 = 0$.

From the outputs results value for $L^1 = 1$ and $M^1 = 0$, which shows that there is an output result produced, but there are no carry operations here.

3.3 Seesaw Circuit for Complex Logical Operations of One-Bit Binary Number Circuit

In this paper, the seesaw circuit is used as the basic component of molecular logic circuit unit, which is composed of six kinds of DNA strands: input strand, output strand, threshold strand, fuel strand, gate strand, gate compounds. The DNA seesaw logic gates are designed on the basis of DNA strand displacement reaction. To simplify the description, the seesaw logic gates are abstracted as a form of a node, and there are several lines on the node in which the lines can connect other nodes, are shown in Fig. 4. The red digital -1.2 , -2.4 and -0.6 are all the threshold values, the value of fuel is two times of the total output values. One-input-two-output amplifying gate, one-input-three-output amplifying gate and one-input-four-output amplifying gate are displayed in Fig. 4(a)–(c), respectively. Two-input-one-output integration gate and three-input-one-output integration gate are shown in Fig. 4(d)–(e), respectively. The seesaw motifs of the AND gates are shown in Fig. 4(f)–(g), which include two-input AND gate and three-input AND gate. The seesaw motifs of the OR gates are shown in Fig. 4(h)–(i), which contain two-input OR gate and three-input OR gate. In the DNA seesaw logic gates, the digit “0” and digit “1” are expressed by the low concentration and the high concentration, respectively. These seesaw gates can be used to construct the seesaw logic circuit.

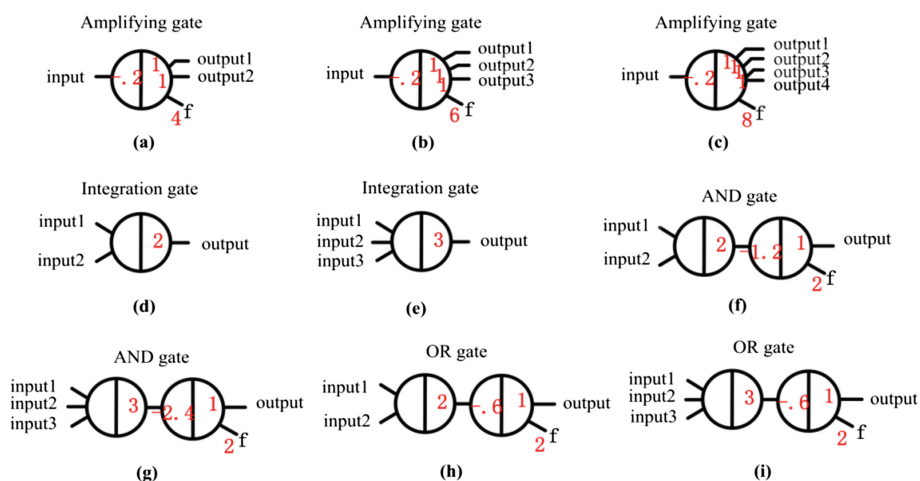


Fig. 4. The seesaw motifs of basic gates. (a) The amplifying gate of one-input-two-output; (b) The amplifying gate of one-input-three-output; (c) The amplifying gate of one-input-four-output; (d) The integration gate of two-input-one-output; (e) The integration gate of three-input-one-output; (f)–(g) Abstract diagrams of the seesaw AND gates; (h)–(i) Abstract diagrams of the seesaw OR gates. (Color figure online)

Seesaw biochemical logic circuit applied to the logic gates of biochemical reactions mainly includes the following four gates: amplifying gate, integrating gate, threshold gate and report gate. In the seesaw cascade circuit, amplifying gate includes threshold and fuel produces multiple outputs. If the total concentration of the input signals is greater than the threshold's concentration, then the output signals can be gained, otherwise, there are not any signals obtained. In order to promote the output signals fully released, the initial concentration of fuel is usually the twice concentration of the binding output signals. The function of the integrating gate is opposite to the amplifying gate, which is used to receive multiple inputs and integrated into an output signal after the reaction. In addition, either AND or OR logic operation with threshold gate can be performed by one integrating gate. The function of the threshold gate is that it can filter the input signals through the magnitude of concentration. In this article, according to the theoretical design requirements of experiment, the threshold values of OR gate and AND gate are "0.6" and "1.2", respectively. The whole seesaw logic circuit of the addition, subtraction, multiplication and division complex logical operations of one-bit binary number design by dual-rail is shown in Fig. 5. To see the relationship between the different logic gates more straightforward clearly, the different strands are used to represent the different logic gates in the whole seesaw logic circuit, respectively.

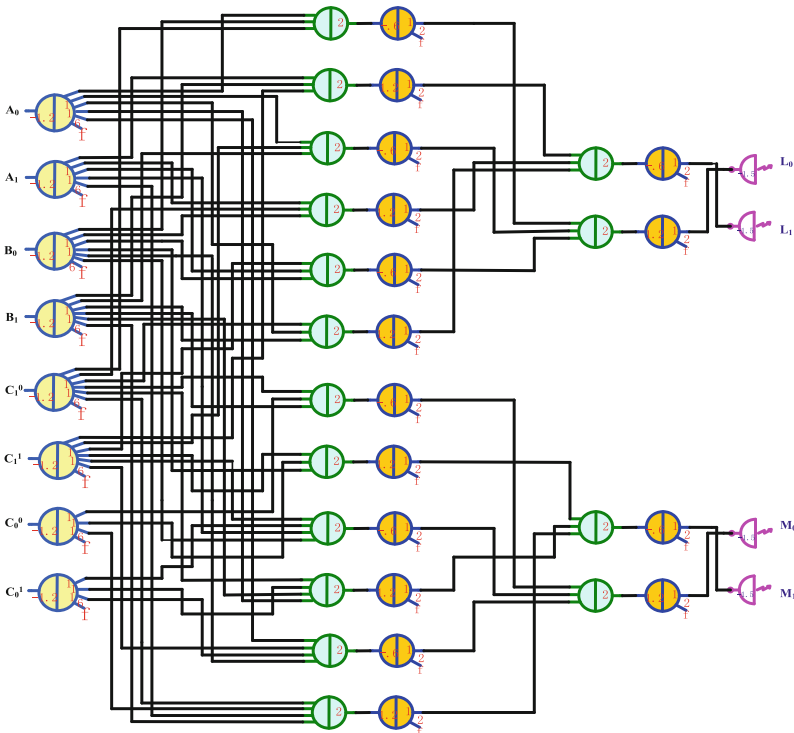
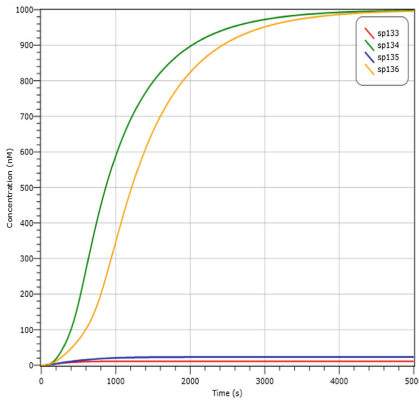
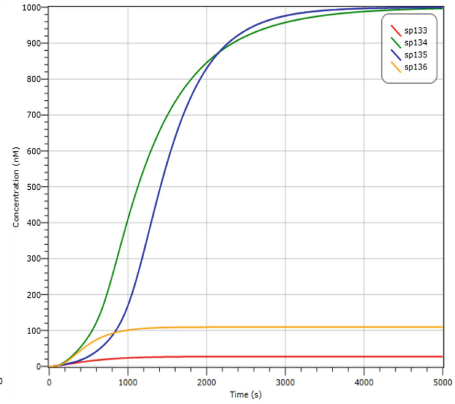


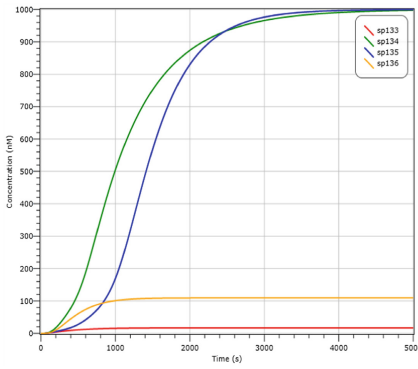
Fig. 5. The seesaw circuit of the complex logical operations of one-bit binary number.



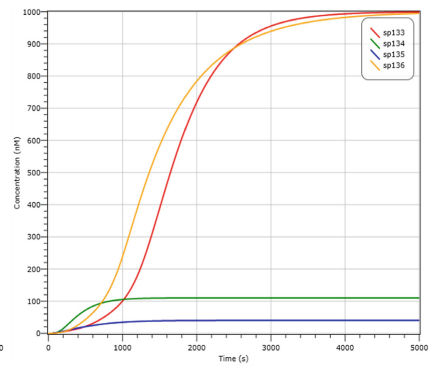
(a) $C_1C_0AB=0000, L=0, M=0$



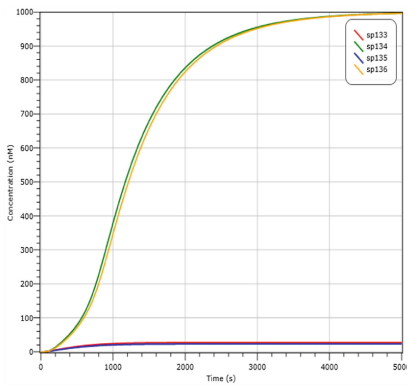
(b) $C_1C_0AB=0001, L=1, M=0$



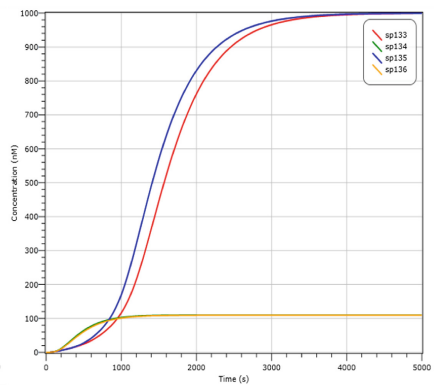
(c) $C_1C_0AB=0010, L=1, M=0$



(d) $C_1C_0AB=00101, L=0, M=1$

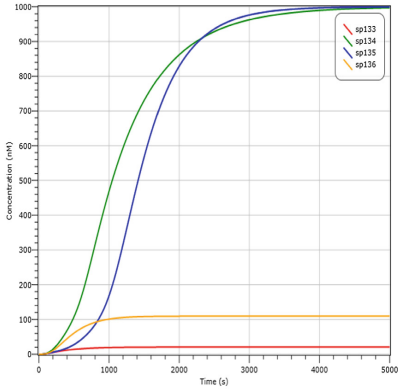


(e) $C_1C_0AB=0100, L=0, M=0$

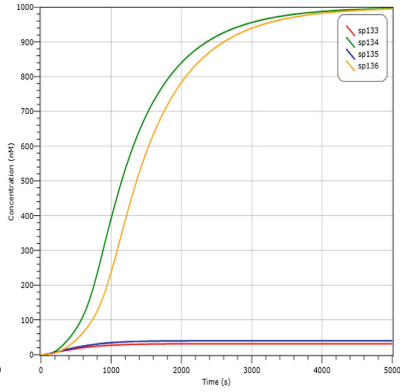


(f) $C_1C_0AB=0101, L=1, M=1$

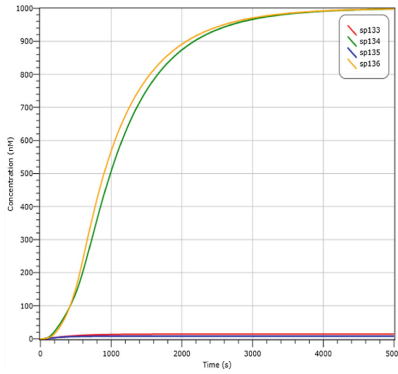
Fig. 6. The simulation results of the complex logical operations of one-bit binary number. (Color figure online)



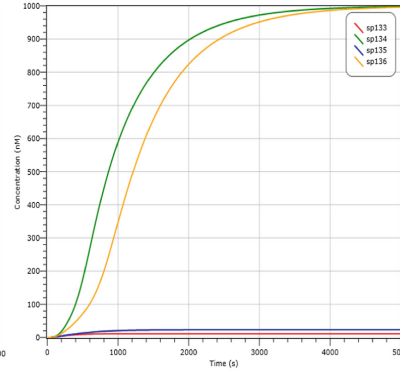
(g) $C_1C_0AB=0110, L=1, M=0$



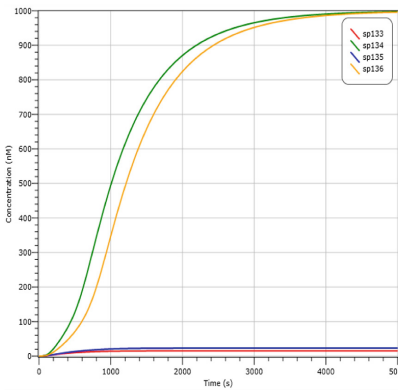
(h) $C_1C_0AB=0111, L=0, M=0$



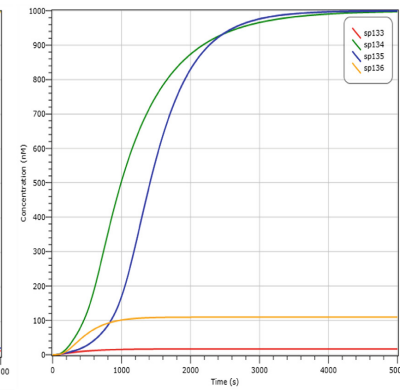
(i) $C_1C_0AB=1000, L=0, M=0$



(j) $C_1C_0AB=1001, L=0, M=0$

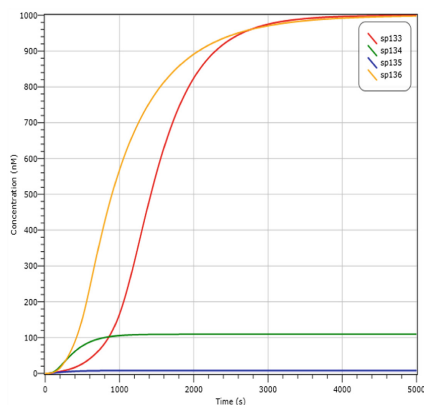


(k) $C_1C_0AB=1010, L=0, M=0$

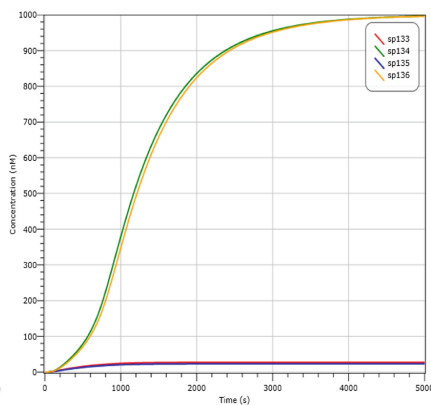


(l) $C_1C_0AB=1011, L=1, M=0$

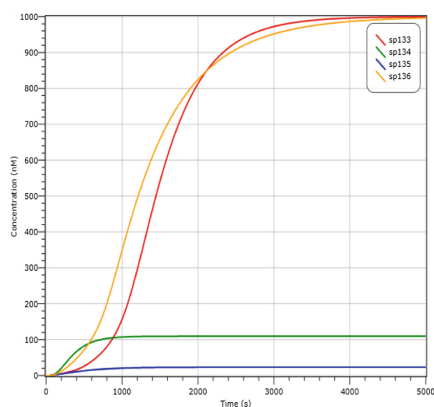
Fig. 6. (continued)



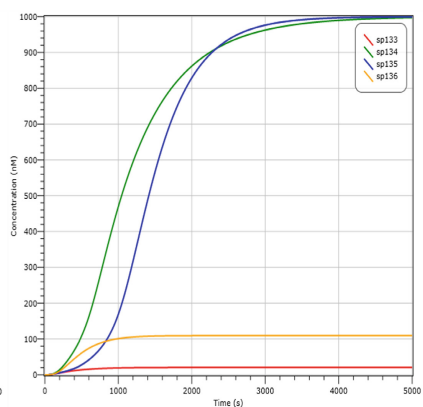
(m) $C_1C_0AB=1100, L=0, M=1$



(n) $C_1C_0AB=1101, L=0, M=0$



(o) $C_1C_0AB=1110, L=0, M=1$



(p) $C_1C_0AB=1111, L=1, M=0$

Fig. 6. (continued)

4 Simulation with Visual DSD

In the biochemical logic circuit, Visual DSD is a kind of professional simulation software, which can be achieved from programming to compiling, simulation and analysis. Here, the reaction process of the addition, subtraction, multiplication and division complex logical operations based on the displacement of DNA strands is investigated by using DSD simulation software. In the Visual DSD software, the addition, subtraction, multiplication and division complex logical operations circuit is constructed through compiling program code. After the code is compiled, the DNA specie is generated automatically by using the Visual DSD software. Network and detailed sequences of domains can be provided for the further convenient analysis. In

the end, the output results for the addition, subtraction, multiplication and division complex logical operations circuit can be obtained by performing the “Simulate” button. The simulation results of the addition, subtraction, multiplication and division complex logical operations can be shown correctly in Fig. 6(a–p).

In this article, these input signals are C_1 , C_0 , A and B , respectively. When the value of the inputs signals C_1^0 , C_1^1 , C_0^0 , C_0^1 and A^0 , A^1 , B^0 , B^1 are chosen, the different output results of L and M are gained in the complex logical operations of one-bit binary number. The values of output results are represented by four lines, in which, the green curve is L , the red curve is M . The response time is 5000 s, the total input concentration is 1000 nM. If the ultimate concentration of L_i ($i = 0, 1$) ranges from 0 to 200 nM, then the value of L is logic “0”. If the final concentration of L_i ($i = 0, 1$) is changed between 800 nM and 1000 nM, then the value of L is logic “1”. If the ultimate concentration of M_i ($i = 0, 1$) ranges from 0 to 200 nM, then the value of M is logic “0”. If the final concentration of M_i ($i = 0, 1$) is changed between 800 nM and 1000 nM, then the value of M is logic “1”. In the plots of the design of the addition, subtraction, multiplication and division complex logical operations of one-bit binary number logic circuit based on DNA strand displacement, the input signals are C_1^0 , C_1^1 , C_0^0 , C_0^1 and A^0 , A^1 , B^0 , B^1 , the output signals are L and M .

In order to determine whether there is a result of the design of the addition, subtraction, multiplication and division complex logical operations, the result can be determined by changing the curve density of the binary input values. If the final concentration of output curve L_1 is from 900 to 1000 nM, then there is a output result produced, if the final concentration of output curve L_1 is from 0 to 200 nM, then there is no output result produced. Otherwise, If the final concentration of output curve M_1 is from 900 to 1000 nM, then there is carry or borrow operations produced, if the final concentration of output curve M_1 is from 0 to 200 nM, then there is no carry or borrow operations here.

The output signals L_1^0 , L_1^1 , M_1^0 and M_1^1 are obtained through the chosen value of inputs the binary numbers A , B , C_0 and C_1 . It is shown that the simulation results of the addition, subtraction, multiplication and division complex logical operations corresponding to all combinations of input signals are obtained from 0000 to 1111 in Fig. 6 (a–p), respectively. When the input signal $C_1C_0 = 00$, the logic circuit performs an addition logic operation. When $C_1C_0 = 01$, the logic circuit performs a subtraction logic operation. When $C_1C_0 = 10$, the logic circuit performs multiplication logic operation, when $C_1C_0 = 11$, the logic circuit performs the division multiplication logic operation, and the output signals are represented by L and M . When $L = 1$, it indicates that the output result is generated, and when $L = 0$, it indicates that there is no output result; if When the output signal $M = 1$, it means that there is a borrowing operation when performing the operation, otherwise there is no borrowing operation. According to the logic value of the input signal AB , the value of the output result is also different, so that when performing complex logic circuit operations, a large number of logic circuit operations are involved.

When $C_1C_0 = 00$, the addition operation is performed, if the input operation signal $AB = 00$, the output value of L is 0, and the value of M is 0, indicating that there is no output signal generated and there is no carry operation, the simulation result is shown

in Fig. 6(a); if the input operation signal $AB = 01$, the output value of L is 1, and the value of M is 0, indicating that there is output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(b); if the input operation signal $AB = 10$, the output value of L is 1, and the value of M is 0, indicating that there is output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(c); if the input operation signal $AB = 11$, the output value of L is 0, and the value of M is 1, indicating that there is no output signal generated and there is carry operation, the simulation result is shown in Fig. 6(d);

When $C_1C_0 = 01$, the subtraction operation is performed, if the input operation signal $AB = 00$, the output value of L is 0, and the value of M is 0, indicating that there is no output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(e); if the input operation signal $AB = 01$, the output value of L is 1, and the value of M is 1, indicating that there is output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(f); if the input operation signal $AB = 10$, the output value of L is 1, and the value of M is 0, indicating that there is output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(g); if the input operation signal $AB = 11$, the output value of L is 0, and the value of M is 0, indicating that there is no output signal generated and there is carry operation, the simulation result is shown in Fig. 6(h);

When $C_1C_0 = 10$, the multiplication operation is performed, if the input operation signal $AB = 00$, the output value of L is 0, and the value of M is 0, indicating that there is no output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(i); if the input operation signal $AB = 01$, the output value of L is 0, and the value of M is 1, indicating that there is output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(j); if the input operation signal $AB = 10$, the output value of L is 0, and the value of M is 0, indicating that there is output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(k); if the input operation signal $AB = 11$, the output value of L is 1, and the value of M is 0, indicating that there is no output signal generated and there is carry operation, the simulation result is shown in Fig. 6(l);

When $C_1C_0 = 11$, the division operation is performed, if the input operation signal $AB = 00$, the output value of L is 0, and the value of M is 1, indicating that there is no output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(m); if the input operation signal $AB = 01$, the output value of L is 0, and the value of M is 0, indicating that there is output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(n); if the input operation signal $AB = 10$, the output value of L is 0, and the value of M is 1, indicating that there is output signal generated and there is no carry operation, the simulation result is shown in Fig. 6(o); if the input operation signal $AB = 11$, the output value of L is 1, and the value of M is 0, indicating that there is no output signal generated and there is carry operation, the simulation result is shown in Fig. 6(p);

5 Conclusion

In this paper, the structure of the reaction mechanism model of DNA strand displacement has been constructed in the design of the addition, subtraction, multiplication and division complex logical operations firstly. Secondly, the logical circuit model of the design of the addition, subtraction, multiplication and division complex logical operations based on DNA strand displacement has been designed and implemented through the dual-rail circuits. Finally, the reaction process of DNA strand displacement has been simulated and the results of the logical operations can be displayed correctly through the specialized Visual DSD software. According to the results of the simulation, the displacement of DNA strands is an effective method for logic computation.

This investigation for the addition, subtraction, multiplication and division complex logical operations circuit based on DNA strand displacement by dual-rail circuits designed may have a great prospect for the development and application in the biological information processing, molecular computing, and so on. Due to the limited experimental conditions, the experiments of DNA strand compound displacement will be the future research directions.

Acknowledgment. The work is supported by the State Key Program of National Natural Science of China (Grant No. 61632002), the National Natural Science of China (Grant Nos. 61472372, 61572446, 61603348, 61602424), Science and Technology Innovation Talents Henan Province (Grant No. 174200510012), Research Program of Henan Province (Grant Nos. 15IRTSTHN012, 162300410220, 17A120005).

References

1. Qian, L., Winfree, E.: Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**(6034), 1196–1201 (2011)
2. Winfree, E.: DNA computing by self-assembly. *Bridge* **33**(4), 31–38 (2003)
3. Cardelli, L.: Two-domain DNA strand displacement. *Math. Struct. Comput. Sci.* **23**(2), 247–271 (2013)
4. Adleman, L.: Molecular computation of solutions to combinatorial problems. *Science* **266**(5187), 1021–1024 (1994)
5. Bui, H., Garg, S., Miao, V., Song, T., Mokhtar, R., Reif, J.: Design and analysis of linear cascade DNA hybridization chain reactions using DNA hairpins. *New J. Phys.* **19**(1), 015006 (2017)
6. Chen, X.: Expanding the rule set of DNA circuitry with associative toehold activation. *J. Am. Chem. Soc.* **134**(1), 263–271 (2011)
7. Bartlett, E., Brissett, N., Plocinski, P., Carlberg, T., Doherty, A.: Molecular basis for DNA strand displacement by NHEJ repair polymerases. *Nucleic Acids Res.* **44**(5), 2173–2186 (2016)
8. Li, F., Tang, Y., Traynor, S., Li, X., Le, C.: Kinetics of proximity-induced intramolecular DNA strand displacement. *Anal. Chem.* **88**(16), 8152–8157 (2016)
9. Qian, L., Winfree, E.: A simple DNA gate motif for synthesizing large-scale circuits. In: Goel, A., Simmel, F.C., Sosik, P. (eds.) *DNA 2008*. LNCS, vol. 5347, pp. 70–89. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03076-5_7

10. Chen, Y., Dalchau, N., Srinivas, N., Phillips, A., et al.: Programmable chemical controllers made from DNA. *Nat. Nanotechnol.* **8**(10), 755–762 (2013)
11. Wang, Y., Sun, J., Zhang, X., Cui, G.: Full adder and full subtractor operations by DNA self-assembly. *Adv. Sci. Lett.* **4**(2), 383–390 (2011)
12. Yang, D., Tan, Z., Mi, Y., Wei, B.: DNA nanostructures constructed with multi-stranded motifs. *Nucleic Acids Res.* **45**(6), 3606–3611 (2017)
13. Song, T., Garg, S., Mokhtar, R., Bui, H., Reif, J.: Analog computation by DNA strand displacement circuits. *ACS Synth. Biol.* **5**(8), 898–912 (2016)
14. Yang, X., Tang, Y., Traynor, S.M., Li, F.: Regulation of DNA strand displacement using an allosteric DNA toehold. *J. Am. Chem. Soc.* **138**(42), 14076–14082 (2016)
15. Zhang, X., Ying, N., Shen, C., Cui, G.: Fluorescence resonance energy transfer-based photonic circuits using single-stranded tile self-assembly and DNA strand displacement. *J. Nanosci. Nanotechnol.* **17**(2), 1053–1060 (2017)
16. Zhang, X., Zhang, W., Zhao, T., Wang, Y., Cui, G.: Design of logic circuits based on combinatorial displacement of DNA strands. *J. Comput. Theor. Nanosci.* **12**(7), 1161–1164 (2015)
17. Lakin, M., Stefanovic, D.: Supervised learning in adaptive DNA strand displacement networks. *ACS Synth. Biol.* **5**(8), 885–897 (2016)
18. Qian, L., Winfree, E., Bruck, J.: Neural network computation with DNA strand displacement cascades. *Nature* **475**(7356), 368 (2011)
19. Wang, Z., Wu, Y., Tian, G., Wang, Y., Cui, G.: The application research on multi-digit logic operation based on DNA strand displacement. *J. Comput. Theor. Nanosci.* **12**(7), 1252–1257 (2015)
20. Li, W., Yang, Y., Yan, H., Liu, Y.: Three-input majority logic gate and multiple input logic circuit based on DNA strand displacement. *Nano Lett.* **13**(6), 2980–2988 (2013)
21. Carlse, K., Jakobsen, C., Kallelose, T., Paerregaard, A., et al.: F-calprotectin and blood markers correlate to quality of life in pediatric inflammatory bowel disease. *J. Pediatr. Gastroenterol. Nutr.* **65**(5), 539–545 (2017)
22. Zhong, W., Tang, W., Fan, J., Zhang, J., Zhou, X., Liu, Y.: A domain-based DNA circuit for smart single-nucleotide variant identification. *Chem. Commun.* **54**(11), 1311–1314 (2018)
23. Wang, Z., Tian, G., Wang, Y., Wang, Y., Cui, G.: Multi-digit logic operation using DNA strand displacement. In: Pan, L., Păun, G., Pérez-Jiménez, Mario J., Song, T. (eds.) *BIC-TA 2014*. CCIS, vol. 472, pp. 463–467. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-45049-9_75
24. Chen, Z., Liu, Y., Xin, C., Zhao, J., Liu, S.: A cascade autocatalytic strand displacement amplification and hybridization chain reaction event for label-free and ultrasensitive electrochemical nucleic acid biosensing. *Biosens. Bioelectron.* **113**, 1–8 (2018)
25. Fern, J., Schulman, R.: Design and characterization of DNA strand-displacement circuits in serum-supplemented cell media. *ACS Synth. Biol.* **6**(9), 1774–1783 (2017)



An Improved GMM-Based Moving Object Detection Method Under Sudden Illumination Change

Jian Cheng, Yusen Gang, Shuai Bai, Yi-nan Guo^(✉),
and Dongwei Wang

China University of Mining and Technology, Xuzhou 221116, Jiangsu, China
guoyinan@cumt.edu.cn

Abstract. In traditional GMM-based moving object detection method, the foreground was segmented by comparing the current frame with the constructed Gaussian distributions, which might occur the detection failure under sudden illumination change because the Gaussian distributions were disobeyed in current frame. Therefore, an improved GMM-based moving object detection method under sudden illumination change is proposed in terms of the fact that the pixel intensity in the neighboring zone changes with the similar degree as illumination suddenly changes. The mean quadratic deviation of the gray values in four different directions are employed to build four independent Gaussian distributions instead of the pixel intensity. And a pixel is considered to be the foreground as all of mean quadratic deviations disobey the historical Gaussian distributions. Simulation results show that the proposed method can more exactly detect the foreground with less false pixels as illumination changes abruptly. Moreover, constructing the Gaussian model parallel with four threads reduces the total computation time, which meets the need of the video processing in practical engineering.

Keywords: Mean quadratic deviation · Gaussian distribution
Moving object detection · Sudden illumination change

1 Introduction

Moving object detection, an essential issue in the human-activity analysis, gesture recognition and video surveillance, is to recognize the physical movement of an objective in a given region. The detection accuracy has a direct impact on the behavior analysis and understanding of the objectives. Studies on the detecting moving objects generally have done from the following three aspects. Background subtraction method (BS) modeled a reference frame as background [3–6], and then detected the moving objects from the difference between the current frame and background [7, 8]. Under sudden illumination change, segmenting the moving objects based on the static background is difficult due to the pixel intensity of video varying significantly. Frame differencing method (FD) found the moving targets by comparing two successive frames [1]. The foreground can be detected timely with slowly-changed illumination, but segmented as the whole image under sudden illumination change. Optical Flow

algorithm (OF) [2] normally failed in challenging the environments with illumination suddenly changed due to the violation of the brightness constraint.

The moving object detection algorithms under sudden illumination change have attracted more attention in recent years. The background was re-initialized once the lighting condition changed [9, 10], whereas setting a suitable threshold to abandon the old background and detecting moving objects timely during the re-initialization was difficult. Assuming that illumination uniformly changed in the scene, the median of changed gray values over all pixels was used as the illumination scaling factor to update the background [11]. However, the hypotheses about the uniformly-changed pixel intensity was not supported in practice. Different with above linear mapping, the current frame was segmented by calculating a nonlinear mapping from each region [12]. Vosters [13, 14] employed eigen background to handle local illumination change. A modeling method for illumination-sensitive background was proposed [15], and performed well under illumination slowly changed but failed in sudden illumination change. Mahmoudpour [16] presented a global illumination compensation approach based on two background models with different adaption rates. Heikkil [17] proposed a texture-based method for modeling the background by the local binary pattern (LBP) instead of the pixel intensity. It avoided the detection failure from sudden illumination change, but the influence of high-order pixels on LBP was much more significant than low-order pixels.

Gaussian mixture model (GMM) [7] as a novel background subtraction method, had been successfully applied in moving object detection due to its advantages in modeling the background with several Gaussian distributions and segmenting the foreground by comparing the current frame with these Gaussian distributions. Though the parameters of Gaussian distributions were updated in time, sudden illumination change might deteriorate its performances, even failed in detecting the targets because the Gaussian distributions were disobeyed in current frame. Therefore, an improved GMM-based moving object detection method under sudden illumination change is proposed. Given the fact that the gradient of pixel intensity is less affected by the lighting condition, the background is modeled in terms of the mean quadratic deviation of the gray values in four neighborhood directions instead of pixel intensity.

In the following part of this paper, the principle of GMM-based moving object detection method is introduced at first. In Sect. 3, the proposed algorithm is presented and analyzed in detail. The adaptability and robustness of the algorithms to illumination change are analyzed for two videos in Sect. 4. At last, the highlights and future works are given.

2 GMM-Based Moving Object Detection

GMM, the extension of single Gaussian model, describes the probability of pixel intensity for each pixel in the image by the weighted sum of several Gaussian distributions [7], which has better robustness and adaptability for the object detection with complex background. Denote $g_{i,t}$ as the pixel intensity of i -pixel at time t , $\eta(\mu_{ij,t}, \sigma_{ij,t})$ as the j -th Gaussian distribution of $g_{i,t}$ at time t and $w_{j,t}$ as its weigh, $\mu_{ij,t}$ and $\sigma_{ij,t}$ as the mean and standard deviation, respectively, then the Gaussian mixture model is formulated as follows.

$$p(g_{i,t}) = \sum_{j=1}^K w_{j,t} \eta(\mu_{ij,t}, \sigma_{ij,t}) \quad (1)$$

$$\eta(\mu_{ij,t}, \sigma_{ij,t}) = \frac{1}{(2\pi)^{1/2} \sigma_{ij,t}} \exp\left(-\frac{(g_{i,t} - \mu_{ij,t})^2}{2\sigma_{ij,t}^2}\right) \quad (2)$$

After initializing the mean and standard deviation of GMM, each pixel of the current frame is matched with GMM in terms of the 1-norm distance between $x_{i,t}$ and $\mu_{ij,t-1}$, denoted as $d_{ij,t} = |g_{i,t} - \mu_{ij,t-1}|$. Let δ be the matching threshold. The pixel is considered as the background if the following matching condition is satisfied. Once no Gaussian distribution accords with the pixel, it is called the foreground.

$$\frac{d_{ij,t}}{\sigma_{ij,t-1}} < \delta \quad (3)$$

Denote α as the learning rate, $\text{sgn}()$ as sign function, the weight of each Gaussian distribution is updated after detecting the foreground. With larger α , the background is updated faster, but GMM may be not convergence, even lead to ghost detection.

$$w_{ij,t} = (1 - \alpha)w_{ij,t-1} + \alpha \max\left\{0, \text{sgn}\left(\delta - \frac{d_{ij,t}}{\sigma_{ij,t-1}}\right)\right\} \quad (4)$$

The mean and standard deviation of a Gaussian distribution failing to match a pixel remain unchanged. Otherwise, they are updated as follows.

$$\mu_{ij,t} = (1 - \rho)\mu_{ij,t-1} + \rho g_{i,t} \quad (5)$$

$$\sigma_{ij,t}^2 = (1 - \rho)\sigma_{ij,t-1}^2 + \rho(g_{i,t} - \mu_{ij,t})^2 \quad (6)$$

where $\rho = \alpha/w_{j,t}$ is the update rate. For any pixel having no matching Gaussian distribution, a new model assigning a large standard deviation and a small weight is initialized instead of the Gaussian distribution with the least $w_{j,t}/\sigma_{ij,t}$. The mean of the new Gaussian distribution is set to the pixel intensity and the parameters of the rest models keep constant.

Given the fact that more Gaussian distributions contained in GMM, its adaptability to the environment is better, but the computation complexity and time-consuming both become more [18, 19]. Hence, the number of the Gaussian models usually is set to $K \in [3, 5]$. Though GMM has the satisfied detection accuracy by updating the parameters for a new frame with illumination gradually changed, most of pixels will be misjudged as the foreground with sudden illumination change.

3 GMM-Based Moving Object Detection Method Under Sudden Illumination Change

With suddenly changed illumination, the pixel intensity of the current frame has the significant difference from its historical frames. Especially, the pixel intensity for the neighboring pixels lying in a small zone is uniformly changed and the difference among these pixels is much smaller. Based on this, an improved GMM-based moving object detection method under sudden illumination change is proposed. The mean quadratic deviation of the gray values in four different directions are employed to build four independent Gaussian distributions instead of the pixel intensity. And a pixel is considered to be the foreground as all of mean quadratic deviations disobey the historical Gaussian distributions.

Suppose that the neighboring zone consists of 5×5 pixels around i -th pixel. Denote x_i and y_i as the location of i -th pixel, the mean quadratic deviations of pixel intensity in four directions along $135^\circ, 45^\circ, 90^\circ, 0^\circ$ as shown in Fig. 1(a) are calculated as follows:

$$\overline{\sigma_{i,t}^{135^\circ}} = \sqrt{\sum_{k=-2}^1 (g_{i,t}(x_i+k, y_i-k) - g_{i,t}(x_i+k+1, y_i-k-1))^2} \quad (7)$$

$$\overline{\sigma_{i,t}^{45^\circ}} = \sqrt{\sum_{k=-2}^1 (g_{i,t}(x_i+k, y_i+k) - g_{i,t}(x_i+k+1, y_i+k+1))^2} \quad (8)$$

$$\overline{\sigma_{i,t}^{90^\circ}} = \sqrt{\sum_{k=-2}^1 (g_{i,t}(x_i, y_i+k) - g_{i,t}(x_i, y_i+k+1))^2} \quad (9)$$

$$\overline{\sigma_{i,t}^{0^\circ}} = \sqrt{\sum_{k=-2}^1 (g_{i,t}(x_i+k, y_i) - g_{i,t}(x_i+k+1, y_i))^2} \quad (10)$$

Taking the normalized image shown in Fig. 1(b) as example, the mean quadratic deviations of pixel intensity in four directions reflect the characteristics of the image.

Given the fact that above mean quadratic deviations roughly obey Gaussian distribution in image sequence, four Gaussian models are independently constructed for the mean quadratic deviations in four directions.

$$f\left(\overline{\sigma_{i,t}^k}\right) = \frac{1}{\sqrt{2\pi}\sigma_{ik,t}} \exp\left(-\frac{\overline{\sigma_{i,t}^k} - \mu_{ik,t}}{2\sigma_{ik,t}^2}\right), k = 0^\circ, 90^\circ, 45^\circ, 135^\circ \quad (11)$$

The initial mean of Gaussian distributions denoted as $\mu_{ik,t}$ are assigned as the mean quadratic deviation of the first frame. For each arrived frame, the mean quadratic

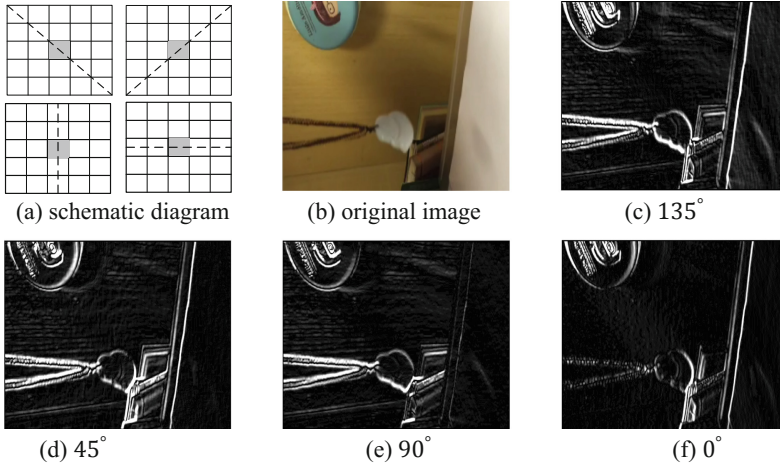


Fig. 1. The mean quadratic deviations of pixel intensity in four directions

deviations in four directions are calculated, and then matched with Gaussian distributions at time $t - 1$.

$$d_{ik,t} = \left| \overline{\sigma_{i,t}^k} - \mu_{ik,t-1} \right| \tag{12}$$

If $d_{ik,t} > \delta \times \sigma_{ik,t-1}$ is satisfied, the mean quadratic deviation in k -th direction is changed in current frame. And i -th pixel is detected as foreground as the mean quadratic deviations in four direction all disobey the Gaussian distribution at time $t - 1$. δ is the matching threshold. Too smaller δ results in the ghost detection, whereas the moving objects cannot be found in time under larger δ . After detecting the foreground, the parameters of GMM are updated as follows.

$$\mu_{ik,t} = (1 - \rho)\mu_{ik,t-1} + \rho\overline{\sigma_{i,t}^k} \tag{13}$$

$$\sigma_{ik,t}^2 = (1 - \rho)\sigma_{ik,t-1}^2 + \rho(\overline{\sigma_{i,t}^k} - \mu_{ik,t})^2 \tag{14}$$

where $\rho = \alpha d_{ik,t} / \sigma_{ik,t-1}$ is the update rate of parameter. The greater the mean quadratic deviation disobeys the Gaussian distribution, the parameters are updated faster and speed up the convergence.

The proposed GMM-based moving object detection method can avoid amplifying the noise. Assuming that a noise pixel is labeled by the black point in Fig. 2(a). Taking the integrated mean quadratic deviation of four directions in the neighboring zone as criterion to detecting foreground, 16 pixels in gray may be misjudged as the foreground due to the effect of noise. Hence, the independent GMM along a direction is modeled to minimizing the pixels influenced by the noise, and a pixel is judged as the foreground

only when the mean quadratic deviation in four directions all changed, which improving the detection accuracy.

In addition, the moving objects can be positioned more accurately by the proposed method. As shown in Fig. 2(b), any pixel in the neighboring zone labeled in gray may be misjudged as the target due to the integrated mean quadratic deviation of four directions. Taking the Gaussian distribution independently formed in each direction as the judgment condition, a pixel is not considered as the foreground when the directions with the changed mean quadratic deviation are less than four, as shown in Fig. 2(c). This shrinks the zone possibly misjudged and improves the positioning accuracy.

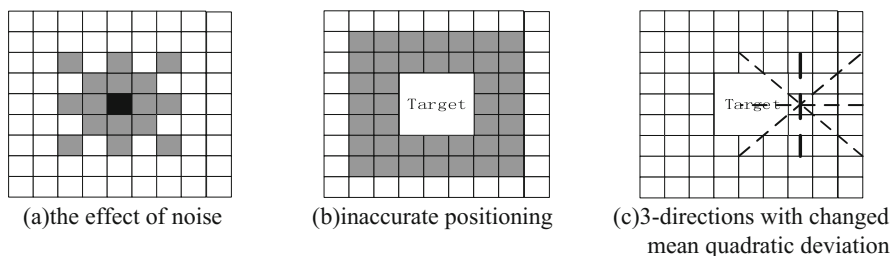


Fig. 2. The advantages of mean quadratic deviation

4 Experiments and Analysis

All experiments are done on MATLAB with i5-2450 M CPU and 4G memory. A testing set, PetsD2TeC2 (<http://www.multitel.be/cantata/>), is employed to verified the performances of the proposed method. In this experiment, the initial mean and standard deviation are set to 0 and 6, respectively. Taking the former 100 frames in PetsD2TeC2 as the image sequence, the mean of Gaussian distribution and the update rate for the pixel locating in (238,48) are shown in Figs. 3 and 4. The current frame disobeys the initial Gaussian distribution due to its mean is set to 0 and the corresponding mean quadratic deviation along 0° is 16, which results in the larger update rate. The mean of Gaussian distribution, subsequently, converges to the actual value quickly and the learning rate becomes smaller.

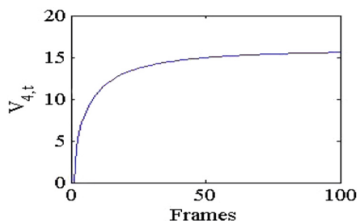


Fig. 3. The mean of Gaussian distribution

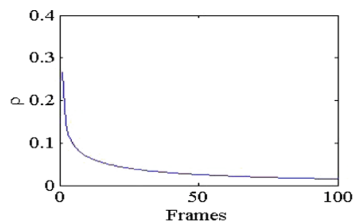


Fig. 4. The update rate

Figure 5 depicts the foreground in the former 16 frames detected by the proposed method. At the beginning, almost all texture of image is detected as foreground due to the initial mean setting to 0. Along with updating the parameters, more and more pixels can be matched with Gaussian distribution and judged as background. In 16-th frame, all pixels are correctly classified to background. Therefore, the parameters can be updated quickly by the proposed algorithm and the frames for learning are normally set to 30.

In order to verify the adaptability of the proposed method under sudden illumination change, two different videos are employed to compare the performances among FD, GMM and LBP. In the first video, a jade swings in the indoor scene. There exist global illumination change caused by turning on the lamp and local illumination change formed by jade blocking light. Three successive frames including 776-th, 777-th and 778-th frames, are extracted and global illumination is suddenly changed in 778-th frame. In the second video, PetsD2TeC2, there are the moving persons and cars with illumination slowly changed.

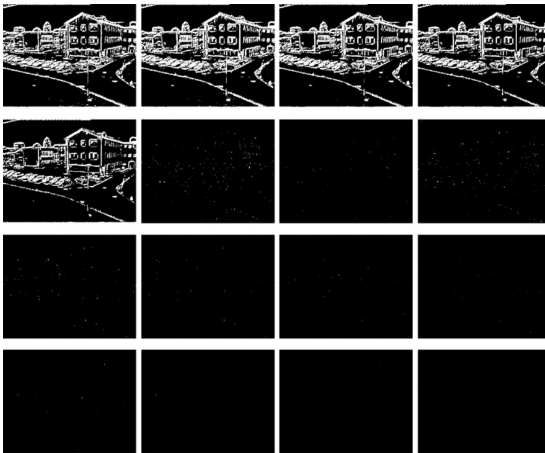


Fig. 5. Detecting the foreground for the former 16 frames

As shown in Figs. 6 and 7, FD and GMM can perform well under the condition of stable or slowly changed illumination, but fail in the situation with sudden illumination change. LBP can avoid the detection failure under sudden illumination change, but form many false directions in the dense region of image. The proposed method can more exactly detect the foreground with less false pixels than LBP as illumination changes abruptly. Though the mean quadratic deviations in four directions employed in proposed method guarantee the adaptability to illumination change, only the texture of moving object is considered and the information on the pixels with similar gray values is lost. Based on this, only the outline of the moving objects can be found for the image with less texture.

Compared the time-consuming among FD, GMM, LPB and the proposed method shown in Table 1 indicates that the proposed method spends more computation time

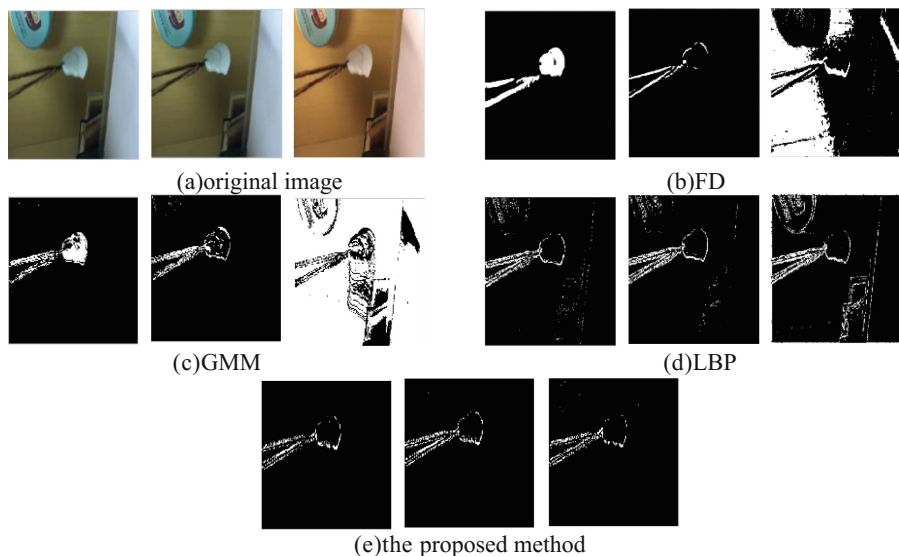


Fig. 6. Jade

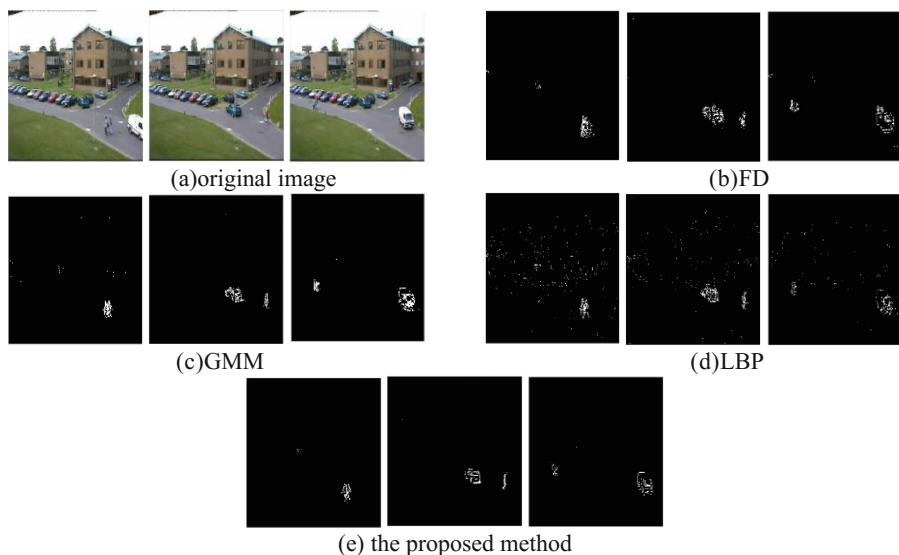


Fig. 7. PetsD2TeC2

than GMM and FD, but is faster than LBP. However, the Gaussian distributions of GMM are interrelated, while the Gaussian distributions in four directions of the proposed algorithm are completely independent. So four parallel threads are employed to

construct the Gaussian model in four direction and provide the information to the main thread for forming the foreground image, which lowers time-consuming of the proposed algorithm. Compared the time-consuming of the proposed method realized by C++ with 1 thread or 4 threads shows that the parallel computation shrinks the total computation time and meets the need of the video processing in practical engineering.

Table 1. Comparison of the time-consuming among different algorithms

The testing set	Image size	FD	GMM	LBP	Proposed method (Matlab)	Proposed method (C++, 1 thread)	Proposed method (C++, 4 threads)
Jade	320 × 568	0.019 s	1.312 s	3.837 s	2.981 s	57 ms	24 ms
PetsD2TeC2	336 × 448	0.021 s	1.128 s	2.865 s	2.482 s	49 ms	20 ms

5 Conclusion

Under sudden illumination change, the pixel intensity in the neighboring zone changes with the similar degree. Based on this, an improved GMM-based moving object detection method under sudden illumination change is proposed. The mean quadratic deviations of the gray values in four different directions are employed to build four independent Gaussian distributions instead of the pixel intensity. And a pixel is considered to be the foreground as all of standard deviations disobey the historical Gaussian distributions. Simulation results show that the proposed method can more exactly detect the foreground with less false pixels and has the better adaptability as illumination changes abruptly. Moreover, constructing the Gaussian model with four threads parallel reduces the total computation time and meets the need of the video processing in practical engineering. Improving the integrity of segmenting the image in the proposed algorithm is our next work.

Acknowledgement. This work was supported by the National Natural Science Foundation of China (No. 61573361), the National Key Research and Development Program (No. 2016 YFC0801406).

References

1. Jun-Qin, W.: An adaptive frame difference method for human tracking. *Adv. Inf. Sci. Serv. Sci.* **4**(1), 381–387 (2012)
2. Barron, J.L., Fleet, D.J., Beauchemin, S.S.: Performance of optical flow techniques. *Int. J. Comput. Vis.* **12**(1), 43–77 (1994)
3. Huang, Z.K., Chau, K.W.: A new image thresholding method based on Gaussian mixture model. *Appl. Math. Comput.* **205**(2), 899–907 (2008)
4. Shental, N., Bar-Hillel, A., Hertz, T., et al.: Computing Gaussian mixture models with EM using equivalence constraints. *Adv. Neural. Inf. Process. Syst.* **16**(8), 465–472 (2004)

5. Zivkovic, Z.: Improved adaptive Gaussian mixture model for background subtraction. In: Proceedings of the 17th IEEE International Conference on Pattern Recognition, pp. 28–31. IEEE Computer Society, Cambridge (2004)
6. Lee, D.S.: Effective Gaussian mixture learning for video background subtraction. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(5), 827–832 (2005)
7. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: Proceedings of the 12th IEEE Conference on Computer Vision and Pattern Recognition, pp. 246–252. IEEE Computer Society, Colorado (1999)
8. Chen, Z., Ellis, T.: A self-adaptive Gaussian mixture model. *Comput. Vis. Image Underst.* **22**(5), 35–46 (2014)
9. Jenifa, R.A.T., Akila, C., Kavitha, V.: Rapid background subtraction from video sequences. *Int. J. Comput. Sci. Eng.* **4**(3), 1077–1086 (2012)
10. Barnich, O., Droogenbroeck, M.V.: ViBe: a universal background subtraction algorithm for video sequences. *IEEE Trans. Image Process.* **20**(6), 1709–1724 (2011)
11. Chen, Z., Ellis, T.: Self-adaptive Gaussian mixture model for urban traffic monitoring system. In: IEEE International Conference on Computer Vision Workshops, pp. 1769–1776. IEEE (2011)
12. Paruchuri, J.K., Sathiyamoorthy, E.P., Cheung, S.C.S., et al.: Spatially adaptive illumination modeling for background subtraction. In: IEEE International Conference on Computer Vision Workshops, pp. 1745–1752. IEEE Computer Society (2011)
13. Vosters, L.P.J., Shan, C., Gritti, T.: Background subtraction under sudden illumination changes. In: IEEE International Conference on Advanced Video and Signal Based Surveillance, pp. 384–391. IEEE Computer Society (2010)
14. Vosters, L., Shan, C., Gritti, T.: Real-time robust background subtraction under rapidly changing illumination conditions. *Image Vis. Comput.* **30**(12), 1004–1015 (2012)
15. Cheng, F.C., Huang, S.C., Ruan, S.J.: Illumination-sensitive background modeling approach for accurate moving object detection. *IEEE Trans. Broadcast.* **57**(4), 794–801 (2011)
16. Mahmoudpour, S., Kim, M.: Robust foreground detection in sudden illumination change. *Electron. Lett.* **52**(6), 441–443 (2016)
17. Heikkil, M., Pietikinen, M.: A texture-based method for modeling the background and detecting moving objects. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(4), 657–662 (2006)
18. Yu, J.: A nonlinear kernel Gaussian mixture model based inferential monitoring approach for fault detection and diagnosis of chemical processes. *Chem. Eng. Sci.* **68**(1), 506–519 (2012)
19. Povey, D., Burget, L., Agarwal, M., et al.: The subspace Gaussian mixture model-A structured model for speech recognition. *Comput. Speech Lang.* **25**(2), 404–439 (2011)



A Method of Accurately Accepting Tasks for New Workers Incorporating with Capacities and Competition Intensities

Dunwei Gong¹, Chao Peng^{2(✉)}, Xinchao Zhao³, and Qiuzhen Lin⁴

¹ China University of Mining and Technology, Xuzhou 221116, Jiangsu, China

² Xuzhou Vocational Technology Academy of Finance and Economics,
Xuzhou 221116, Jiangsu, China
Chaopeng_PC@163.com

³ Beijing University of Posts and Telecommunications, Beijing 100876, China

⁴ Shenzhen University, Shenzhen 518060, China

Abstract. Crowdsourcing has been one of focuses from scholars and businessmen along with rapid development and wide-spread applications of Internet, and platforms utilizing crowdsourcing have been popular among netizens. Among these platforms, the lack of a new worker's capacity of accepting tasks seriously affects his/her incomes obtained by fulfilling tasks issued by requesters, which reduces his/her enthusiasm for crowdsourcing. We propose a method of accurately accepting tasks for new workers incorporating with capacities and competition intensities in this paper. In the proposed method, various types of competitions among workers are taken into consideration through information of similar tasks and workers. Then, the competitive intensity between a new worker and each of other competitors is calculated by utilizing the time consumption in fulfilling a task. Finally, the strategy of accepting tasks is generated by solving the formulated optimization problem using a genetic algorithm, with the purpose of improving the accuracy of accepting tasks. We evaluate the proposed method with data provided by Taskcn and ZBJ, the two representative commercial crowdsourcing platforms in China, and compare the results of different methods with the actual incomes. The experimental results show that the proposed method can improve the accuracy of accepting tasks, which is beneficial to increasing the labor efficiency and incomes for new workers.

Keywords: Crowdsourcing · New worker · Accurate acceptance of tasks
Competition intensity · Genetic algorithm

1 Introduction

Along with rapid development and wide-spread applications of Internet, crowdsourcing has been one of focuses from scholars and businessmen, and has become a manner of efficiently solving problems raised by netizens or companies through open platforms of Internet [1].

Crowdsourcing has been successfully applied in many fields, among which some representative ones are as follows. (1) Databases [2, 3]. In the field of databases,

researchers have mainly adopted crowdsourcing to increase (reduce) the scale of effective (unvalued) data, and managed databases with massive information. (2) Natural language processing [4, 5]. Due to the diversity of natural languages, a plenty of problems in communication or translation can hardly be solved by computers. Therefore, researchers have utilized data from the public to identify or correct language errors. (3) Information retrieval [6, 7]. Along this line, studies focus mainly on the quality and the security of data in information retrieval, given the fact that there generally exist various frauds in network transactions. To guarantee the reliability of information in transactions, researchers have improved the quality by collecting evaluations and feedbacks from the public and experts. (4) Software testing [8, 9]. In the community of software testing, researchers have employed crowdsourcing to recruit professionals to solve problems with long periods and time consumptions, which provide a new way to software testing. Nowadays, more and more tasks have been fulfilled through crowdsourcing platforms due to a growing number of requesters and workers being involved in these platforms.

Up to present, previous studies of accurately accepting tasks have focused mainly on recommendation for previous workers by utilizing the preferences of workers and (or) the information of tasks. However, rare work is involved with the problem of accepting tasks for new workers. The lack of information on new workers poses a big problem for research. At present, the researches mainly contain content - based and collaborative filtering methods, and also mine group information to improve the accuracy of recommendation. To this end, we present a method of accurately accepting tasks for new workers incorporating with capacities and competition intensities. In the proposed method, the problem of accepting tasks is first formulated as a constraint optimization problem with an unknown parameter representing the time consumption of a new worker in fulfilling a task. Then, the competition intensity is calculated based on information provided by workers and tasks. Finally, the strategy of accepting tasks is generated by solving the optimization problem using a genetic algorithm.

The existing work does not consider the influence of the relationship between network users on accuracy. Based on the existing work, this paper explores the group relationship to improve the accuracy of accepting tasks. In addition, the existing work of space tasks and traditional tasks are studied separately, but workers are likely to select these tasks at the same time. How to unify different types of tasks is the innovation of this paper. This paper has the following three-fold contributions: (1) establishing an integrated environment with traditional and spatial tasks (2) defining two types of competition relations, and presenting a method of calculating the competition intensity of a new worker using information provided by workers and tasks, and (3) evaluating the proposed method by a series of experiments with data provided by two typical crowdsourcing platforms.

The remainder of this paper is organized as follows. Section 1 reviews the related work. In Sect. 2, the optimization problem of accepting tasks is formulated and the calculation of the competition intensity is described in detail. The strategy of accepting tasks is generated using a genetic algorithm in Sect. 3. Section 4 evaluates the proposed method. Finally, Sect. 5 concludes the whole paper and points out topics to be investigated in the future.

2 Method of Accurately Accepting Tasks

2.1 Formulation of the Problem of Accepting Tasks

The optimization problem of accepting tasks can be described as follows. A worker selects and fulfills tasks in a crowdsourcing platform to obtain rewards as much as possible. When formulating the problem, we divide tasks into different levels according to their rewards since different tasks have different remunerations. The level of a task is denoted as $i = 1, 2, \dots, I$, with m_i the amount of tasks in the i -th level. Besides, the flag of whether or not the worker selects and fulfills the j -th task in the i -th level is denoted as x_{ij} , and if the worker selects and fulfills the task, $x_{ij} = 1$; otherwise, $x_{ij} = 0$. When $x_{ij} = 1$, b_{ij} is denoted as the reward of the corresponding task gained by the worker. Then, $F(x)$ is the total rewards gained from the platform with the following expression:

$$\begin{aligned}
 \max \quad & F(x) = \sum_{i=1}^I \sum_{j=1}^{m_i} b_{ij} c_{ij} x_{ij} \\
 \text{s.t.} \quad & \sum_{i=1}^I \sum_{j=1}^{m_i} t_{ij} x_{ij} \leq T \\
 & \sum_{i=1}^I \sum_{j=1}^{m_i} x_{ij} \leq M
 \end{aligned} \tag{1}$$

It is worth noting that the competition intensity of the j -th task in the i -th level is denoted as c_{ij} , T and M are the maximal time and the maximal number of tasks, respectively.

The competition intensity, c_{ij} , is generally unknown. To this end, how to calculate the value of c_{ij} based on previous information becomes very important, which will be given in the next section.

2.2 Calculation of the Competition Intensity

As more and more netizens participate in crowdsourcing platforms, a task is often selected by a number of workers. On this circumstance, the competition from the workers will become more and more obvious [10]. Based on the studies on previous crowdsourcing platforms, we divide all the competition relations into the following two categories: direct competition (DC) and potential competition (PC). Literally, DC refers to a situation that a new worker competes for the same task with a number of workers. In crowdsourcing platforms, a task generally contains information associated with the worker who selected and fulfilled it. Based on it, we can find workers who have a direct competition relation with the new worker. In contrast, PC represents a possible competition relation between a new worker and others that does not exist at present, but may appear in the future.

Idea. Based on the preliminary research on the accuracy of accepting tasks for a new worker, his/her recommended task set including information about accepting tasks can be obtained. In addition, there are many published tasks and regular workers associated with the new worker on a crowdsourcing platform. If these data are fully utilized, the

competition intensity that reflects the relation between the new worker and each of regular workers will be calculated. Figure 1 depicts the process of calculating the value of c_{ij} .

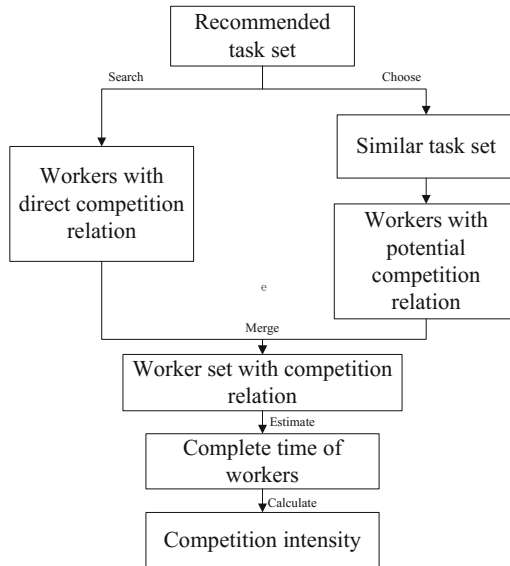


Fig. 1. The process of calculating the value of c_{ij}

Seeking Competitors. The set of workers who have a direct competition relation with the new worker is denoted as W_D , which can be obtained by traversing information of the recommended task set. The set of workers who compete with the new worker potentially is denoted as W_I . The strategy of seeking W_I is presented in detail as follows.

The emergence of mobile devices, such as smartphones, iPads, and laptops, has greatly met the demands of tasks in terms of time and location. In crowdsourcing platforms, workers close to the location of tasks are generally easy to be rewarded, indicating that the geographical location is one of important factors that have an influence on the accuracy in accepting tasks. In many commercial crowdsourcing platforms, traditional tasks, however, cannot be ignored when more and more spatial tasks appear. Simultaneously considering two types of tasks is beneficial to solving real-world problems related to crowdsourcing. As a result, we combine traditional and spatial tasks to improve the authenticity of accepting tasks in this paper.

In previous studies, traditional tasks can be described by the following attributes: the expected completion time, the number of rewards, and the domain, which are represented with t , b , and d , respectively [11]. Since spatial tasks have an additional attribute, the geographic location, the geographic coordinates are added to represent the location of a task, denoted as $P(x, y)$. To have a unified representation of different types of tasks, we employ δ to denote the weight of an attribute, and if a task contains an

attribute, $\delta=1$; otherwise, $\delta=0$. Furthermore, T_a refers to the recommended task set, and T_β represents tasks in the platform, $T_\alpha = (\delta_1 p_\alpha, \delta_2 b_\alpha, \delta_3 t_\alpha, \delta_4 d_\alpha)$, $T_\beta = (\delta_1 p_\beta, \delta_2 b_\beta, \delta_3 t_\beta, \delta_4 d_\beta)$. Then, the similarity between T_a and T_β , denoted as $D(T_a, T_\beta)$, can be formulated after they are normalized with the following expression:

$$D(T_a, T_\beta) = \frac{1}{\sqrt{\sum_{h=1}^4 \delta_h}} \sqrt{\delta_1 (p_{ij}(e_1, e_2) - p_l(e_1, e_2))^2 + \delta_2 (b_{ij} - b_l)^2 + \delta_3 (t_{ij} - t_l)^2 + \delta_4 (d_{ij} - d_l)^2} \quad (2)$$

Let D^0 be a threshold of the distance between tasks, and $0 \leq D^0 \leq 1$. If T_β satisfies that $D(T_a, T_\beta) \leq D^0$, it will be similar with T_a . Let $S(T_a)$ be a set consisting of all the similar tasks, then $S(T_a) = \{T_\beta | D(T_a, T_\beta) \leq D^0\}$. Based on data in $S(T_a)$, the set of workers who selected and (or) fulfilled tasks in $S(T_a)$ can be sought, denoted as W_I .

The same workers is possible to be sought in both sets, W_D and W_I . Therefore, we merge the two sets into a big one, denoted as W .

Estimation of the Time Consumption in Completing a Task. In a crowdsourcing platform, information associated with workers in W is generally known. For workers who selected tasks in the recommended set, we can directly get their time consumption in fulfilling tasks. For other workers, we use the previous estimation method [11] to get the time consumption.

Let $S_k(T_a)$ be the set of similar tasks completed by the k -th worker in W , and t_β^k be the time consumption of u^k in completing T_β . Then, the estimation of the time consumption, t_α^k , of u^k in completing T_α , denoted as t_α^k , can be given as follows:

$$t_\alpha^k = \frac{1}{\sum S_k(T_a)} \sum_{T_\alpha \in S_k(T_a)} (1 - D(T_{ij}, T_{ij}) \cdot t_\beta^k) \quad (3)$$

The set of the estimated time consumption in fulfilling T_α for each worker in W is denoted as $t_\alpha = \{t_\alpha^k | k \in W\}$.

Calculation of the Competition Intensity. Different workers generally have different competition intensities, and the worker with a stronger competition is more likely to obtain the reward of a task than others. For a new worker, the competition strength of the worker with the strongest competition is the most valuable. In this way, the competition intensity of the new worker in fulfilling a task can be achieved by comparing his/her time consumption with the shortest one.

Let t_{ij}^{\min} be the minimal estimation of the time consumption for the j -th task in the i -th level completed by workers in W , then t_{ij}^{\min} can be calculated as follows:

$$t_{ij}^{\min} = \min\{t_{ij}\} \quad (4)$$

Based on the minimal estimation, a time-dependent function is constructed to calculate the competition intensity between the new worker and others. Let t_α^{new} be the

new worker's estimation of the time consumption in fulfilling the j -th task in the i -th level. Then, the competition intensity of, c_{ij} , is calculated as follows:

$$c_{ij} = \frac{t_{ij}^{min}}{t_a^{new}} \tag{5}$$

From formula (5), the new worker is more likely to obtain the reward when he/she is more competitive than others, that is, $c_{ij} > 1$.

When some tasks in the recommended task set are deleted due to the new worker's low competition intensity, other tasks in the current platform will be added to the set so as to satisfy constraints from the working hours and the number of tasks. Following that, the new worker will go on choosing tasks.

The steps of the proposed method of integrating capacity aware and competitive intensity for new workers in accepting tasks accurately are provided as follows.

- Step 1: Set the values of the parameters, s^0 .
- Step 2: Seek the set of the workers competed with the new worker.
- Step 3: Obtain the estimation time consumption of the workers sought in step 2 with the formula (3).
- Step 4: Calculate the intensity of competition between the new worker and other workers under the task in the recommended set by using formula (5). If $c_{ij} > 1$, it is reserved; otherwise, it is removed.
- Step 5: Calculate the intensity of competition under the other tasks available in the platform, and add the task which is $c_{ij} > 1$ to the recommended set until the set is full.

3 Problem Solving Using a Genetic Algorithm

Since genetic algorithms are a kind of efficient methods of solving combinatorial optimization problems [12, 13], we employ them to solve the problem in this paper.

Fitness Function. There are constraints which can be transformed into a part of the fitness function by the method of the penalty function when evaluating an individual. We design the fitness function of an individual by incorporating the constraints into the objective function including the competition intensity. Then, the fitness function of an individual, denoted as $Fit(x)$ can be expressed as:

$$Fit(X) = \sum_{i=1}^I \sum_{j=1}^{m_i} c_{ij} b_{ij} x_{ij} - \gamma_1 \left(\sum_{i=1}^I \sum_{j=1}^{m_i} t_{ij} x_{ij} - T \right) - \gamma_2 \left(\sum_{i=1}^I \sum_{j=1}^{m_i} x_{ij} - M \right) \tag{6}$$

where γ_1 and γ_2 are two penalty factors reflecting the constraints related to the time consumption and the number of tasks, and their values have a close relation with the scales of the time consumption and the number of tasks.

In addition, the position and the probabilities of the crossover and mutation operators are adjusted according to the proportion of different levels of tasks in all tasks, which is helpful to guide the evolution and speed up convergence.

Genetic Operators. A genetic algorithm includes the following three genetic operators: selection, crossover, and mutation. The selection operator is employed to select individuals for conducting the crossover and mutation operators, and the roulette method is adopted in this study. For the crossover operator, it generates two offspring with the purpose of exploring superior individuals in the search space based on two selected ones, and the single-point crossover approach is used here. Regarding the mutation operator, it is utilized to generate better individuals for a selected one, and we employ the method of single-point crossover in this paper.

4 Experiments

4.1 Experimental Data

In this section, we employ data from Taskcn, a domestic commercial crowdsourcing platform, and also collect information from April 1, 2017 to July 15, 2017 in ZBJ [14] to reduce the experimental error resulted from the small amount of data. In addition, the actual incomes of these new workers earned from March 22, 2017 to September 3, 2017 are tracked to compare with those gained by the proposed method.

4.2 Experimental Results and Analysis

Parameter Settings. We investigate the influence of the values of γ_1, γ_2 on the genetic algorithm employed in this paper, which is reflected by the reward obtained by a new worker. Since a number of combinations of γ_1, γ_2 have huge amount of computation, the method of orthogonal design is employed to get the best combination of these optimal parameters. In addition, the optimal values of crossover and mutation probabilities, p_c and p_m , are set to 0.3 and 0.5, respectively, via a number of trials. We set T and M to 30 and 100, respectively, and change γ_1 and γ_2 in the range of [40, 70] and [5, 20], respectively, in the experiments. If the number of levels of a factor in orthogonal design increases, the number of tests required will geometrically raise. Therefore, four levels are determined in the range of each factor. What is more, an empty column of a factor is added to the orthogonal test to achieve a good result.

A scheme of orthogonal test is constructed by selecting an appropriate orthogonal table. Since each factor has four levels in Table 1, we choose the first three columns of the orthogonal table, $L_{16}(4^3)$. Therefore, the total number of tests is 16. The result of the orthogonal experiment is obtained by the following two steps. The optimal level of each factor is first determined by evaluating its rank in calculating the average of test indexes at different levels of each factor and the range of these averages. Then, the scheme of the optimal combination of these factors is achieved by combining the optimal levels of all the factors. The larger the range of a factor, the greater effect of the change in the factor is on the test index. Based on the above experiments, γ_1, γ_2 are set to 70, 15, respectively, in the subsequent experiments.

We employ the same value of the parameters in the previous method. The values of $\alpha, s^0, D^0, \gamma_1, \gamma_2$ are set to 0.9, 0.5, 0.7, 70, and 15, respectively, and applied to the subsequent experiments. The detailed experimental process is detailed in [11].

Evaluation of the Proposed Method. To illustrate the accuracy of the proposed method, the concept of precision is employed in this section. In an actual crowd-sourcing process, the ratio of the number of tasks paid off to the total number of selected tasks can well reflect the accuracy of accepting tasks. Let L_{num} be the number of tasks in the recommended task set, and R_{num} be the number of tasks paid off. Then, the precision of accepting tasks can be expressed as follows:

$$P = \frac{L_{num} \cap R_{num}}{L_{num}} \times 100\% \tag{7}$$

From formula (7), the larger the value of P , the more accuracy the method is.

To reduce the error resulted from an individual task or worker in the experiments, the average accuracy is employed, which is defined as follows:

$$P' = \frac{1}{n} \sum_{g=1}^n P \tag{8}$$

In the original data set, there are a number of workers that are randomly selected to obtain the value of P' . The proposed method is compared with the one that does not consider the competition intensity.

Table 1. Comparison between the values of P' obtained by different methods.

No.	Method P'	Value with competition intensity	Value without competition intensity	Difference
1		0.42	0.33	+0.09
2		0.56	0.43	+0.13
3		0.37	0.20	+0.17
4		0.34	0.33	+0.01
5		0.46	0.39	+0.07

It can be seen from Table 1 that in groups 1, 2, 3, and 5, the value of P' obtained by incorporating the competition intensity is more than 0.07 higher than that obtained without the competition intensity. Compared with the method without the competition intensity, this paper can improve the accuracy of accepting tasks. However, in group 4, the values of P' obtained by the two methods have a difference of 0.01 which is not

Table 2. Comparison between the values of P' obtained by different workers with different methods in group 4.

Worker	Method	Value with competition intensity	Value without competition intensity	Difference
	P			
1		0.41	0.36	+0.05
2		0.27	0.22	+0.05
3		0.21	0.21	0.00
4		0.23	0.26	-0.03
5		0.33	0.33	0.00
6		0.29	0.32	-0.03
7		0.36	0.41	-0.05
8		0.46	0.40	+0.06
9		0.26	0.21	+0.05
10		0.58	0.58	0.00

obvious, suggesting that the two methods have no significant difference in the precision. To analyze the results in group 4 definitely, the value of P gained by each worker is further calculated.

Table 2 lists the values of P obtained by different methods from each worker in group 4. From this table, the values of P from workers 1–3, 5, and 8–10 using the proposed method are larger than those without the competition intensity, and each difference is larger than or equal to 0.05, suggesting that the proposed method has a significant influence on improving the accuracy of accepting tasks. The values of P from workers, 4, 6, and 7 using the proposed method are not larger than those without the competition intensity, and the differences are -0.03 , -0.03 , and -0.05 , respectively, indicating that the accuracy of accepting tasks is significantly improved. The reasons for this can be provided as follows. In a crowdsourcing platform, there are generally a large number of competitors who are competitive under a task. Although a new worker fulfills the task, it is possible that he/she is fruitless in obtaining the corresponding remuneration. In order to avoid unprofitable behavior, the proposed method generally suggests the new worker giving up the task. If the new worker chooses the task, there will be some chances that he/she gets the reward associated with the task. In addition, it is possible that there are a number of potential competitors for the new worker with respect to a task. To avoid the case that the new worker and potential workers with a stronger competition simultaneously select the task, the

proposed method generally advise the new worker abandoning the task that potential workers may choose. In the subsequent process of accepting tasks, it is possible that these potential competitors will not select the task. As a result, the new worker will achieve a reduced reward.

If the competition in accepting tasks is ignored, the new worker may not be paid for a task when he/she encounters competitors with a strong competition. On this circumstance, a decision will be made for the new worker using the method incorporating with the competition intensity. Generally, the experimental results show that the new worker can effectively avoid unprofitable tasks using the proposed method, thus leading to a high accuracy of accepting tasks.

The following process is adopted to evaluate the superiority of the proposed method. We first randomly select a number of new workers with reward and record their rewards obtained in a period. Then, we gain the rewards of these new workers using the method without the competition intensity. Finally, we calculate the rewards by the proposed method, and evaluate the proposed method by comparing those rewards in different conditions. To fulfill this task, three groups of new workers are randomly selected from the same platform. The experimental results are provided as follows.

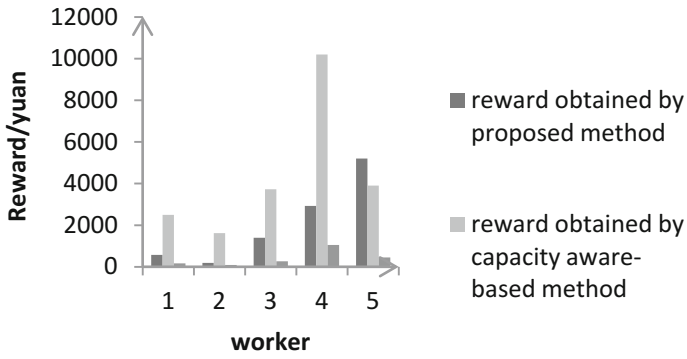
Figure 2 shows the reward of each new worker when utilizing different methods, where the abscissa represents the identifier of a new worker, and the ordinate means his/her reward.

Figure 2 shows that for groups A, B, and C, rewards obtained by the proposed method are clearly larger than actually ones, indicating that the method incorporating with the competition intensity can effectively improve the rewards for the new workers. The reason lies in that the new workers avoid unprofitable tasks during accepting tasks by utilizing information associated with tasks and workers.

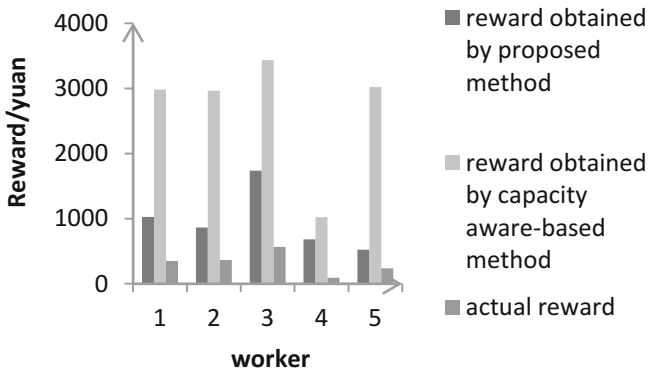
The difference between the reward obtained by the proposed method and the actual one is denoted as H_1 , and that between the reward obtained by the capacity aware-based method and the actual one is denoted as H_2 . Table 3 lists their values for group A.

Table 3 reports that the difference between the reward obtained by the proposed method and the actual one is smaller than that between the capacity aware-based method and the actual one. The reason lies in that the new worker selects a task in an environment without the competitive relation. There exist some tasks with a high reward theoretically, whose number, however, is very small in real-world applications. Furthermore, the reward obtained by the proposed method is higher than that achieved by the one without the competition intensity for the 5 new workers, showing that competitive tasks can be avoided based on the competition relation among workers, and replaced with others easy to get paid. In this way, the new worker will get as more rewards as possible in a period.

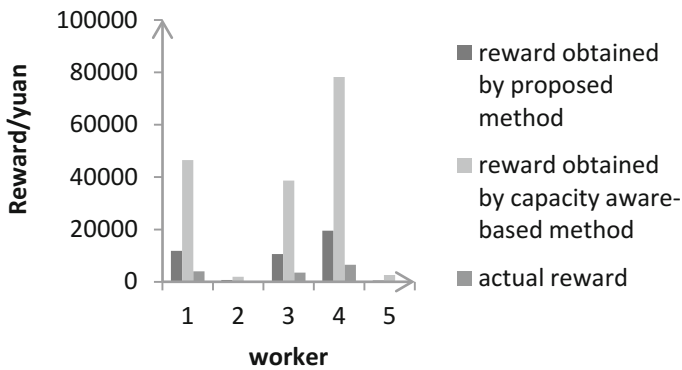
We can obtain the following conclusion through the experimental results and analysis: the proposed method can significantly improve the accuracy of accepting tasks for a new worker, which is beneficial to increasing the rewards for the new worker.



(A)



(B)



(C)

Fig. 2. The reward of each new worker obtained by different methods

Table 3. The difference between the reward obtained by different methods and the actual one.

Worker	H_1	H_2
1	406.928	2335.039
2	101.155	1531.375
3	1133.111	3459.969
4	1872.296	9145.914
5	4754.860	3449.000

5 Conclusion

We have studied the problem of accurately accepting tasks for new workers, and proposed a method of accurately accepting tasks incorporating with capacities and competition intensities in this paper. The experimental results show that the proposed method can well estimate the capacity of a new worker in accepting tasks, which is beneficial to increasing his/her incomes. The thresholds used in this paper are, however, subjectively set when estimating the capacity of a worker in accepting tasks. In addition, potential competitive workers may choose tasks unselected by the new worker in the subsequent process of accepting tasks. How to automatically adjust these thresholds based on data available and guarantee the stability of data are topics to be studied in the future.

References

1. Feng, J.H., Li, G.L., et al.: A survey on crowdsourcing. *Chin. J. Comput.* **38**(9), 1713–1726 (2015)
2. Li, G., Chai, C., Fan J., et al.: CDB: optimizing queries with crowd-based selections and joins. In: *Proceedings of the 2017 ACM International Conference on Management of Data*, pp. 1463–1478. ACM, New York (2017)
3. Li, G., Fan, J., et al.: Crowdsourced data management: overview and challenges. In: *Proceedings of the 2017 ACM International Conference on Management of Data*, pp. 1711–1716. ACM, New York (2017)
4. Loures, T.C., Vaz de Melo, P.O.S., Veloso, A.A.: Generating entity representation from online discussions: challenges and an evaluation framework. In: *Proceedings of the 23rd Brazillian Symposium on Multimedia and the Web*, pp. 197–204. ACM, New York (2017)
5. Guo, H., Özgür, K., Jeukeng, A.L., et al.: Toward extraction of security requirements from text: poster. In: *Proceedings of the 5th Annual Symposium and Bootcamp on Hot Topics in the Science of Security*, p. 27. ACM (2018)
6. Kim, Y., Collins-Thompson, K., Teevan, J.: Using the crowd to improve search result ranking and the search experience. *ACM Trans. Intell. Syst. Technol. (TIST)* **7**(4), 50 (2016)
7. Von, H.A., Bresler, A., Shuman, O., et al.: Bantuweb: a digital library for resource scarce south african languages. In: *Proceedings of the South African Institute of Computer Scientists and Information Technologists*, pp. 1–10. ACM (2017)
8. Abhinav, K., Dubey, A.: Predicting budget for Crowdsourced and freelance software development projects. In: *Proceedings of the 10th Innovations in Software Engineering Conference*, pp. 165–171. ACM (2017)

9. Dwarakanath, A., Chintala, U., Shrikanth, N.C., et al.: Crowd build: a methodology for enterprise software development using crowdsourcing. In: 2015 2nd International Workshop on CrowdSourcing in Software Engineering, pp. 8–14. ACM (2015)
10. Mridha, S.K., Bhattacharyya, M.: Network based mechanisms for competitive crowdsourcing. In: Proceedings of the ACM India Joint International Conference on Data Science and Management of Data, pp. 318–321. ACM (2018)
11. Gong, D., Peng, C.: A capacity aware-based method of accurately accepting tasks for new workers. In: Tan, Y., Takagi, H., Shi, Y., Niu, B. (eds.) ICSI 2017. LNCS, vol. 10386, pp. 475–480. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61833-3_50
12. Gong, D.W., Sun, J., Miao, Z.: A set-based genetic algorithm for interval many-objective optimization problems. *IEEE Trans. Evol. Comput.* **22**(1), 47–60 (2016)
13. Gong, D.W., Sun, J., Ji, X.: Evolutionary algorithms with preference polyhedron for interval multi-objective optimization problems. *Inf. Sci.* **233**(2), 141–161 (2013)
14. ZBJ Homepage. <https://www.zbj.com/>. Accessed 3 Sept 2017



Iteration-Related Various Learning Particle Swarm Optimization for Quay Crane Scheduling Problem

Mingzhu Yu¹, Xuwen Cong¹, Ben Niu^{2(✉)}, and Rong Qu³

¹ Department of Transportation Engineering, College of Civil Engineering, Shenzhen University, Shenzhen 518060, China

² College of Management, Shenzhen University, Shenzhen 518060, China
drniu@szu.edu.cn

³ School of Computer Science, University of Nottingham, Nottingham NG8 1BB, UK

Abstract. Quay crane scheduling is critical in reducing operation costs at container terminals. Designing a schedule to handling containers in an efficient order can be difficult. For this problem which is proved NP-hard, heuristic algorithms are effective to obtain preferable solutions within limited computational time. When solving discrete optimization problems, particles are very susceptible to local optimum in Standard Particle Swarm Optimization (SPSO). To overcome this shortage, this paper proposes an iteration-related various learning particle swarm optimization (IVLPSO). This algorithm employs effective mechanisms devised to obtain satisfactory quay crane operating schedule efficiently. Superior solutions can save up to 5 h for handling a batch of containers, thus significantly reduces costs for terminals. Numerical studies show that the proposed algorithm outperforms state-of-the-art existing algorithms. A series of experimental results demonstrate that IVLPSO performs quite well on obtaining satisfactory Pareto set with quick convergence.

Keywords: Improved particle swarm optimization
Quay crane scheduling problem · Various learning mechanism

1 Introduction

Quay crane scheduling is a crucial operation in container terminal management. When a vessel arrives at the container terminal, quay cranes are assigned to unload containers from it. The quay crane scheduling problem consists of quay cranes, containers and bays. Since a vessel contains several bays, to assign which quay crane to which bay in an order can be a complicated problem. The quay crane scheduling problem has been studied by scholars for years. One of the widely studied methods in quay crane scheduling include genetic algorithm (GA) [1] or modified GA. Some researches add random elements in crossover and mutation [2], or utilize biased mechanism [3] and modified GA combined with simulation [4].

Particle swarm optimization (PSO) was proposed by Kennedy [5] and Eberhart [6], and has been utilized in bunches of researches over many subjects including the quay

crane scheduling problems. PSO is a heuristic algorithm simulating a population of birds foraging to food source continuously. Each individual has a foraging position and updates its flight direction based on individual experience and group experience. To overcome the issue that PSO may be trapped into local optimal solution when addressing large-scale problems, scholars introduced mechanisms into improved versions of PSO. These include dynamic adaptive parameters like hierarchical swarm [7, 8]; inertia weight [9] and learning coefficients [10], learning from leaders [8, 11] or other individuals [7, 12] in a hierarchical system to update particles' velocity and position. To balance the exploitation and exploration of PSO, some scholars introduced un-certain elements to increase the diversity of the population by learning from random individuals [8].

In this paper, to address the issue of particles stuck into the local optimum and to accelerate convergence, an iterations-related variable learning particle swarm optimization (IVLPSO) is proposed, integrating new mechanisms learning from the center position and other near-optimal individuals. The idea of center position was proposed by Niu [7], and showed to accelerate convergence. Dynamic inertia weight is also introduced in IVLPSO. Some opposition solutions of high quality solutions (proposed by Ghasemi [12]) are also taken into consideration, (i.e., the number of a dimension is 2, and its opposition solution is $-2in$ [12], it will be adapted in the proposed algorithm). So as to increase the diversity of IVLPSO. All mechanisms we use aim to explore feasible and better solutions in a short time. Compared with standard particle swarm optimization in addressing quay crane scheduling problems [13], IVLPSO shows to perform better than PSO and GA.

This paper is organized as follows: Sect. 2 presents the mathematical formulation of the quay crane scheduling problem, and Sect. 3 describes the process of the IVLPSO algorithm. Experimental results are presented in Sect. 4. Section 5 concludes the paper.

2 Quay Crane Scheduling Problem Formulation

Li [13] proposed a quay crane scheduling problem with the objective of unloading containers from vessel bays with unequal amount of containers as soon as possible, namely, minimizing the total working time. It was assumed that the safety distance between two quay cranes should be no less than two-vessel-bay length. The problem model for the quay crane scheduling problem and the variables are presented in Table 1. The objective function and constrains are presented as follows.

$$\begin{aligned} \text{Minimize:} & \quad T \\ \text{Subject to:} & \quad F_b - S_b \geq 0, \forall 1 \leq b \leq B \end{aligned} \quad (1)$$

$$\sum_{q=1}^Q P_{qbt} \leq 1, \forall 1 \leq b \leq B, \forall 1 \leq t \leq T \quad (2)$$

$$\sum_{q=1}^Q X_{qbt} \leq 1, \forall 1 \leq b \leq B, \forall 1 \leq t \leq T \quad (3)$$

Table 1. Variables and definitions of Quay Crane Scheduling model

Variables	Definitions
Q	The total amount of quay cranes
B	The total amount of vessel bays of one vessel
T	The time when the last container is unloaded from the vessel
b	The index of bays, $b \in B$
q	The index of quay cranes, $q \in Q$
t	The index of time, $t \in T$
S_b	The time when the container of vessel bay b starts to be unloaded
F_b	The time when all containers of vessel bay b have been unloaded
X_{qbt}	$X_{qbt}=1$: quay crane q is working at vessel bay b at time t $X_{qbt}=0$: otherwise
P_{qbt}	$P_{qbt}=1$: quay crane q stays on vessel bay b at time t $P_{qbt}=0$: otherwise

$$\sum_{b=1}^B P_{qbt} = 1, \forall 1 \leq q \leq Q, \forall 1 \leq t \leq T \quad (4)$$

$$\sum_{b=1}^B X_{qbt} \leq 1, \forall 1 \leq q \leq Q, \forall 1 \leq t \leq T \quad (5)$$

$$\sum_{b=1}^B q' \cdot P_{qbt} - \sum_{b=1}^B q \cdot P_{qbt} > 2, \forall 1 \leq q < q' \leq Q, \forall 1 \leq t \leq T \quad (6)$$

$$T = \max_b F_b, \forall 1 \leq b \leq B \quad (7)$$

$$F_b = \max_{t,q} tX_{qbt}, \forall 1 \leq b \leq B \quad (8)$$

$$X_{qbt} = \begin{cases} 1 & \forall 1 \leq q \leq Q, \forall 1 \leq b \leq B, 1 \leq t \leq T \\ 0 & \end{cases} \quad (9)$$

$$P_{qbt} = \begin{cases} 1 & \forall 1 \leq q \leq Q, \forall 1 \leq b \leq B, 1 \leq t \leq T \\ 0 & \end{cases} \quad (10)$$

The objective is to minimize the total working time T . Constraints (1) set the start working time and finish working time of a vessel bay. Constraints (2) represent that there is no more than one quay crane staying in a bay at any time. Constraints (3) restrict no more than one quay crane working in a bay at any time. Constraints (4) illustrate that each quay crane must stay at one bay in any time. Constraints (5) mean that every quay crane can work for only one bay at any time. Constraints (6) ensure quay cranes will not cross each other and keep two bays length safety distance.

Constraints (8) show the relation vessel between a decision variable and finish working time of a vessel bay. Constraints (9) and (10) restrict the domain of the decision variables.

3 IVLPSO for the Quay Crane Scheduling Problem

3.1 Iteration-Related Various Learning Particle Swarm Optimization

The standard PSO simulates bird swarm foraging phenomenon, where each particle acts as a bird to update its velocity and position by its own experience and interaction with the others with parameters called inertia weight and learning coefficients. For more information, please refer to [5]. As one of the classical meta-heuristic algorithms, PSO has been applied widely in optimization problems. However, it suffers from being trapped to local Pareto fronts when tackling complex or large-scale problems. To overcome these drawbacks, IVLPSO is proposed in this paper. It is inspired by the diversified learning strategy used in [7] and [12]. The encoding, operations, and algorithm procedures of the proposed IVLPSO are described as follows.

Encoding. Each particle of IVLPSO is regarded as a potential feasible solution for the quay crane scheduling problem. The dimensions of a particle present the unique vessel bay numbers. The dimensions in a particle are integers presenting the working order of the quay crane, and each quay crane is available only if it has accomplished the previous unloading task. Some constraints of the quay crane scheduling problem can be met using this encoding; (A quay crane can only correspond to one vessel bay at any time, and ensures that each vessel bay’s mission is completed, showing a job order). Other constraints will be met through a series of operations based on the positions of the quay crane and the target bay. These operations ensure that each particle is satisfying all constraints. An example of the encoding for a particle is illustrated in Fig. 1, where the working sequence of the quay crane is from vessel bay 8, 4, 12 ... to 15.

8	4	12	3	19	17	1	5	14	11	10	20	9	16	7	13	18	2	6	15
---	---	----	---	----	----	---	---	----	----	----	----	---	----	---	----	----	---	---	----

Fig. 1. An example of the encoding for a particle

Three Key Operations of IVLPSO. In order to improve the standard PSO to avoid being trapped into local optimum, additional operations are introduced in IVLPSO, described as follows.

Dynamic Inertia Weight. Empirically, value of inertia weight w is set to be always in the range of [0.4, 0.9]. When w is approaching the minimum, PSO has a good performance in exploitation; when w is closer to the maximum, PSO does well in exploration. According to the characteristics of PSO, which is well known on quick convergence, w is set to decrease along with the iterations in IVLPSO. The probability of getting a better solution decreases a lot when the number of iterations exceeds 100, and the dynamic w is thus formulated as in Eq. (11):

$$w = 0.2 + 0.5 \cdot \frac{4 \cdot e^{10} \cdot m}{5M(e^{10} + 1)} \quad (11)$$

where M is the maximum iteration number, m is the present iteration and the value of w is set to between 0.2 and 0.7. The range of M is usually between 100 and 1000, in order to reduce the excessive influence of iterations on the gradient of inertia weight change, e^a in the denominator should be much larger than $5M$. e^{10} can be regarded as far greater than $5M$.

Various Learning Mechanisms. Three mechanisms in Eqs. (13), (14) and (15) are devised as learning exemplar for each particle. These mechanisms broaden the scope of learning and increase the diversity of the updated particles. The change of particle convergence is automatically judged at the late iteration to determine which mechanism is used to update the particles. If the value of the fitness function remains unchanged for dozens of consecutive times, the particle will be updated with Eq. (15), for other ways see the following equations (Eqs. (13) and (14)). The center position of a swarm could lead particles to towards better solutions efficiently. Particles select exemplar to update themselves within the global best position ($gbest$) and center position randomly. Exemplar of each particle is decided by generating a $PC(i)$ in Eq. (12), which was proposed by Niu et al. [7], and the velocity of particles is update using Eq. (13).

$$PC(i) = 0.05 + 0.45 \cdot \frac{e^{\frac{10(i-1)}{M-1}}}{e^{10} - 1} \quad (12)$$

If $R(i) < PC(i)$:

$$v(i+1) = w \cdot v(i) + c_1 \cdot rand \cdot (P_c - x(i)) + c_2 \cdot rand \cdot (P_g - x(i)) \quad (13)$$

If $R(i) \geq PC(i)$:

$$v(i+1) = w \cdot v(i) + c_1 \cdot rand \cdot (P_b - x(i)) + c_2 \cdot rand \cdot (P_g - x(i))$$

where i is the index of the particle, $R(i)$ is a random decimal in the range $[0, 1]$; $PC(i)$ is the learning probability of particle i . $v(i)$ and $x(i)$ are the velocity and position of particle i , respectively. P_c is the center position of the swarm and P_b is $pbest$, and P_g is $gbest$. If $R(i)$ is greater than $PC(i)$, particle i learns from $gbest$ and its personal best ($pbest$); otherwise, particle i regards the center position and $gbest$ as its exemplar.

It is also observed that $gbest$ convergences quite quickly in standard PSO, and the curve of solution quality becomes smooth and steady when iteration number reaches about one fifth of the maximum iteration number. The rest of iterations thus could benefit from some randomness in finding better solutions. Two new strategies are proposed in IVLPSO to increase the diversity of the swarm. First, as long as the iteration number reaches one fifth of the maximum iteration number, exemplars become some near-optimal particles (e.g. the half of the total individuals ranked the top in terms of fitness). This strategy is formulated as below in (14).

$$\begin{aligned}
 &\text{If iteration} > M/5 \text{ and } R(i) < PC(i) : \\
 &\quad v(i+1) = w \cdot v(i) + c_1 \cdot rand \cdot (P_r - x(i)) + c_2 \cdot rand \cdot (P_g - x(i)) \\
 &\text{If } R(i) \geq PC(i) : \\
 &\quad v(i+1) = w \cdot v(i) + c_1 \cdot rand \cdot (P_b - x(i)) + c_2 \cdot rand \cdot (P_g - x(i))
 \end{aligned} \tag{14}$$

where P_r is a random individual among near-optimal individuals of the swarm.

When the iteration number reaches one fifth of the total iterations, $R(i)$ is less than $PC(i)$ and $gbest$ is a constant over many iterations, then the new exemplar is opposition solution (take the opposite value of each number in every dimension. Sometimes the opposite value can be negative in some papers, it equals the maximum minus the current value in this paper). The new exemplar among the newest $pbest$ and three more particles according to the fitness of their opposition solutions, see Eq. (15).

$$\begin{aligned}
 P_f &= \max\{f(opbest), f(op_1), f(op_2), f(op_3)\} \\
 x(i) &= P_f
 \end{aligned} \tag{15}$$

where P_f is the new exemplar, $f(x)$ is the fitness of x ; $opbest$ is the opposition solution of $pbest$; $op_1op_2op_3$ are the opposition solutions of the best three solutions.

Boundary Restrictions. To make sure that solutions satisfy constraints defined in Sect. 2, and solutions on the boundary of each dimension (each dimension represents a number of vessel bay, so the lower boundary is 1 and the upper boundary is the number of total vessel bays. i.e., 20 vessel bays in total in [13], the upper bound is 20; 30 vessel bays in total, the upper bound is 30...) are not ignored, every dimension should take values within a lower bound and an upper bound after its velocity is updated. Boundary restrictions are made using Eq. (16).

$$\begin{aligned}
 &\text{If } x_i > x_{\max} : && x_i = x_{\max} \\
 &\text{If } x_i < x_{\min} : && x_i = x_{\min}
 \end{aligned} \tag{16}$$

where x_i is the position of the i th dimension. x_{\max} is the maximum of position number, which equals the amount of vessel bays. x_{\min} is the minimum of vessel bay number, which equals to 1.

Because every number in each dimension represents a number of a vessel bay, it can only be a positive integer. But it may become a decimal number after the update. So numbers of all dimensions are ranked and we get a new array that can restore to a large extent its integral parts (otherwise the update doesn't make sense); this operation aims to avoid repetition or non-integer after updating. The original corresponding sequencing number of the array needs to be recorded to obtain a similar solution with no redundancy.

3.2 The IVLPSO Algorithm for Quay Crane Scheduling Model

Based on mechanism of IVLPSO for terminal quay crane scheduling, the computational experiments are conducted in MATLAB environment. SPSO and GA are chosen as the comparing algorithms by using the same parameters and settings as IVLPSO. (Population size is 30 and iteration is 500). The pseudo-code of IVLPSO in solving quay crane scheduling problem is shown in Table 2.

Table 2. The pseudo-code of IVLPSO

```

Begin
Initialize parameters and produce the initial population of IVLPSO
( $c_1 = c_2 = 0.4, M = 500, D = 20, N = 30, v_{\min} = -2, v_{\max} = 2$ )
For ( $i = 1 : N$ ) //  $N$ : no. of iterations
    Calculate the fitness of each particle according to model in Section 3.1;
    Reserve  $pbest$  and  $gbest$ ;
End
For ( $t = 1 : M$ ) //  $M$ : population of particles
    For ( $j = 1 : D$ ) //  $D$ : dimensions of the particle
        Update the velocity of every dimension of particles using Eq. (13) and Eq. (14);
        Update position using Eq. (13) and Eq. (15);
        Do boundary restrictions by rules in Eq. (16);
    End
    Update fitness and reserve  $gbest$ ;
End
Output: value of  $gbest$  (shortest working time of quay cranes) and the solution
End

```

4 Experimental Results and Analysis

4.1 Experiment Parameter Settings

The parameter settings of the quay crane scheduling problem are the same as the PSO in [13], and the same scale of the problem, which include the amount of cargo of each vessel bay, the working efficiency of quay cranes and their travel time for the problem. More information and related problem data can be referred to [13].

In the instance of [13] (denoted as task(1)), there are in total 20 bays in the arriving vessel. The number of containers that require to be unloaded from each bay is as follows:

$$task(1) = (80, 168, 180, 66, 200, 180, 220, 60, 50, 140, 46, 210, 20, 90, 160, 110, 250, 50, 200, 160)$$

To show the efficiency of the proposed IVLPSO algorithm, other instances (task(2) - task(4)) have also been generated with different scale of bays and different number of containers. The total number of bays in these four instances varies from 20 to 40, and the average number of containers in the bays varies from 88 to 195.

$$\begin{aligned}
 task(2) &= (275, 47, 264, 266, 282, 140, 144, 250, 197, 38, 207, 232, 192, 73, 243, 89, 282, 280, 121, 278) \\
 task(3) &= \left(\begin{array}{l} 89, 108, 69, 120, 79, 123, 118, 170, 183, 83, 183, 99, 109, 145, \\ 10, 13, 81, 58, 35, 8, 49, 120, 21, 139, 48, 125, 61, 49, 44, 101 \end{array} \right) \\
 task(4) &= \left(\begin{array}{l} 265, 173, 189, 75, 139, 99, 243, 106, 88, 177, 69, 240, 24, 37, 47, \\ 198, 39, 27, 42, 145, 62, 204, 269, 101, 260, 257, 33, 144, 8, 140 \end{array} \right) \\
 task(5) &= \left(\begin{array}{l} 6, 223, 130, 72, 199, 220, 63, 42, 294, 56, 68, 159, 26, 97, 149, 114, 171, 116, 120, 17, \\ 255, 30, 182, 34, 108, 279, 236, 103, 8, 66, 58, 229, 33, 116, 281, 245, 158, 191, 37, 209 \end{array} \right)
 \end{aligned}$$

Working efficiency (containers per hour) of three quay cranes is normally distributed with $N(34.6, 2.69)$.

Tables show the results of the five instances with different combination of bays and containers using PSO, GA and IVLPSO on the same computer and same parameters (population size is 30 in three algorithms, learning factors are 0.4 and 0.4 in PSO and IVLPSO and cross probability is 0.8 and mutation probability is 0.2 in GA), respectively. Instance “20-2640” means the instance considers 20 bays and 2640 containers. These characteristics are set considering the operating data of Shekou terminal in Shenzhen of China in recent years. As shown in Tables 3, 4, 5, 6 and 7. For each instance the algorithms are run for 10 times to record the computational time, Pareto set of quay cranes total working time of each run, and also the average fitness. To make a fair comparison, the three algorithms are run with the same number of populations (i.e. 30) and same number of iterations (i.e. 500).

As shown in Tables 3, 5 and 7, IVLPSO produces better solutions in a shorter time than PSO and GA. Table 8 shows the solution of the best fitness 93124 for instance 1.

Table 3. The experimental results of the quay crane working time (seconds) for Instance 1.

Instance	Number	PSO	GA	IVLPSO
20-2640	1	103877	95163	94200
	2	102567	97893	93498
	3	98992	96427	94476
	4	102512	97925	95633
	5	104874	97664	97553
	6	97265	97571	93421
	7	103681	95163	95898
	8	107802	97893	97102
	9	100335	95573	94614
	10	98354	98268	93265
	Avg.	102025	96954	94966
Avg. running time		2394	4396	2339

Table 4. The experimental results of the quay crane working time (seconds) for Instance 2.

Instance	Number	PSO	GA	IVLPSO
20-3900	1	143605	138839	141638
	2	137475	138893	139439
	3	139491	137139	134333
	4	148816	140555	135589
	5	152829	140004	148456
	6	142762	144090	143680
	7	148462	140499	139001
	8	149269	140597	141145
	9	144116	139643	146840
	10	153120	141353	136568
	Avg.	145994	140161	140668
<i>Avg. running time</i>		3816	5149	2577

Table 5. The experimental results of the quay crane working time (seconds) for Instance 3.

Instance	Number	PSO	GA	IVLPSO
30-2640	1	110064	100829	102385
	2	102508	102712	95920
	3	109502	102465	98089
	4	108191	101685	97531
	5	106970	105857	99625
	6	105402	103365	99662
	7	111828	103844	99449
	8	103092	102272	96967
	9	107597	102920	105763
	10	101039	103198	100124
	Avg.	106619	102914	99551
<i>Avg. running time</i>		3401	3800	3349

Figure 2 presents the convergence of the three algorithms. Conclusions can be drawn as follows:

- IVLPSO performs the best among the three algorithms compared. More than half of the results from IVLPSO are better than the minimum of the other two algorithms. This demonstrates the effectiveness of IVLPSO. IVLPSO has showed superiority in quicker convergence without crossover or mutation in GA. Moreover, its diversity is no worse than GA. The convergence of PSO and IVLPSO are similar, except that the latter has a greater decrease in the first 50 iterations. The best solution can save about 2.7 h per vessel, which lead to potentially huge cost saving for port operators.
- In summary, IVLPSO provides a new approach for port operators to solve the quay crane scheduling problem in an effective and efficient way.

Table 6. The experimental results of the quay crane working time (seconds) for Instance 4.

Instance	Number	PSO	GA	IVLPSO
30-3900	1	148383	144977	142877
	2	157627	142357	150699
	3	152620	141767	148995
	4	163540	145282	140179
	5	143826	143005	149207
	6	158534	144288	147549
	7	154602	143572	142624
	8	155703	144393	145696
	9	159499	145712	149781
	10	153832	144245	145907
	Avg.	154816	143959	146351
<i>Avg. running time</i>		4906	4762	3819

Table 7. The experimental results of the quay crane working time (seconds) for Instance 5.

Instance	Number	PSO	GA	IVLPSO
40-5200	1	209790	209011	199736
	2	214227	201734	210672
	3	216649	208632	198676
	4	216565	205376	192594
	5	218036	200641	205711
	6	210105	204521	204584
	7	212526	212303	195200
	8	205123	207575	196767
	9	214251	209439	203684
	10	221760	210282	206894
	Avg.	213903	206951	201452
<i>Avg. running time</i>		4802	7281	5972

Notes: Instance “20-2640” means the task considers 20 bays and 2640 containers.

Table 8. Working schedule of the best solutions for Instance 1.

Bay num.	17	5	10	3	7	11	14	4	20	18
Crane num.	3	1	2	1	2	3	3	1	3	2
Task	80	168	180	66	200	180	220	60	50	140
Bay Num.	1	12	6	16	19	13	9	15	2	8
Crane num.	1	2	1	3	3	2	1	2	1	2
Task	46	210	20	90	160	110	250	50	200	160

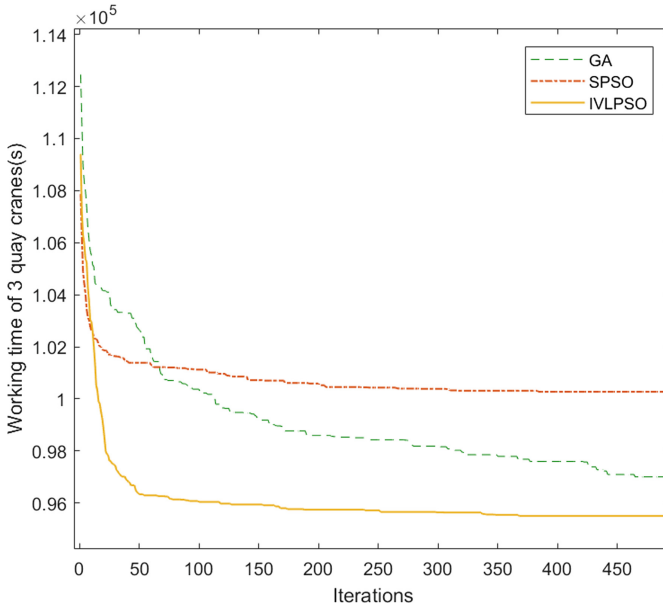


Fig. 2. Convergence of PSO, GA and IVLPSO for Instance 1.

5 Conclusions

This paper proposes an improved PSO (IVLPSO) on the basis of PSO, where various learning mechanisms are employed as the number of iterations changes. The proposed IVLPSO showed to be able to escape from the local optimum, and has a better ability to find better solutions for a quay crane scheduling problem in terms of minimizing the working time. To show its advantages, the proposed new algorithm is compared against GA and PSO over five instances. Experimental results demonstrate the superiority of IVLPSO for the quay crane scheduling problem. IVLPSO converges fast, only requiring one-third of the time that the other two algorithms consume to obtain satisfactory solutions. In our future work, the proposed IVLPSO will be extended to solve multi-objective problems and large-scale problems with more advanced learning mechanisms.

Acknowledgment. This work is partially supported by The National Natural Science Foundation of China (Grants Nos. 71571120, 71271140, 61472257), Natural Science Foundation of Guangdong Province (2016A030310074).

References

1. Chung, S.H., Choy, K.L.: A modified genetic algorithm for quay crane scheduling operations. *Expert Syst. Appl.* **39**(4), 4213–4221 (2012)
2. Kaveshgar, N., Huynh, N., Rahimian, S.K.: An efficient genetic algorithm for solving the quay crane scheduling problem. *Expert Syst. Appl.* **39**(18), 13108–13117 (2012)

3. Correcher, J.F., Alvarez-Valdes, R.: A biased random-key genetic algorithm for the time-invariant berth allocation and quay crane assignment problem. *Expert Syst. Appl.* **89**, 112–128 (2017)
4. Azevedo, A.T.D., Neto, L.L.D.S., Chaves, A.A., Moretti, A.C.: Solving the 3D stowage planning problem integrated with the quay crane scheduling problem by representation by rules and genetic algorithm. *Appl. Soft Comput.* **65**, 495–516 (2018)
5. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: *IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948 (1995)
6. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: *International Symposium on MICRO Machine and Human Science*, pp. 39–43. IEEE (2002)
7. Niu, B., Huang, H., Tan, L., Duan, Q.: Symbiosis-based alternative learning multi-swarm particle swarm optimization. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **14**(1), 4–14 (2017)
8. Ge, H., Sun, L., Tan, G., Chen, Z., Chen, C.L.: Cooperative hierarchical PSO with two stage variable interaction reconstruction for large scale optimization. *IEEE Trans. Cybern.* **47**(9), 2809–2823 (2017)
9. Wei, L.-X., Li, X., Fan, R., Sun, H., Hu, Z.-Y.: A hybrid multi-objective particle swarm optimization algorithm based on R2 indicator. *IEEE Access* **6**, 14710–14721 (2018)
10. Tehsin, S., Rehman, S., Saeed, M.O.B., Riaz, F., Hassan, A., Abbas, M., et al.: Self-organizing hierarchical particle swarm optimization of correlation filters for object recognition. *IEEE Access* **5**, 24495–24502 (2017)
11. Zhu, Q., Lin, Q., Chen, W., Wong, K.C., Coello Coello, C.A., Li, J., et al.: An external archive-guided multiobjective particle swarm optimization algorithm. *IEEE Trans. Cybern.* **47**(9), 2794–2808 (2017)
12. Kang, Q., Xiong, C.F., Zhou, M.C., Meng, L.P.: Opposition-based hybrid strategy for particle swarm optimization in noisy environments. *IEEE Access* **6**, 21888–21900 (2018)
13. Li, H.: Research on simulation based optimization approaches for logistic systems in container port, pp. 86–95 (2013)



An Image Encryption Algorithm Based on Chaotic System Using DNA Sequence Operations

Xuncaizhang, Zheng Zhou, Ying Niu^(✉), Yanfeng Wang,
and Lingfei Wang

School of Electrical and Information Engineering,
Zhengzhou University of Light Industry, Zhengzhou 450002, China
niuying@zzuli.edu.cn

Abstract. Digital image encryption technology is an important means to effectively protect the secure transmission of images. With the advantage of the sensitivity and pseudo randomness of the chaotic map to the initial condition and the inherent spatial configuration of DNA molecule and the unique information processing ability, an image encryption algorithm based on chaotic system and DNA sequence operations is proposed. First, the Logistic map is employed to generate the index sequence to scramble the pixel positions. Second, the hash value of the plaintext image is computed using the SHA-3 algorithm, which is used as the initial key of the 2D-LSCM chaotic system. Third, dynamic DNA encoding is performed on the image and the XOR operations are carried out with the generated random sequence to achieve pixel scrambling and further enhances the security of the encryption algorithm. Finally, preset DNA encoding rules are selected through DNA sequence operations to further enhance the confusion and diffusion characteristics of the algorithm. The experimental and security analysis results show that the algorithm not only has large key space and strong sensitivity to the key, but also can effectively resist statistical attacks and differential attacks.

Keywords: 2D-LSCM chaotic mapping · DNA sequence operation
Scramble and diffusion · Security analysis

1 Introduction

With the remarkable development of advanced science and multimedia technology, digital image processing has been widely used in various aspects of human social life. Such as remote sensing, biomedical, communication, intelligent robot and so on. Therefore, more and more attention has been paid to image information, and it has become more important to protect the security of image data, especially in special fields such as military, commercial and medicine. The conventional encryption technology generally adopts symmetric key system or public key system [1], whose protection means is to encrypt the file into ciphertext, so that the illegal user cannot read it. The traditional encryption algorithms include ECC, 3DES, MD5 and so on [2]. Because the image data have the characteristics of strong correlation between adjacent pixels, large

amount of data and high redundancy, the security of the traditional encryption methods cannot satisfy the real-time requirement of the image data and the efficiency is not high, so it is unsuitable for the encryption of the multimedia data such as images and videos [3]. Therefore, a large number of encryption methods have been proposed, such as chaos-based encryption algorithm [4–7] and DNA-based encryption algorithm [8–11].

As a complex nonlinear system, chaotic system has the sensitivity to initial values and control parameters, pseudo-randomness and unpredictability of the orbit, which coincide with the characteristics required for cryptography [12]. Using a chaotic sequence as a random key can achieve the same encryption effect as a one-time pad (OTP), which is theoretically undecipherable. The traditional encryption algorithm based on chaotic mapping is divided into two processes: scrambling and diffusion [13]. These two processes can be carried out simultaneously or in a step-by-step way. In fact, researchers often combine scrambling and diffusion to get higher security. Since the 2D-LSCM chaotic system has a larger key space, stronger sensitivity and more complex dynamic characteristics and pseudo-randomness, so it is more suitable for image encryption [14].

With the further development of DNA computing, DNA cryptography has become a new field of cryptography [15], and DNA is an important carrier of genetic information storage in organism, which plays an important role in the metabolism of organism. Because of its ultra-large parallelism, ultra-high storage density, ultra-low power consumption and unique molecular structure and intermolecular recognition mechanism, its outstanding information storage and information processing capabilities are determined. DNA molecules have great potential in information security technology such as information encryption, concealment and authentication [16], which provides a new way for the development of modern cryptography. In 1995, Boneh et al. [17] cracked the 56-bit key in 4 months, which was the first time that the traditional data encryption standard was cracked by DNA computing. Subsequently, the development of DNA cryptography has become a hot research topic. In 2000, Gehani et al. [18] took DNA strings as the carrier of information and used biochemical technology to implement an OTP encryption algorithm on DNA molecules.

In recent years, combining the dual advantages of DNA and chaos, the image encryption algorithms based on DNA molecules and chaotic systems have emerged. In 2011, Zheng et al. proposed an image block encryption algorithm based on spatiotemporal chaotic system, which can realize image encryption in parallel and is used for color image encryption [19]. In 2014, Zhang et al. [20] proposed an improved image encryption algorithm based on DNA coding and multi-chaotic mapping. Using the hyper-chaotic system to scramble pixel positions and pixel values and carrying out pseudo DNA operations. Finally, the encrypted image is obtained by DNA decoding. In 2015, Wei et al. [21] proposed an improved image encryption method based on DNA coding and chaotic mapping. This method not only solved the reversibility of the target method, but also combined the characteristics of the chaotic mapping with the DNA coding features, and applied it to the color image encryption method. The algorithm has made great contributions to the intersection of biology and cryptography. In 2018, Shi et al. [22] used chaotic maps to generate scrambling sequences and scrambled the image in the bit plane. At the same time, it achieved the double encryption effects of image scrambling and diffusion. It used the DNA encoding rules

to encode the scrambled image and performed DNA sequence operations to achieve the image diffusion and finally achieved the image encryption effect. The encryption algorithm based on chaotic systems and DNA sequence operations proposed in this paper exactly compensates for the shortcomings of traditional chaotic encryption methods. It can fully utilizes the inherent advantages of DNA and encrypt a large amount of image information, which is in line with the development needs in current era and future digital age, and has a good application prospect in the field of cryptography.

The remaining part of the paper is organized as follows: in Sect. 2, the basic theory of 2D-LSCM chaos system, DNA coding and its operation rules, and bit scrambling technology are briefly introduced. Section 3 presented the algorithm and flow chart of this encryption method. In Sect. 4, the experimental results are shown and a series of security analyses are carried out on the experimental results. Section 5 summarized the paper as a whole.

2 Preliminaries

2.1 Chaotic System

2D Logistic-Sine Coupling Map (2D-LSCM) derived from one-dimensional chaotic map Logistic mapping and Sine mapping [23]. The Logistic mapping is one of the commonly used chaotic maps with three characteristics as follows: (1) extremely dependent on initial conditions; (2) non-periodicity; (3) there exists a singular attractor. Its mathematical expression is defined as follows:

$$f(x) = \mu x(1 - x) \tag{1}$$

where $x \in [0, 1]$, when $\mu \in [3.56995, 4]$, the system is in chaotic state.

The Sine mapping is defined as follows:

$$f(y) = \beta \sin(\pi y) \tag{2}$$

where β is control parameter and $\beta \in [0, 1]$.

Combined with the above two chaotic systems, a complex chaotic system 2D-LSCM is obtained, and its mathematical expression is defined as follows:

$$\begin{cases} x_{i+1} = \sin(\pi(4\theta x_i(1 - x_i) + (1 - \theta)\sin(\pi y_i))) \\ y_{i+1} = \sin(\pi(4\theta y_i(1 - y_i) + (1 - \theta)\sin(\pi x_{i+1}))) \end{cases} \tag{3}$$

Where θ is control parameter and $\theta \in [0, 1]$, when $\theta \in (0, 0.34) \cup (0.67, 1)$, 2D-LSCM is in hyper-chaotic state. It can be seen from the definition that the combination of the two chaotic systems extends the dimension of the system from 1D to 2D, which can effectively improve the complexity of the system, and further obtain more complex chaotic behavior. Compared with 2D Logistic mapping, 2D-LSCM mapping has better ergodicity, larger key space, more complex phase space trajectory, and generate more secure chaotic sequences.

2.2 DNA Coding Rules and Their Operation Rules

The DNA molecule consists of four types of deoxyribonucleotides: adenine (A), cytosine (C), guanine (G), and thymine (T). For two single-stranded DNA molecules, a stable structure of DNA molecules is formed by hydrogen bonds between nucleotides. The chemical structure of bases determines the principle of complementary base pairing, also called Watson-Crick complementary base pairing principle [24], that is to say, A and T are paired by two hydrogen bonds, and G and C are paired by three hydrogen bonds. This natural quaternary combination is similar to the binary formed by semiconductor on-off. Therefore, the storage and calculation of information is performed using the arrangement of bases.

Encoding Rules. If the corresponding encoding rules are followed by $A \rightarrow 11, C \rightarrow 10, G \rightarrow 01, T \rightarrow 00$. The complementary numbers are paired with $00 \leftrightarrow 11$ and $01 \leftrightarrow 10$, and the complementary pairing of $T \leftrightarrow A$ and $G \leftrightarrow C$ is consistent with the base pairing rules. There are 8 kinds of combinations that satisfy the complementary base pairing rules, as shown in Table 1. The pixel value of the grayscale image is between 0 and 255, so it can be represented by 8-bit binary numbers or 4-bit DNA coding. For instance: the decimal number 168, with 8-bit binary number is expressed as $(10101000)_2$, the rule 2 is encoded as GGGA, the rule 6 decoding is 00000010, only through the ways of encoding and decoding can achieve information encryption.

Table 1. Eight DNA encoding and decoding rules

1	2	3	4	5	6	7	8
A-00	A-00	T-00	T-00	G-00	G-00	C-00	C-00
G-01	C-01	G-01	C-01	A-01	T-01	A-01	T-01
C-10	G-10	C-10	G-10	T-10	A-10	T-10	A-10
T-11	T-11	A-11	A-11	C-11	C-11	G-11	G-11

Base Complementary Rules. For each base x of the DNA sequence, the DNA complementary rule must satisfy the following conditions:

$$\begin{cases} x \neq B(x) \neq B(B(x)) \neq B(B(B(x))) \\ x = B(B(B(B(x)))) \end{cases} \tag{4}$$

Where, $B(x)$ is the base pair of x , and it can guarantee the complementary base pairing of the single mapping, according to the rule of complementary base pairing defined in formula (4). That is to say, each base pair is assigned a separate counterpart. Taking into account the number of DNA complementary rules, a total of 6 groups meet the DNA complementary rules, respectively.

Rule 1 : $A \rightarrow T, T \rightarrow G, G \rightarrow C, C \rightarrow A$
 Rule 2 : $A \rightarrow T, T \rightarrow C, C \rightarrow G, G \rightarrow A$
 Rule 3 : $A \rightarrow G, G \rightarrow C, C \rightarrow T, T \rightarrow A$
 Rule 4 : $A \rightarrow G, G \rightarrow T, T \rightarrow C, C \rightarrow A$
 Rule 5 : $A \rightarrow C, C \rightarrow G, G \rightarrow T, T \rightarrow A$
 Rule 6 : $A \rightarrow C, C \rightarrow T, T \rightarrow G, G \rightarrow A$

Base Operation Rules. For grayscale images, the gray value of each pixel is represented by an 8-bit binary number. If DNA encoding is adopted, each pixel needs to encode a 4-base sequence. After converting the image matrix into DNA sequence, the operation rules of DNA sequence is utilized in image processing. According to the complementary pairing rule, encoding is performed for the above rule 1, that is to say, $A \rightarrow 00, G \rightarrow 01, C \rightarrow 10, T \rightarrow 11$, and an operation rule between bases is given in Ref [9] and similar operation rules can be established if other coding rules are adopted.

2.3 SHA Algorithm

The SHA-3(384) algorithm is a hash function based on the sponge structure, which is one of the most basic modules in modern cryptography [25]. With any length of message value as input, a fixed length of hash value is generated. But hash function is an irreversible compression, once hash operation is performed, the result will not be recovered to the plaintext. The key generated by the hash value, even if the original image has a very slight change, the hash value generated by the encryption process will be totally different, and the encryption key will be utterly different. Combining the original image information with the key, the anti-brute force attack is 2^{192} , so the encryption method can effectively resist brute force attack.

2.4 Bit Scrambling

Scrambling is an important means of hiding plaintext information in cryptographic algorithms, and the diffusion of plaintext to ciphertext is realized by position substitution. The bit permutation provides functions such as chaos and diffusion that byte operations cannot achieve. At present, most of the algorithms use XOR operations, this substitution method is low security. By selecting a specific set of plaintext ciphertext pairs, the chaotic sequence of pixel values is deciphered. Pixel value substitution encryption adds a bit scrambling operation based on pixel value to the original substitution encryption, which effectively resists the chosen plaintext attack and enhances the security of the algorithm.

3 Encryption Algorithms

The encryption algorithms used in this paper mainly include 2D-LSCM chaotic system, DNA coding and its XOR rules and subtraction rules, complementary rules, DNA-level scrambling and diffusion operations. The detailed steps of the encryption algorithm are presented in the following.

Step 1: Set the initial value of the chaotic Logistic system and generate the $M \times N$ chaotic sequence;

Step 2: Extract the 8-bit digits after the decimal point of the chaotic sequences and sort according to the order from small to large, the positions of each element in the original sequences are recorded as the index sequences;

Step 3: Sort the pixel values of the image after converting the bit scrambling binary into decimal numbers to generate the scrambled image P' ;

Step 4: Key generation

The gray value of scrambled image is calculated by SHA-3(384), and a set of 384-bit hash values are generated, and the hash values are converted into binary as the key K . Which is used to generate the initial value of the 2D-LSCM chaotic system. The key K is divided into 48 bytes, represented as k_1, k_2, \dots, k_{48} ; record $Q_1 = k_1 \oplus k_2 \oplus k_3 \oplus \dots \oplus k_{16}$; $Q_2 = k_{17} \oplus k_{18} \oplus k_{19} \oplus \dots \oplus k_{32}$; $Q_3 = k_{33} \oplus k_{34} \oplus k_{35} \oplus \dots \oplus k_{48}$.

$$\begin{cases} x_0 = \frac{1}{256} \text{mod}(\text{Bin2dec}(Q_1), 256) \\ y_0 = \frac{1}{256} \text{mod}(\text{Bin2dec}(Q_2), 256) \\ \theta = \frac{1}{256} \text{mod}(\text{Bin2dec}(Q_3), 256) \end{cases} \quad (5)$$

Step 5: According to the initial values of x_0, y_0 and θ , iterate 2D-LSCM chaotic system $M \times N$ times and get chaotic sequence as shown in the formula (6).

$$X = [x_1, x_2, \dots, x_{M \times N}] \quad (6)$$

The X sequence is further processed to get the sequence $A = [a_1, a_2, \dots, a_{M \times N}]$.

$$a_i = \text{mod}(\text{floor}(x_i \times 10^{14}), 8) \quad i = 1, 2, \dots, M \times N \quad (7)$$

Step 6: According to the values of a_i , different encoding rules are selected for image encryption, and the image matrix is transformed into a DNA sequence, and the diffused DNA sequence is obtained through the following calculation, which is called C sequence.

- ① If $a_i = 0$, then $C_i = P_i \oplus C_{i-1}$;
- ② If $a_i = s$, then $C_i = R_s(P_i) \oplus C_{i-1}$, $s = 1, 2, \dots, 6$;
- ③ If $a_i = 7$, then $C_i = P_i - C_{i-1}$.

where P_i is the DNA sequence converted from plaintext image; C_i is the diffused DNA sequence; \oplus and '-' are DNA exclusive OR and subtraction operations as is shown in Ref [9], respectively; R_s is the s th base complementary rule as is shown in Sect. 2.2.

Step 7: The image after DNA operation is decoded and converted into pixel form, which is an encrypted image. The specific encryption flowchart is shown in Fig. 1.

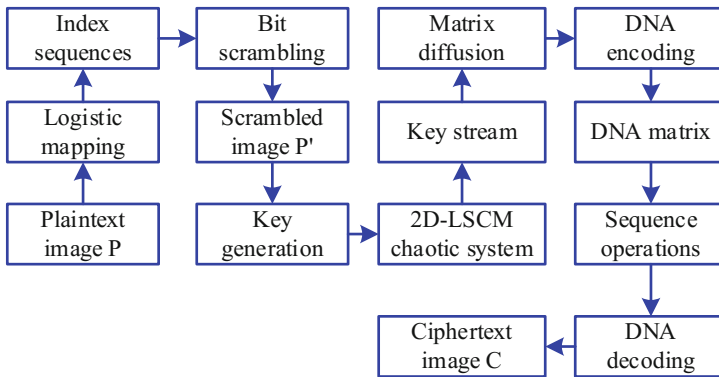


Fig. 1. The flowchart of encryption algorithm.

4 Experimental Results and Security Analysis

In the environment of MATLAB 2017b, the algorithm proposed in this paper was simulated, and the standard 256*256 Lena grayscale image was used as the plaintext image for the simulation experiment. The experimental results are shown in Fig. 2, where Fig. 2(a) and (d) are the plaintext Lena image and Boat image, Fig. 2(b) and (e) are the ciphered Lena image and Boat image, and Fig. 2(c) and (f) are the deciphered Lena image and Boat image, respectively.

A good encryption algorithm should be sensitive to the key and be able to resist common attacks, such as exhaustive attacks, statistical attacks, differential attacks and data loss attacks, and must have a large enough key space to resist brute force attacks. In this section, we will discuss and analyze the performance and security of the proposed encryption algorithm.

4.1 Exhaustive Attack Analysis

Key Space Analysis. A good encryption algorithm must have a large enough key space to resist brute-force attacks against the key. In this encryption algorithm, the key contains: x , y , θ and SHA-3 functions. If the calculation accuracy of x , y and θ is 10^{-14} , the key space of the 2D-LSCM chaotic system is $10^{14} * 10^{14} * 10^{14} = 10^{42}$, and the key space of SHA-3(384) is 2^{192} . The total key space is: $10^{42} * 2^{192} \approx 6.28 * 10^{99}$. It can be seen that the algorithm has enough key space to resist brute force attacks.

Key Sensitivity Analysis. To test the sensitivity of the key, for the 2D-LSCM chaotic system mapping, the value of x_0 is increased by 0.00000001 and the other key is unchanged. The encrypted image is decrypted using the modified key and the decryption result is shown in Fig. 3(c). It can be seen that the original image cannot be

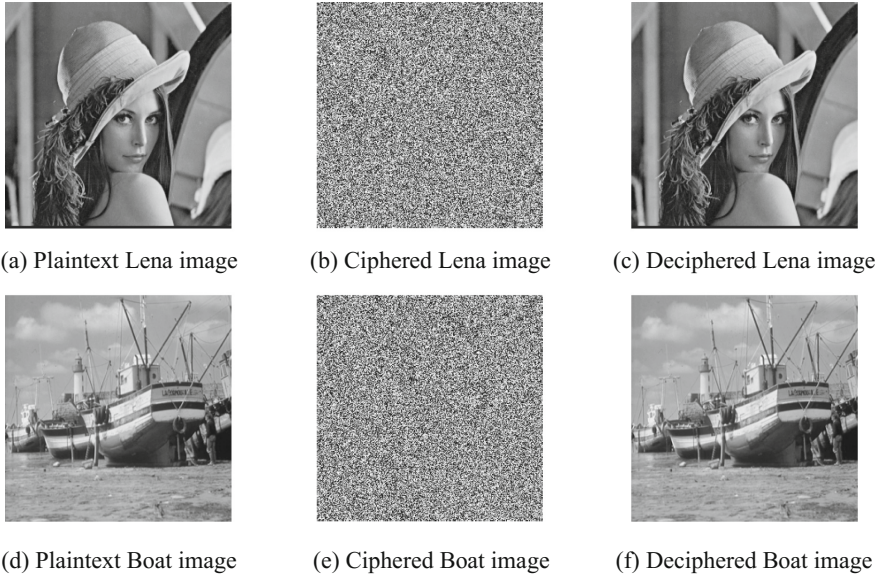


Fig. 2. Experimental simulation results.

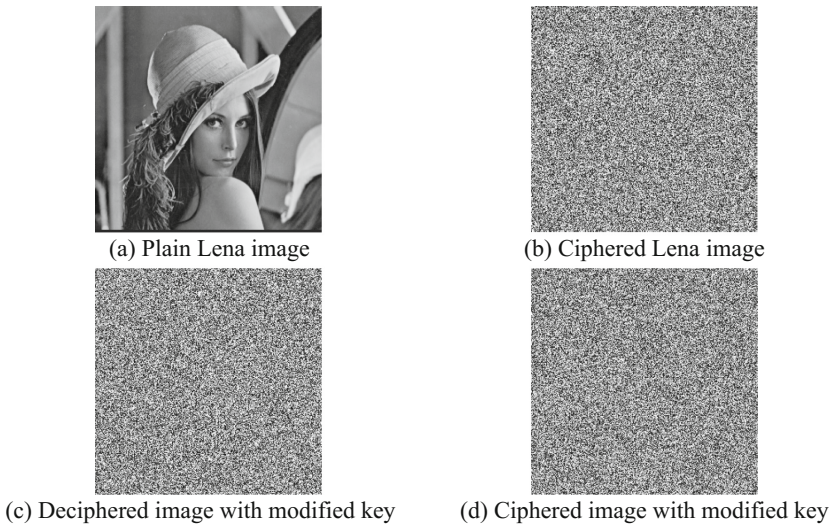


Fig. 3. Plaintext image and ciphered image, ciphered and deciphered image with modified key.

decrypted correctly after minor changes to the key. Furthermore, the image is re-encrypted with the modified key, and the encrypted image is shown in Fig. 3(d). Compared with 3(b), it is shown that the difference rate of the corresponding pixels between two ciphertext images is more than 99.62%, it can be seen that as long as a

small change in the key, the encryption and decryption results are different, which shows that the algorithm has strong key sensitivity, resist brute force attacks and has good key security.

4.2 Statistical Attack Analysis

Histogram Analysis. The histogram of the image can reflect the general regularity of the image and intuitively visualize the distribution of the image characteristics. Statistical analysis is performed on the plaintext image and the ciphered image, wherein Fig. 4(a) and (b) show plaintext and ciphered Lena image histogram, 4(c) and 4(d) represent plaintext and ciphered Boat image histogram, respectively. According to the analysis of the pixel values of the plaintext image, the pixel values of the plaintext image are relatively concentrated, that is, the pixel distribution at the two ends of interval (0, 255) is relatively small and the middle distribution is more; and the corresponding histogram of the ciphered image is basically uniformly distributed. It is difficult for an attacker to recover the plaintext image using the statistical properties of pixel gray values. This shows that the algorithm has capacity to resist statistical attacks.

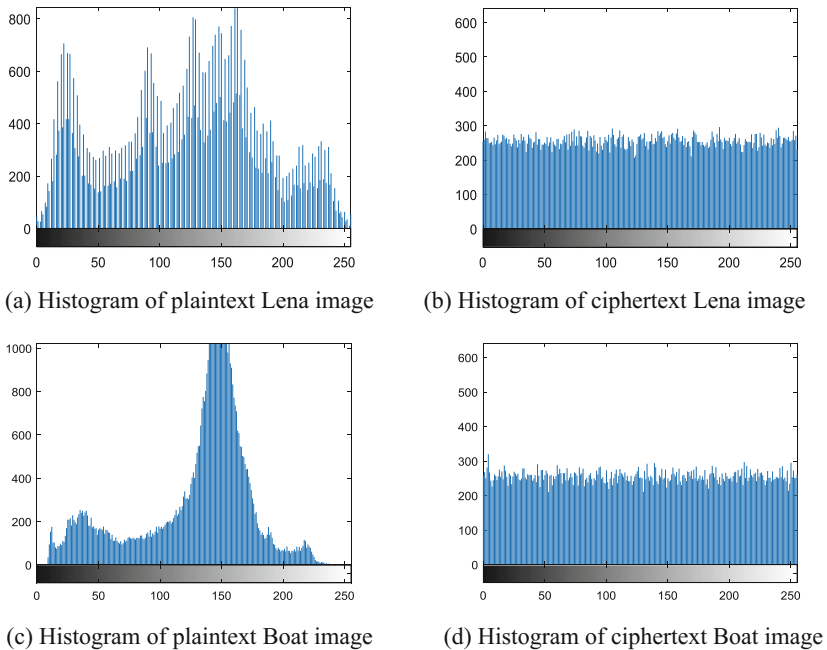


Fig. 4. Gray histogram analysis.

Correlation Analysis. The correlation coefficient of adjacent pixels can reflect the diffusion degree of image pixels. The closer the correlation coefficient is to 0, the less relevance is between the pixels of the image. The closer to 1, the stronger the

correlation between pixels is. In the plaintext image, the correlation between adjacent pixels is very high. To resist statistical attacks, the correlation of adjacent pixels of the ciphered image must be reduced. 2500 pairs of adjacent pixels in the horizontal, vertical and diagonal directions are randomly selected from the plaintext image and ciphered image, and the correlation between pixels is calculated using the formula (8)-(11).

$$E(x) = \frac{1}{N} \sum_{i=1}^N x_i \quad (8)$$

$$D(x) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))^2 \quad (9)$$

$$COV(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))(y_i - E(y)) \quad (10)$$

$$\rho_{xy} = \frac{COV(x, y)}{\sqrt{D(x)} \times \sqrt{D(y)}} \quad (11)$$

Where x and y are the grayscale values of adjacent pixels in the image, $COV(x, y)$ is covariance, $D(x)$ is variance, and $E(x)$ is the mean value. Similarly, the correlation coefficient of the adjacent pixels in the plaintext image and ciphered image are compared as shown in Table 2, and the correlation coefficients of the adjacent pixels of the ciphered Lena image and Boat image are 0.0029096 and 0.0030657, respectively. Figure 5 shows the correlation between the plaintext image and the ciphered image in the horizontal, vertical and diagonal directions. Therefore, the image encryption algorithm has strong ability to resist statistical attacks.

Table 2. Correlation coefficients of adjacent pixels between plaintext and ciphertext image

Correlation coefficient	Horizontal	Vertical	Diagonal
Plaintext Lena	0.9704	0.9420	0.9129
Ciphered Lena	0.0010	-0.0019	0.0211
Plaintext Boat	0.9440	0.9272	0.8842
Ciphered Boat	0.0219	-0.0077	-0.0013

Information Entropy. Information entropy is defined as the degree of uncertainty in the system. It can be used to express the uncertainty of image information. The more chaotic the information of the image, the higher the entropy. For grayscale images, the more uniform the distribution of gray values, the greater the information entropy, the greater the randomness and the higher the security. The formula of information entropy is defined as follows:

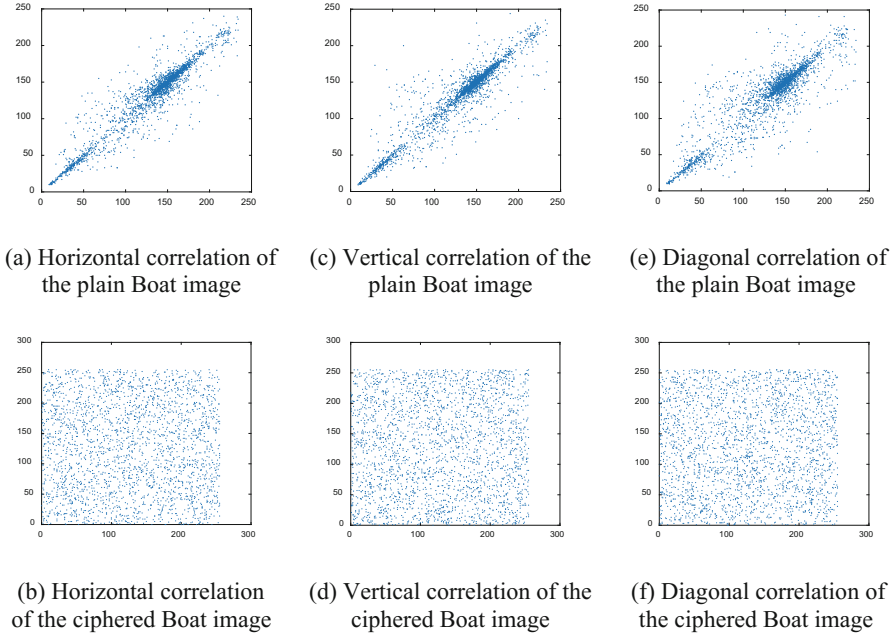


Fig. 5. Correlation analysis of boat as a ciphered image in three directions.

$$H(m) = - \sum_{i=0}^L P(m_i) \log_2 P(m_i), \sum_{i=0}^L P(m_i) = 1 \quad (12)$$

Where L is the grayscale level of the image, m_i is the i th gray value of the image, and $P(m_i)$ is the probability that m_i appears. For 256 grayscale images, the theoretical value of information entropy is 8. The closer information entropy is to the theoretical value, the less likely the image is attacked. The information entropy of the ciphered Lena image and Boat image are 7.9893 and 7.9889, respectively, which shows the effectiveness of the encryption algorithm.

4.3 Differential Attack Analysis

Differential attack means that the attacker can change the plaintext slightly, compare the changes before and after the corresponding ciphertext, and then find out the relationship between the plaintext image and the ciphertext image. The number of pixel change rate (NPCR) and unified average changing intensity (UACI) are used to detect the ability of image encryption schemes to resist differential attacks. The mathematical formulas for NPCR and UACI are shown as follows:

$$C(i, j) = \begin{cases} 0, & \text{if } P_1(i, j) = P_2(i, j) \\ 1, & \text{if } P_1(i, j) \neq P_2(i, j) \end{cases} \quad (13)$$

$$NPCR = \frac{\sum_{i=1}^M \sum_{j=1}^N C(i,j)}{M \times N} \times 100\% \quad (14)$$

$$UACI = \frac{\sum_{i=1}^M \sum_{j=1}^N |P_1(i,j) - P_2(i,j)|}{255 \times M \times N} \times 100\% \quad (15)$$

Where M and N represent the length and width of the image, respectively, and $P_1(i,j)$ and $P_2(i,j)$ represent the corresponding ciphertext values before and after the plaintext changes, respectively. The closer the NPCR value is to 100%, the more sensitive the image encryption scheme is to the plaintext, the stronger the ability to resist differential attack. The ideal value of UACI is 33%, and the closer it is to the ideal value, the stronger the ability to resist differential attacks. According to the above formulas, the NPCR and UACI of the Lena image are 99.64% and 31.01%, and the NPCR and UACI of the Boat image are 99.65% and 28.32%, respectively. Which proves that the image encryption scheme has the ability to resist differential attacks.

5 Conclusions

In this paper, an image encryption algorithm based on the DNA sequence operations and 2D-LSCM chaotic system is proposed. The algorithm first calculates the initial value of the chaotic sequence according to the plaintext information, and makes the algorithm not attacked by the plaintext. Second, the complexity of the algorithm and the unpredictability of the ciphertext are increased by using the chaotic sequence. Finally, according to the position of pixels, randomly select the encoding rules to achieve encryption effect. The experimental analysis shows that the algorithm has better encryption effect, larger key space and higher key sensitivity. In addition, the algorithm can also resist statistical attack and differential attack. Therefore, the encryption scheme proposed in this paper can be used for secure image transmission.

Acknowledgments. The work for this paper was supported by the National Natural Science Foundation of China (Grant nos. 61602424, 61472371, 61572446, and 61472372), Plan for Scientific Innovation Talent of Henan Province (Grant no. 174100510009), Program for Science and Technology Innovation Talents in Universities of Henan Province (Grant no. 15HASTIT019), and Key Scientific Research Projects of Henan High Educational Institution (18A510020).

References

1. Baldi, M., Bianchi, M., Chiaraluce, F., Rosenthal, J., Schipani, D.: Enhanced public key security for the McEliece cryptosystem. *J. Cryptol.* **29**(1), 1–27 (2016)
2. Mahajan, P., Sachdeva, A.: A study of encryption algorithms AES, DES and RSA for security. *Glob. J. Comput. Sci. Technol.* **13**(15), 15–22 (2013)
3. Zaki, A.K., Indiramma, M.: A novel redis security extension for NoSQL database using authentication and encryption. In: *IEEE International Conference on Electrical, Computer and Communication Technologies*, pp. 1–6. IEEE (2015)

4. Suryadi, M.T., Nurpeti, E., Widya, D.: Performance of chaos-based encryption algorithm for digital image. *Telkomnika* **12**(3), 675–682 (2014)
5. Niu, Y., Zhang, X., Han, F.: Image encryption algorithm based on hyperchaotic maps and nucleotide sequences database. *Comput. Intell. Neurosci.* (2017). Article ID 4079793
6. Cui, G., Liu, Y., Zhang, X., Zhou, Z.: A new image encryption algorithm based on DNA dynamic encoding and hyper-chaotic system. In: He, C., Mo, H., Pan, L., Zhao, Y. (eds.) *BIC-TA 2017. CCIS*, vol. 791, pp. 286–303. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-7179-9_22
7. Zhang, Y.: The image encryption algorithm based on chaos and DNA computing. *Multimed. Tools Appl.* **77**(16), 21589–21615 (2018)
8. Gehani, A., Labean, T., Reif, J.: DNA-based cryptography. *Asp. Mol. Comput.* **54**(456), 233–249 (2004)
9. Zhang, X., Zhou, Z., Niu, Y.: An image encryption method based on the feistel network and dynamic DNA encoding. *IEEE Photonics J.* (2018). <https://doi.org/10.1109/JPHOT.2018.2859257>
10. Zhang, X., Han, F., Niu, Y.: Chaotic image encryption algorithm based on bit permutation and dynamic DNA encoding. *Comput. Intell. Neurosci.* (2017). Article ID 6919675
11. Zhang, X., Zhou, Z., Jiao, Y., Niu, Y., Wang, Y.: A visual cryptography scheme-based DNA microarrays. *Int. J. Perform. Eng.* **14**(2), 334–340 (2018)
12. Guo, Y.: A summary of image encryption algorithm based on chaotic sequence. *Open Autom. Control Syst. J.* **6**(1), 1110–1114 (2014)
13. Yang, C., Hua, M., Jia, S.: Image encryption algorithm based on chaotic mapping and Chinese remainder theorem. *Metall. & Min. Ind.* **7**(4), 206–212 (2015)
14. Hua, Z., Jin, F., Xu, B., Huang, H.: 2D Logistic-Sine-Coupling map for image encryption. *Signal Process.* **149**, 148–161 (2018)
15. Kumar, V., Raheja, E.G., Sareen, M.S.: New field of cryptography: DNA cryptography. *Int. J. Comput. Technol.* **37**(6), 24–27 (2013)
16. Dodis, Y., An, J.H.: Concealment and its applications to authenticated encryption. In: Biham, E. (ed.) *EUROCRYPT 2003. LNCS*, vol. 2656, pp. 312–329. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-39200-9_19
17. Dan, B., Dunworth, C., Lipton, R.J.: Breaking DES using a molecular computer. In: *Proceedings of a DIMACS Workshop*, 4 April 1995, Princeton University, vol. 27 (1995)
18. Gehani, A., LaBean, T., Reif, J.: DNA-based cryptography. In: Jonoska, N., Păun, G., Rozenberg, G. (eds.) *Aspects of Molecular Computing. LNCS*, vol. 2950, pp. 167–188. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-24635-0_12
19. Zheng, H.Y., Wen-Jie, L.I., Xiao, D.: Novel image blocking encryption algorithm based on spatiotemporal chaos system. *J. Comput. Appl.* **31**(11), 3053–3055 (2011)
20. Zhang, Q., Liu, L., Wei, X.: Improved algorithm for image encryption based on DNA encoding and multi-chaotic maps. *AEUE-Int. J. Electron. Commun.* **68**(3), 186–192 (2014)
21. Liu, Y., Zhao, G., Wei, G., Zhang, J.: An improved DNA coding image encryption algorithm Combining entropy and chaos. In: *International Symposium on Computers and Informatics*, pp. 880–887. Atlantis Press (2015)
22. Shi, F., Zhang, H., Zhang, X.: Image encryption algorithm based on chaotic map and DNA coding. *Comput. Eng. Appl.* **54**(5), 91–95 (2018)
23. Hua, Z., Zhou, Y., Pun, C.M., Chen, C.L.P.: 2D sine logistic modulation map for image encryption. *Inf. Sci. Int. J.* **297**, 80–94 (2015)
24. Zenk, J., Tuntivate, C., Schulman, R.: Kinetics and thermodynamics of Watson-Crick base pairing driven DNA Origami dimerization. *J. Am. Chem. Soc.* **138**(10), 3346–3354 (2016)
25. Al Shaikhli, I., Alahmad, M., Munthir, K.: Hash function of finalist SHA-3: analysis study. *Int. J. Adv. Comput. Sci. Inf. Technol.* **2**(2), 1–12 (2014)



An Image Encryption Algorithm Based on Dynamic DNA Coding and Hyper-chaotic Lorenz System

Guangzhao Cui, Lingfei Wang, Xuncaizhang^(✉), and Zheng Zhou

School of Electrics and Information Engineering,
Zhengzhou University of Light Industry, Zhengzhou 450002, China
zhangxuncaizhang@163.com

Abstract. A new image encryption algorithm with low correlation coefficient and high information entropy based on dynamic DNA coding and hyper-chaotic Lorenz system is proposed in this paper. In order to generate the initial values in hyper-chaotic system to construct DNA sequence matrix and dynamic S-box, this algorithm use the SHA-256 algorithm to generate a common key. Then use the SCAN mode and the S-box to scramble the pixel position. In order to scramble the pixel values, this paper uses the DNA sequence matrix to calculate cipher-text. Theoretical analysis and simulation results show that the algorithm improves the sensitivity of key space, and the encryption system can resist statistical attacks and differential attack.

Keywords: Dynamic DNA coding · SCAN mode · Dynamic 2D S-box

1 Introduction

Since human entered the modern society, the security of information has been paid more and more attention. The information security influences politics, military affairs and economy to some extent. It is an urgent task to protect the security of information. There are many ways to protect the security of information, such as encrypting information, adding security mechanisms and enhance security of operating system. Among them, applying encryption algorithm to encrypt information is the most effective technical mean to protect information security.

Traditional information encryption algorithms such as (Data Encryption Standard, DES) [1] and (Advanced Encryption Standard, AES) [2] can effectively protect text information. However, with the development of computer hardware's computing capability and the great progress of cracking encryption technique, these algorithms had been proved weak. Moreover, image data has the characteristics of large data volume, high data redundancy and strong information correlation [3, 4]. So, these traditional encryption algorithms are no longer suitable for image encryption system. Therefore, in recent years, many scholars have been researching for information security and have proposed many new algorithms about digital image encryption system. For example, Bourbakis and Alexopolos [5] proposed an algorithm based on SCAN mode in 1992. Yi [6] proposed a chaotic scrambling encryption system.

Acharya et al. [7] proposed an algorithm based on Hill encryption method, using the invertible matrix to encrypt image data. The method using DNA strand to one-time pad system was firstly proposed by Gehani [8].

In these new methods, chaotic system has pseudo-random characteristic, unpredictable track, sensitive initial values and sensitive parameters, it has promoted the progress of encryption algorithm [9]. However, the randomness of low-dimensional chaotic system is poor, and the track of low-dimensional is predicted in a short time. If the key space is small, the performance of the encryption system will discount. And the system is easily to be cracked by brute-force attack. Compared with low-dimensional chaotic systems, hyper-chaotic systems have more initial values, larger key space, more complex tracks and stronger pseudo-random characteristic. The application of hyper-chaotic system to the encryption process can greatly increase the security of the system.

The invention of bio-computing technology has opened up a new field for computer science, electrical science and information science. The methods of bio-computing include membrane computing, DNA computing etc. In 1994, Adleman [10] completed the first DNA computing experiment and published the research results on Science. Subsequently, some scholars studied in depth and discovered many advantages of DNA computing [11]. The DNA molecule has a complementary double stranded structure. This unique structure can be used for large-scale parallel operation with ultra-low energy consumption. The units of DNA molecular fragment are KB (kilo-base pair) and MB (mega-base pair). From these characteristics of DNA molecules, storage density of DNA molecules is very large. Computation speed can be greatly improved by using the method of DNA molecular computation [12].

The algorithm proposed in this paper combines the high efficiency of DNA coding and the pseudo-random characteristics of the hyper-chaotic system, it makes the encryption system more efficient and can save the running time [13].

2 Fundamental Theories

2.1 Hash Functions

Hash algorithms are very widely used algorithms in cryptography. The output of a hash algorithm is a fixed length string. Hash operation is a compression operation with certain mapping relations. It means that different information sequences may produce the same Hash sequence after Hash algorithms operated. The space of Hash sequence is much less than the information sequence, and a Hash sequence may be produced by a variety of information sequences. Therefore, it is impossible to deduce the original information sequence through the Hash sequence, the Hash algorithms are irreversible. Hash algorithms can not only be applied for digital signature, but also can be applied to verify the authenticity of information and the integrity of data. The standard of Hash algorithms are divided into two main categories: MD series (including MD4, MD5, HAVAL, etc.) and SHA series (including SHA-1 [12], SHA-2, SHA-256, SHA-384 and SHA-512). The algorithm this paper proposed uses the SHA-256 algorithm of SHA series. Original image is input to SHA-256 algorithm to obtain the Hash sequence H as the key to decryption system. Hash sequence H can also be applied to verify the authenticity and integrity of the information.

2.2 Coding and Computing of DNA Sequences

DNA Coding. After Adleman completed the first DNA computing experiment in 1994, some researchers later found that DNA computing [13] had a series of characteristics like parallel computing ability, huge storage capacity and ultra-low power consumption. DNA molecules are composed of the four DNA nucleotides, which are A (adenine), T (thymine), G (guanine) and C (cytosine). Nucleotides A and T, G and C follow the principle of complementary pairing. While A, T, G and C respectively representing numbers, they can represent decimal digits 0 to 3. These four decimal digits can be converted to binary digits as 00, 01, 10 and 11, so each nucleotide can transfer 2 bit information. This coding has $4! = 24$ kinds of encoding methods. But only 8 kinds of encoding methods follow the rule of complementary pairing. These 8 kinds of encoding methods are shown in Table 1.

Table 1. 8 kinds of DNA coding rules

Rule	1	2	3	4	5	6	7	8
A	00	00	01	01	10	10	11	11
T	11	11	10	10	01	01	00	00
G	01	10	00	11	00	11	01	10
C	10	01	11	00	11	00	10	01

For a digital image, each pixel has a value between 0 and 255 and this value can be converted to an 8 bit binary string. Each pixel can be represented by four DNA nucleotides. For example, choose the first kind of encoding rule to encode the decimal digit 27, first convert decimal digit 27 to an 8 bit binary string as 00011011, then use DNA sequence AGCT to represent this binary string. Decoding process is the inverse process of encoding. If you choose the wrong decoding rules to decode, you will get the wrong information. For example, if the DNA sequence as ATGC encoded by rule 1 is decoded by rule 2, an 8-bit binary string as 00100111 is obtained. The decimal digit of this binary string is 39, it's different to 27. Therefore, using different encoding methods to store information can protect the security of information to some extent. However, only a single scheme is adopted, the security of the encryption system is not strong [14]. So some other encryption methods should be added into the algorithm.

Table 2. Computing rules of DNA coding in rule 1

+	A	G	C	T	-	A	G	C	T
A	A	G	C	T	A	A	T	C	G
G	G	C	T	A	G	G	A	T	C
C	C	T	A	G	C	C	G	A	T
T	T	A	G	C	T	T	C	G	A

Computing Rules of DNA Sequence. DNA computing rules are algebraic operation based on the double helix structure of DNA molecules, such as addition and subtraction. Table 1 has introduced 8 kinds of DNA encoding rules. Each DNA encoding rule introduced in Table 1 corresponds to an addition rule and a subtraction rule. In addition and subtraction, only the binary digits represented by nucleotides are added or subtracted. The result only need to be retained the last 2-bit binary digits. Table 2 lists the corresponding computing rules of rule 1 encoding method.

2.3 Hyper-chaotic System

Hyper-chaotic Lorenz system [15] can generate four 4D chaotic sequences, which is defined in formula (1):

$$\begin{cases} \dot{x} = a(y - x) + u \\ \dot{y} = cx - y - xz \\ \dot{z} = xy - bz \\ \dot{u} = -yz + ru \end{cases} \quad (1)$$

In Formula (1), a, b, c, r are the parameters of the hyper-chaotic Lorenz system, when $a = 10, b = 8/3, c = 28,$ and $-1.52 < r < -0.06,$ the system enters a state of chaos, the orbit of the hyper-chaotic Lorenz system is complex, unpredictable, and it's suitable for encryption. In this image encryption algorithm, the initial value of the hyper-chaotic Lorenz system is determined by the 256-bit binary Hash sequence generated by the original image. The 256-bit binary Hash sequence is divided into 32 blocks, with the Numbers k_1 to k_{32} . Each block is an 8-bit binary string. The initial value of hyper-chaotic Lorenz system is determined by these 8-bit binary strings. The formulas to calculate these initial values are in formula (2):

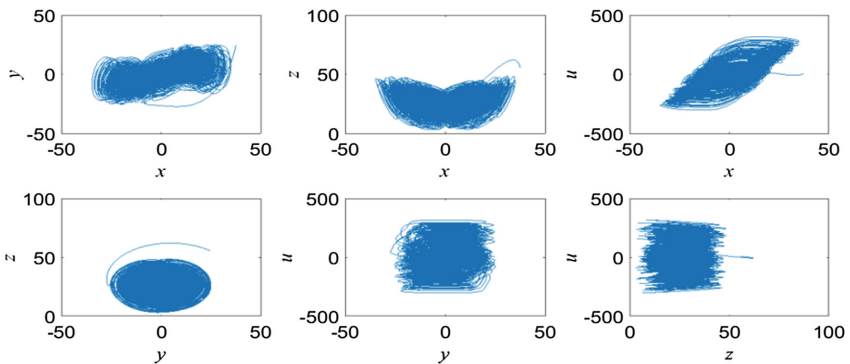


Fig. 1. Simulation result of hyper-chaotic Lorenz system

$$\begin{cases} x(1) = \frac{(k_1 \oplus k_2 \oplus k_3 \oplus k_4 \oplus k_5 \oplus k_6 \oplus k_7 \oplus k_8)}{4} + 2 \\ y(1) = \frac{(k_9 \oplus k_{10} \oplus k_{11} \oplus k_{12} \oplus k_{13} \oplus k_{14} \oplus k_{15} \oplus k_{16})}{4} + 4 \\ z(1) = \frac{(k_{17} \oplus k_{18} \oplus k_{19} \oplus k_{20} \oplus k_{21} \oplus k_{22} \oplus k_{23} \oplus k_{24})}{4} + 6 \\ u(1) = \frac{(k_{25} \oplus k_{26} \oplus k_{27} \oplus k_{28} \oplus k_{29} \oplus k_{30} \oplus k_{31} \oplus k_{32})}{4} + 8 \end{cases} \quad (2)$$

The image simulated by hyper-chaotic Lorenz system is shown in Fig. 1. The initial values are generated by image Lena.

3 Design of Encryption System Scheme

3.1 SCAN Mode

The algorithm proposed in this paper uses SCAN mode to reduce correlation. SCAN mode is a position scrambling mode, which uses one or more scanning methods to scramble and rearrange the pixel position of the original image to form a new pixel image. Some frequently-used methods of SCAN mode is shown in Fig. 2.

SCAN mode is a mode that selects one or more rules shown in Fig. 2 to rearrange the pixels. By extending these rules can also deduce a variety of paths which are not listed here. Decryption process using SCAN mode is the inverse of encryption process.

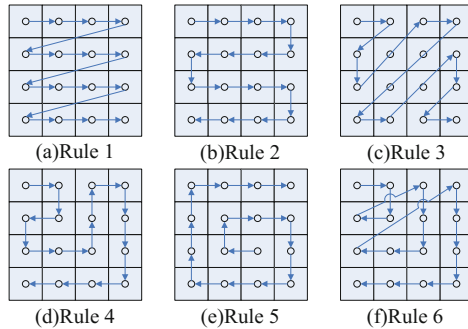


Fig. 2. Some frequently-used methods of SCAN mode

3.2 Design of S-Box

S-box is widely applied in cryptography [16]. There are two functions of S-box. One is mapping pixel values. The original image is encrypted by changing the size of pixel value. The other is mapping the position of the pixel and the original image is encrypted through the method of scrambling position. This paper proposed a dynamic 2D S-boxes algorithm to save computation time. The S-boxes used for each encryption process are different, so this encryption system is more secure. The initial values of hyper-chaotic Lorenz system are generated by Hash algorithm, and S-boxes are generated by the chaotic sequences of the hyper-chaotic Lorenz system. Steps to generate S-boxes are as follow:

- (1) The first step is extracting a 16 numbers unordered sequence as x' from sequence x in Formula (2) and recording the positions of these numbers to generate the X-coordinate of S-box;
- (2) The second step is extracting a 16 numbers unordered sequence as y' from sequence y in Formula (2) and recording the positions of these numbers to generate the Y-coordinate of S-box;
- (3) The third step is rearranging the sequence x' in a small to large way to form a new sequence x'' and recording the position of these numbers in sequence x'' .
- (4) The fourth step is rearranging the sequence y' in a small to large way to form a new sequence y'' and recording the position of these numbers in sequence y'' .
- (5) The fifth step is extracting a 16×16 pixels block in original image, and scrambling the coordinates of these pixels in this block. The abscissa is mapped from the sequence x' to the sequence x'' and the ordinate is mapped from y' to y'' to obtain the scrambled block.
- (6) Repeat steps (1) to (5) until the entire original image information is encrypted.
- (7) The flow chart of generating the 2D S-box is shown in Fig. 3. The decryption process of the S-box is reverse to the encryption process. The advantage of generating a 2D S-box is that it only needs two unordered sequences of 16 numbers to generate an S-box with the size of 16×16 . The 16×16 size S-box in other algorithms need 256 unordered numbers, with large computation and low algorithm efficiency.

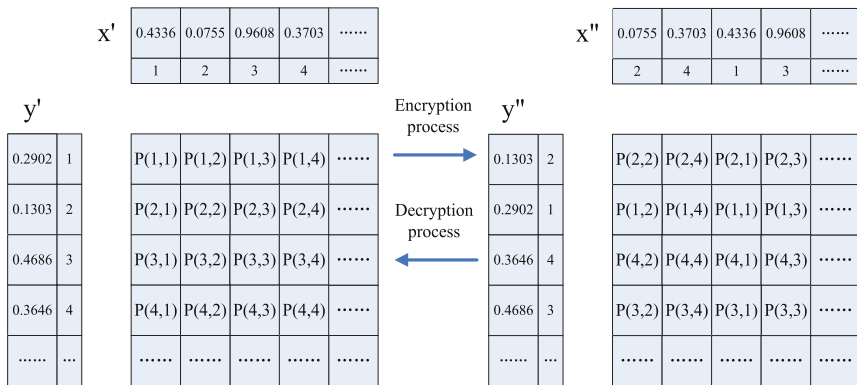


Fig. 3. Design of 2D S-box

3.3 Encryption Scheme

Taking the 256×256 pixel Lena image as an example, the encryption algorithm steps are as follows:

- (1) SHA-256 algorithm is used to calculate the plaintext image and obtain the 256-bit binary sequence H ;

- (2) The rule of DNA coding is determined by the first 32 binary digits of H sequence. The formula to calculate the rule is in formula (3):

$$rule = mod(hex2dec(H[1 : 32]), 8) \tag{3}$$

- (3) Chaotic sequences x, y, z and u are obtained by using sequence H through the Formula (2) in Sect. 2.3;
- (4) Using the formula (4) to construct unordered sequence x' and unordered sequence y', 16 * 16 dynamic S-boxes are generated by the sequence x' and y';

$$\begin{cases} x'(i) = x(i) * 1000 - floor(x(i) * 1000) \\ y'(i) = y(i) * 1000 - floor(y(i) * 1000) \end{cases} \tag{4}$$

- (5) Using the SCAN mode to scramble the original image to obtain a new image C₁;
- (6) Dealing image C₁ by the dynamic 2D S-box method in Sect. 3.2 to obtain the a image C₂;
- (7) Cycling each pixel value of image C₂ by 6 bit to the left to obtain a new image C₃;
- (8) Using DNA Coding to encode image C₃ and hyper-chaotic sequence z by the rule generated in step (2), then use DNA addition to calculate them, and get the encrypted image C₄;
- (9) Chaotic sequence u and pixels in image C₄ were used to carry out XOR calculation to obtain encrypted image C;

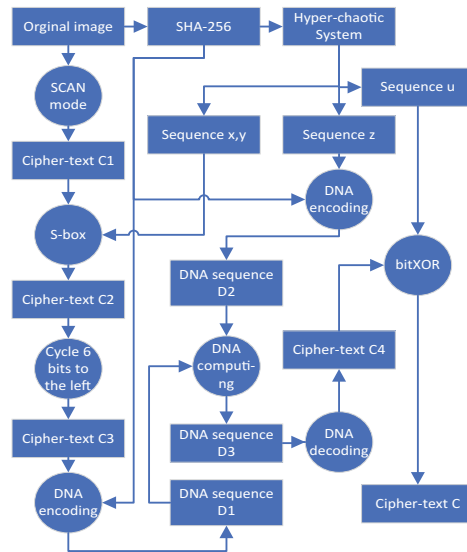


Fig. 4. Flow chart of the algorithm

The flow chart of encryption process is shown in Fig. 4. The decryption process in this scheme is the inverse process of the encryption process. To avoid repetition, the decryption process is not described in this article.

4 Security Analysis of Encryption System

The original image and the encrypted image are shown in Fig. 5. A secure information encryption system should have the ability to resist many kinds of attacks. Generally speaking, there are four schemes to attack an information encryption system: cipher-text-only attack, known plaintext attack, chosen plaintext attack [17] and chosen cipher-text attack.

To analyze the security of an encryption system, the following six factors need to be analyzed.

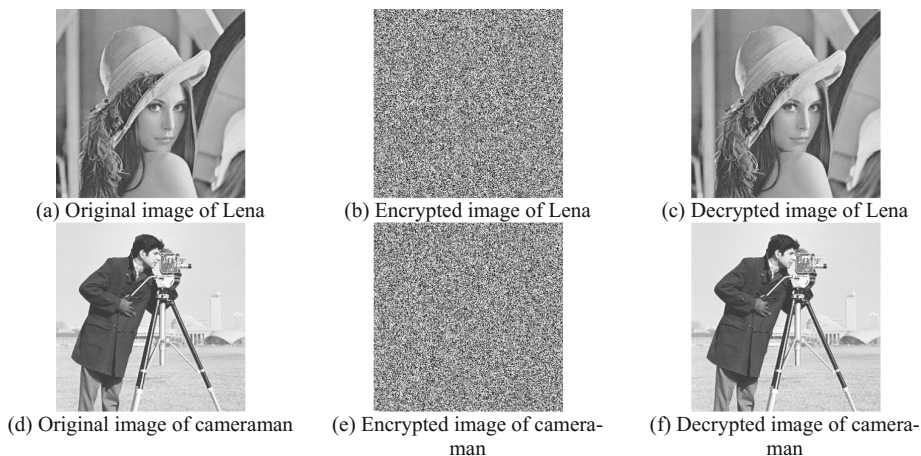


Fig. 5. Original image, encrypted image and decrypted image

4.1 Key Space and Key Sensitivity Analysis

The encryption system which has larger key space will have the stronger ability to resist the brute-force attack. The SHA-256 algorithm has a key space of 2^{128} , the key space of hyper-chaotic system initial values to generate the chaotic sequence is 10^{36} and there are 8 kinds of DNA coding. So the key space of the encryption system is about $3.4028 * 10^{74}$, the key space is big enough to resist brute-force attack.

When the key has a slightly change, it will have a serious impact on the decryption process and make the system unable to be decrypted normally. When the system has a strong key's sensitivity, we can consider that this system is more secure. To analyze the sensitivity of the key only need to make small changes to the key to compare the decryption effect. The initial values of hyper-chaotic system in decryption process of Lena are: $x(1) = 37.5000001$, $y(1) = 24.5000001$, $z(1) = 55.7500001$, $u(1) = 8.52730001$ Fig. 6 shows that when the initial values have a slightly change, the decryption process of Lena image will be failed.

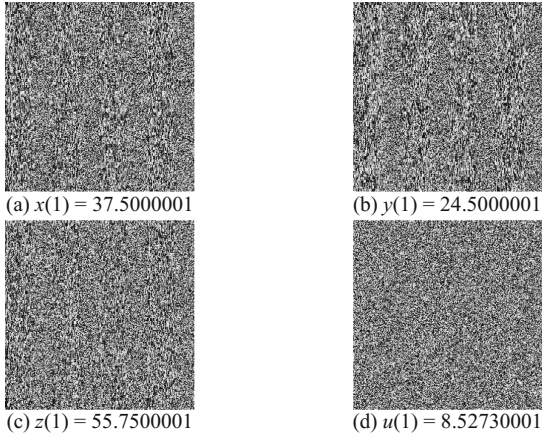


Fig. 6. Key sensitivity analysis

4.2 Histogram Analysis

This section analyzed the security of the encryption algorithm by comparing the histograms of the original image and the encrypted image. Encryption system should

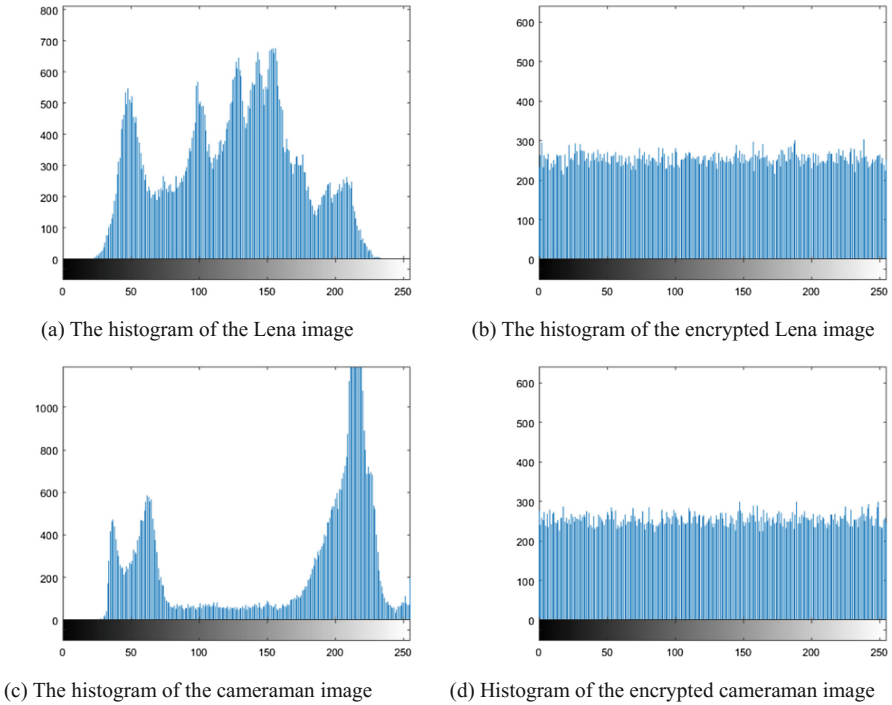


Fig. 7. Histograms of original image and encrypted image

break the original statistical characteristics of original image. The histogram of the original image and decrypted image was shown in Fig. 7. As can be seen from histogram, the pixel distribution of the original image is concentrated in some frequencies, but the pixel distribution of the encrypted image is uniformly, which breaks the histogram statistical rule of the original image. The hacker can't attack the encrypted image with the histogram statistical property of pixel value. Therefore, the algorithm has good anti-statistical analysis ability.

4.3 Correlation Analysis

Whether the correlation between adjacent pixels can be strongly destroyed is one of the indicators to evaluate the quality of the encryption system. The formulas for calculating the correlation coefficients between adjacent pixels are in formula (5):

$$\begin{cases} E(x) = \frac{1}{N} \sum_{i=1}^N x_i \\ D(x) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))^2 \\ Cov(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))(y_i - E(y)) \\ r_{xy} = \frac{Cov(x,y)}{\sqrt{D(x)} * \sqrt{D(y)}} \end{cases} \quad (5)$$

By selecting the same point to compare the correlation coefficient between the original image and the encrypted image, the performance of the encryption system can be judged. As is shown in Table 3, the algorithm has strongly destroyed the correlation between the adjacent pixels, the encryption effect is good.

Table 3. Correlation coefficients

	Horizontal	Vertical	Diagonal
Original image of Lena	0.9673	0.9398	0.9085
Encrypted image of Lena	-0.0046	0.0045	-0.0074
Lena in Ref. [18]	0.0062	0.0052	0.0069
Original image of Cameraman	0.9597	0.9237	0.9145
Encrypted image of Cameraman	0.0022	-0.0176	-0.0003

4.4 Information Entropy Analysis

In 1948, Shannon put forward information entropy with the concept of entropy in thermodynamics and explained the relationship between probability and information redundancy in mathematical language. The information entropy calculation formula of a random information sequence X is in formula (6):

$$H(X) = - \sum_{x \in \chi} P(x_i) \log_2 P(x_i) \quad (6)$$

In the image information, the value range of pixel is 0–255. $P(x_i) = \frac{1}{256}$, The information entropy of an ideal random image is 8. The information entropy of original image Lena is 7.4532. The information entropy of original image Cameraman is 6.9046. The information entropy values of image Lena and encrypted image was shown in Table 4.

Table 4. Information entropy

	Lena	Cameraman
This algorithm	7.9896	7.9896
Ref. [19]	7.9874	7.9874

4.5 Differential Attack Analysis

The anti-differential attack capability of the system is measured by NPCR (number of pixel change rate) and UACI (unified average changing intensity). The calculation formula of NPCR and UACI are in formula (7):

$$\begin{cases} NPCR = \frac{\sum_{i=1}^M \sum_{j=1}^N C(i,j)}{M*N} * 100\% \\ C(i,j) = \begin{cases} 0, P_1(i,j) = P_2(i,j) \\ 1, P_1(i,j) \neq P_2(i,j) \end{cases} \\ UACI = \frac{\sum_{i=1}^M \sum_{j=1}^N |P_1(i,j) - P_2(i,j)|}{255*M*N} * 100\% \end{cases} \quad (7)$$

The ideal value of NPCR value is 100%, and the closer it is to the ideal value, the stronger its ability to resist differential attacks is. The ideal value of UACI is 33%, and the closer it is to the ideal value, the stronger its ability to resist differential attacks is. These two values of this algorithm are shown in Table 5. The results are good than Ref. [19].

Table 5. Simulation of NPCR and UACI

	NPCR	UACI
Encrypted image of Lena	99.60%	28.71%
Encrypted image of Lena in Ref. [19]	99.60%	28.13%
Encrypted image of Cameraman	99.64%	34.79%

5 Conclusion

In this paper, an image encryption algorithm based on dynamic DNA coding and hyper-chaotic system is proposed. The simulation experiment proves that this algorithm has high efficiency, good encryption effect and strong ability to resist attacks. This algorithm can be applied to protect image security.

Acknowledgment. The work for this paper was supported by the National Natural Science Foundation of China (Grant nos. 61602424, 61472371, 61572446, and 61472372), Plan for Scientific Innovation Talent of Henan Province (Grant no. 174100510009), Program for Science and Technology Innovation Talents in Universities of Henan Province (Grant no. 15HAS-TIT019), and Key Scientific Research Projects of Henan High Educational Institution (18A510020).

References

1. Saini, N., Pandey, N., Singh, A.P.: Implementation of security model in cognitive networks. In: International Conference on Communication and Signal Processing, pp. 2055–2058. IEEE (2016)
2. Ye, Y., Wu, N., Zhang, X., Dong, L., Zhou, F.: An optimized design for compact masked AES S-Box based on composite field and common subexpression elimination algorithm. *J. Circuits Syst. Comput.* **27**(11), 1850171 (2018)
3. Seripeariu, L., Frunza, M.D.: A new image encryption algorithm based on inversable functions defined on Galois fields. In: International Symposium on Signals, Circuits and Systems, pp. 243–246. IEEE (2005)
4. Chen, R.J., Lai, Y.T., Lai, J.L.: Architecture design of the re-configurable 2-D von Neumann cellular automata for image encryption application. In: IEEE International Symposium on Circuits and Systems, pp. 3059–3062. IEEE (2005)
5. Bourbakis, N., Alexopoulos, C.: Picture data encryption using scan patterns. *Pattern Recognit.* **25**(6), 567–581 (1992)
6. Zhang, Y., Kang, B.S., Zhang, X.F.: An image encryption algorithm based on chaotic sequences. In: 16th International Conference on Artificial Reality and Telexistence, pp. 221–223. IEEE (2000)
7. Acharya, B., Rath, G.S., Patra, S.K., Panigrahp, S.K.: Novel methods of generating self-invertible matrix for hill cipher algorithm. *Int. J. Secur.* **1**(1), 14–21 (2007)
8. Gehani, A., LaBean, T., Reif, J.: DNA-based cryptography. In: Jonoska, N., Păun, G., Rozenberg, G. (eds.) *Aspects of Molecular Computing*. LNCS, vol. 2950, pp. 167–188. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-24635-0_12
9. Rhouma, R., Belghith, S.: Cryptanalysis of a new image encryption algorithm based on hyper-chaos. *Phys. Lett. A* **372**(38), 5973–5978 (2008)
10. Adleman, L.M.: Molecular computation of solutions to combinatorial problems. *Science* **266** (5187), 1020–1024 (1994)
11. Zhang, X., Zhou, Z., Niu, Y.: An image encryption method based on the feistel network and dynamic DNA encoding. *IEEE Photonics J.* **10**(4), 3901014 (2018)
12. Zhang, X., Zhou, Z., Jiao, Y., Niu, Y., Wang, Y.: A visual cryptography scheme-based DNA microarrays. *Int. J. Perform. Eng.* **14**(2), 334–340 (2018)
13. Cui, G., Liu, Y., Zhang, X., Zhou, Z.: A new image encryption algorithm based on DNA dynamic encoding and hyper-chaotic system. In: He, C., Mo, H., Pan, L., Zhao, Y. (eds.) *BIC-TA 2017. CCIS*, vol. 791, pp. 286–303. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-7179-9_22
14. Wang, X., Yin, Y.L., Yu, H.: Finding collisions in the full SHA-1. *Crypto* **3621**, 17–36 (2005)
15. Zhang, X., Wang, Y., Cui, G., Niu, Y., Xu, J.: Application of a novel IWO to the design of encoding sequences for DNA computing. *Comput. Math Appl.* **57**(11–12), 2001–2008 (2009)

16. Özkaynak, F., Yavuz, S.: Analysis and improvement of a novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system. *Nonlinear Dyn.* **78** (2), 1311–1320 (2014)
17. Gao, T., Chen, G., Chen, Z., Cang, S.: The generation and circuit implementation of a new hyper-chaos based upon Lorenz system. *Phys. Lett. A* **361**(1), 78–86 (2007)
18. Silva-García, V.M., Flores-Carapia, R., Rentería-Márquez, C., Luna-Benoso, B., Aldape-Pérez, M.: Substitution box generation using Chaos: an image encryption application. *Appl. Math. Comput.* **332**, 123–135 (2018)
19. Akhavan, A., Samsudin, A., Akhshani, A.: Cryptanalysis of an image encryption algorithm based on DNA encoding. *Opt. Laser Technol.* **95**, 94–99 (2017)



Application of BFO Based on Path Interaction in Yard Truck Scheduling and Storage Allocation Problem

Lei Liu, Lu Xiao, Lulu Zuo, Jia Liu, and Chen Yang^(✉)

College of Management, Shenzhen University, Shenzhen 518060, China
yangc@szu.edu.cn

Abstract. Nowadays, Yard Truck Scheduling (YTS) and Storage Allocation Problem (SAP) are still two main problems in container terminal operations. Based on the current situation of using double trailer and practical constricts (e.g. container sorting storage, shore bridge and field bridge processing time), this paper proposes a realistic YTS-SAP model (YTSSAP). Additionally, a path interaction bacterial foraging optimization algorithm (PIBFO) is applied to solve the YTSSAP according to the differential tumbling label method and path interaction strategy. In the two strategies, each individual is given a label. If an individual has been employed as one individual's interaction object, it will not be selected by the other bacteria. The population is easy to find the optimal solution using the path interaction strategy. The experiment results illustrate that PIBFO performs superior in dealing with the YTSSAP compared with coevolutionary structure-redesigned-based BFO (CSRBFO), comprehensive learning particle swarm optimizer (CLPSO) and genetic algorithm (GA). CLPSO obtains the worst results in terms of the performance and convergence rate when using it to solve the YTSSAP.

Keywords: Yard truck scheduling · Storage allocation problem
Bacterial foraging optimization · Path interaction strategy

1 Introduction

With the rapid development of global economy, the volume of global trade in goods has also risen sharply. Maritime transport is the backbone of international trade. Due to the advantages of simple tally work, high efficiency of loading and unloading, and low cost of freight transportation, container transportation has become the mainstream of marine transportation in international trade [1].

In order to maintain the advantages of container port dispatching in recent markets, it is necessary to deal with two main problems (i.e. meeting the customer's requirements for operational efficiency and reducing the operating cost of the port). In terms of operational efficiency, fast container handling can significantly reduce the container ship's residence time at the port and the pick-up time of the freight company, which contributes to improve customer satisfaction [2]. For operating costs, reducing the travel time of truck can reduce the cost of fuel consumption and maintenance [3].

Therefore, so many scholars improved the model by reducing the container handling time and the travel time of truck. Considering the time of container handling, a bi-objective Berth Allocation Problem (BAP) model [4] was proposed to minimize port staying time and transfer rate with the consideration of vessel priority. In addition, in order to allocate quay space and service time rationally, the berth allocation problem (BAP) was solved by the combined service of feeder ships and container vessels [5]. Niu [6] improved the yard truck scheduling and storage allocation model and completed two objectives (i.e. minimizing the total delay for all jobs and improving the truck ready time). For the travel time of truck, Al-Dhaheri et al. [7] considered the entire container handling process (e.g., seaside operations and container transfer operations between the quay and the yard) and proposed a stochastic mixed integer-programming model. A simulation-based Genetic Algorithm was applied to deal with the model. In order to reduce the truck waiting time, Gao et al. [8] studied the optimization of double different size crane (DDSC) system with different truck ready time. Moreover, there are two significant issues in container terminal operations, i.e. the yard truck scheduling (YTS) and the storage allocation problem (SAP). To minimize the total time cost of the request delay and the total truck travel time, YTS and SAP were combined into a whole optimization problem (YTS-SAP) in [9]. However, these versions of improved model cannot illustrate some practical problems, such as double trailer and the cross problem of loading job and discharging job. Hence, the area of the integrated YTS-SAP for more general situation is extended in this paper.

Based on the foraging behavior of the *E. coli* bacteria, Passino. [15] proposed a new optimization method in 2002, which was called BFO algorithm. Since then, BFO has been applied to deal with many practical optimization problems, such as multi-area automatic generation control [10], feature selection [11], face recognition [12] and vehicle routing problem [13]. BFO demonstrates the superiority of solving complex non-continuous problems. Therefore, we employ an improved BFO algorithm to solve the yard truck scheduling and storage allocation problem.

In this paper, the current situation of using double trailer is introduced in the yard truck scheduling and storage allocation problem model (YTSSAP). Combined with the classification of the containers, the effectiveness of the parking time of the parking lot and the operation time of the parking lot in a complex environment, the model is designed to minimize the total delayed time and the travel time. Additionally, the bacteria foraging optimization is improved using the differential tumbling label method and path interaction strategy. And the proposed algorithm is applied to solve the YTSSAP compared with coevolutionary structure-redesigned-based BFO (CSRBFO), comprehensive learning particle swarm optimizer (CLPSO) and genetic algorithm (GA).

The paper is organized as follows. Section 2 gives the model of yard truck scheduling and storage allocation problem. Section 3 provides the proposed PIBFO. Section 4 presents experimental results and Sect. 5 concludes the paper.

2 Yard Truck Scheduling and Storage Allocation Problem Model (YTSSAP)

In former research, the YTS-SAP model was introduced in detail in [6]. The traditional yard truck scheduling and storage allocation problem (YTSSAP) only considers the scheduling of single-hung-multi-vehicles under path planning. However, the practical

problems are not simple, including one-dot-double-trailer and double-trailer-pick-up scheduling sub-problems. Considering various constraints, e.g. container sorting storage, Shore Bridge and field bridge processing time, this paper used a new YTSSAP model [14]. In [14], double trailer is introduced to make the model close to the reality. The goal of this model is to minimize the average total delay time and total yard truck travel time (i.e. Eq. (1)). The total truck travel time consists of three parts, i.e. the time from the empty truck to the end of the first hanging job by the best picking method, the driving time of the second hanging operation and the time of returning to the parking lot from their last job. The notations and mathematical model are shown as follows.

Notations:

J^+ is the set of the loading containers with the cardinality of n^+ ; J^- is the set of the discharging containers with the cardinality of n^- ; J is the set of total jobs, i.e. $J = J^+ \cup J^-$. K and K_c denote the set of all storage locations in the yard and the set of C -type storage locations respectively; $[a_i, b_i)$ is the window time for each job. The processing of job i cannot be advanced before the starting time a_i . The due time of job i , b_i can be violated. If b_i is violated, the delayed time d_i will be generated. w_i is the start time of service at job i ; m is the number of yard trucks; R is the set of all routes indexed by r where $|R| = m$; o_i and e_i denote the origin and destination of job i , respectively; pl is the location of the parking lot. $dst_{p,q}$ is the Euclidean distance from location p to location q ; v_{load} and v_{empty} are the velocities of loaded trucks and empty trucks; $\tau_{p,q}$ is the travel time from location p to location q . Pq and Py are processing time of each container for the shore bridge and the field bridge respectively; M is the penalty parameter; α_1 and α_2 are weight coefficients.

In the double-trailer model, there are two pieces of goods on the truck. The first delivery is the first hanging job, and the later delivery is the second hanging job.

t_{ij} is the transit time of the first hanging job j , i.e. the travel time from the due job i to the destination of job j ; h_i is the transit time of the second hanging job; s_{ij} denotes the prepared time from the job i to the job j . The detailed notations are displayed as follows.

$$\begin{aligned}
 t_{ij} &= \begin{cases} t_{oi,ej}, & \text{if job } i \text{ is loading job} \\ t_{oi,\xi k}, & \text{if job } i \text{ is discharging job and allocated to the storage location } k \end{cases} \\
 h_i &= \begin{cases} \tau_{ei-1,ei}, & \text{if job } i \text{ is loading job and the previous job is also loading job} \\ \tau_{ei-1,\xi k}, & \text{if job } i \text{ is discharging job and allocated to the storage location } k, \\ & \text{and the previous job is loading job} \end{cases} \\
 h_i &= \begin{cases} \tau_{\xi D,ei}, & \text{if job } i \text{ is loading job and allocated to the storage location } D, \\ & \text{the previous job is discharging job} \\ \tau_{\xi D,\xi k}, & \text{if job } i \text{ is discharging job and allocated to the storage location } k, \\ & \text{and the previous job is also discharging job and allocated to the storage location } D \end{cases} \\
 s_{ij} &= \begin{cases} \tau_{ei,oj}, & \text{if job } i \text{ is loading job} \\ \tau_{\xi k,oj}, & \text{if job } i \text{ is discharging job and allocated to the storage location } k \end{cases}
 \end{aligned}$$

Additionally, other notations are used in model description.

$$l_j = \begin{cases} j + 1, & \text{if there is job after job } j \\ j, & \text{otherwise} \end{cases}$$

$$y_{ij} = \begin{cases} 1, & \text{if there is job } j \text{ after job } i \\ 0, & \text{otherwise} \end{cases}$$

$$x_{ik} = \begin{cases} 1, & \text{if the job } i \text{ is allocated to the storage location } k \\ 0, & \text{otherwise} \end{cases}$$

$$z_{ij} = \begin{cases} 1, & \text{if the job } i \text{ is the second hanging or } 0, \text{ and next job is } j \\ 0, & \text{otherwise} \end{cases}$$

$$g_i = \begin{cases} 1, & \text{if the job } i \text{ is the second hanging} \\ 0, & \text{otherwise} \end{cases}$$

Mathematical model:

$$\begin{aligned} \text{Minimize } Z = & \alpha_1 \times \sum_{i \in J} d_i + \alpha_2 \times \left(\sum_{i \in j \cup \{0\}, j \in J} \min\{s_{ij} + \tau_{j,l_j} + t_{l_j,j}, s_{il_j} + \tau_{l_j,j} + t_{j,j}\} \times z_{ij} \right) \\ & + \sum_{i \in J} h_i \times g_i + \sum_{i \in J} \tau_{dipl} \times y_{i0} \end{aligned} \tag{1}$$

Subject to:

$$\sum_{i \in J_c^-} x_{ik} = 1, \forall k \in \check{\zeta}_{k_c} \text{ and } c \in \Omega \tag{2}$$

$$\sum_{i \in J_c^-} x_{ik} = 0, \forall k \notin \check{\zeta}_{k_c} \text{ and } c \in \Omega \tag{3}$$

$$\sum_{k \in \check{\zeta}_{k_c}} x_{ik} = 1, \forall i \in J_c^- \text{ and } c \in \Omega \tag{4}$$

$$\sum_{k \in \check{\zeta}_{k_c}} x_{ik} = 0, \forall i \notin J_c^- \text{ and } c \in \Omega \tag{5}$$

$$\sum_{j \in J} y_{ik} = 1, \forall i \in J \tag{6}$$

$$\sum_{i \in J} y_{ij} = 1, \forall j \in J \tag{7}$$

$$\sum_{j \in J} y_{l_{pj}} = m \tag{8}$$

$$\sum_{i \in J} y_{i_p} = m \tag{9}$$

$$w_i \geq a_i, \forall i \in J \tag{10}$$

$$d_i \geq w_i + Pq + h_i \cdot g_i + h_i \cdot \bar{g}_i + Py - b_i, \forall i \in J \tag{11}$$

$$s_{ij} = \sum_{k \in K} \tau_{ei,oi}, \forall i \in J^+ \text{ and } \forall j \in J \tag{12}$$

$$s_{ij} = \sum_{k \in K} \tau_{oi,\xi_i} x_{ik}, \forall i \in J^- \text{ and } \forall j \in J \tag{13}$$

$$\tau_{oi,ei} = dst_{oi,ei} / v_{load}, \forall i \in J^+ \tag{14}$$

$$\tau_{oi,\xi_k} = dst_{oi,\xi_k} / v_{load}, \forall i \in J^- \text{ and } k \in K \tag{15}$$

$$\tau_{ei,oj} = dst_{ei,oj} / v_{empty}, \forall i \in J^+ \text{ and } j \in J \tag{16}$$

$$\tau_{\xi_k,oj} = dst_{\xi_k,oj} / v_{empty}, \forall j \in J \text{ and } \forall k \in K \tag{17}$$

$$w_i \geq 0, t_i \geq 0, h_i \geq 0, d_i \geq 0, \forall i \in J \tag{18}$$

$$s_{ij} \geq 0, \forall i \in J \text{ and } \forall j \in J \tag{19}$$

$$x_{ik}, y_{ij} \in \{0, 1\}, \forall i \in J, j \in J' \text{ and } k \in \xi_k \tag{20}$$

$$w_j + M \times (1 - y_{ij}) \geq w_i + Pq + t_i + Py + s_{ij}, \forall i, j \in J \tag{21}$$

$$w_j + M \times (1 - y_{0j}) \geq w_0 + Pq + Py + s_{0j}, \forall j \in J \tag{22}$$

The object is to minimum the time of average total delay and the time of total yard truck travel by optimizing picking order. Constricts (2–3) guarantee that each type of storage location can only store one container of the same type. Constricts (4–5) limit that each container can only be allocated to the same type storage location. Constricts (6–7) ensure that two jobs handled by the same truck are consistent. Constricts (8–9) means that the track starting from the same starting point (i.e. (parking lot) will eventually return to the parking lot. Constrict (10) limits the start time and Constrict (11) calculate the delay time. Constricts (12–13) calculate the preparation time for the two connected jobs. Constricts (14–17) are used to calculate the travel time of the track between two points. Constricts (18–20) limit the domain of decision variables. Constrict (21) explains the relationship between the start time of the two jobs before and after the connection. Constrict (22) gives the start time of the first job of each track.

3 Methodology

3.1 Bacterial Forage Optimization

In 2002, inspired by the foraging behavior of *E. coli* bacteria, Passino, K.M. proposed a novel bionic simulated evolutionary algorithm—Bacterial Forage Optimization (BFO) [15]. In the process of foraging, the bacteria constantly move to look for food in one direction or possibly change directions according to the circumstance. They survive in line with Darwin's theory of Natural Selection—If food is plentiful, bacteria will split and make copies of themselves; Conversely, if there is a lack of food, those bacteria that can't find food will die.

Abstracted from the afore-mentioned biological behaviors, BFO can be described as three steps: chemotaxis, reproduction, and elimination & dispersal.

Chemotaxis: This motion, which mainly models the foraging behavior, implements a type of optimization where bacteria try to approach to the nutrient concentration. It contains two ways—swimming (move in a particular direction for a period) and tumbling (the bacterium does not have a set direction of movement).

Let $\theta^i(j, k, l)$ denotes the position of i th member in the population of the *S* bacteria at the j th chemotactic step, k th reproduction step, and l th elimination & dispersal event:

$$\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i)\phi^i(j) \quad (23)$$

where $C(i)$ is the step size and $\phi^i(j)$ is a unit length random direction.

Reproduction: After completing the specified chemotactic steps, a reproduction step is implemented. This operation mainly simulates the reproductive process, which follows the theory of survival of the fittest. All bacteria are sorted from large to small in accordance with their fitness values. The $S/2$ bacteria with lower values die and the other $S/2$ bacteria with better values divide into two new bacteria (and the copies are at the same position as their parent). Therefore, the number of bacterial population doesn't change.

Elimination & dispersal: In fact, this behavior is not shown in the biological process of bacteria foraging. The reason that BFO introduces this process is to improve the global searching ability, because the chemotaxis of bacteria can make algorithm fall into local optimum easily. Unlike the reproduction operation, the elimination & dispersal occurs according to a certain migration probability P_{ed} . When a bacterium meets the condition, it will be assigned to a random position on the optimization domain.

3.2 Path Interaction BFO (PIBFO)

In the original BFO, bacteria move randomly in search of food. This searching strategy may bring about some shortcomings, e.g. low convergence rate and poor search performance. Therefore, many scholars have made a lot of improvements to address these deficiencies. In this paper, we mainly proposed two strategies to improve the original BFO—differential tumbling label method and path interaction strategy.

Differential Tumbling Label Method. It was first proposed in this work [16]. The differential information between the optimal bacterium and the population is introduced to guide the tumble direction of the individual:

$$\phi^i(j) = K_1 \cdot (X_{best,G} - X_{r_i,G}) + K_2 \cdot (X_{best,G} - X_{i,G}) \tag{24}$$

where i represents the i th bacterium. $K = \text{random}(D, 1)$, it's a vector between 0 to 1; D represents dimension of the problem; K_1 and K_2 are two different random K . $X_{best,G}$ is the best position of the entire bacteria in the current generation G ; $X_{i,G}$ is the position of the i th bacterium.

It is worthwhile to note that there is difference in X_r between this paper and the previous work [16]. r meant a random integer in the range of $[1, S]$ before. In this paper, each individual is given a label. Here we set:

$$R_G = \{r_{1,G}, r_{2,G}, \dots, r_{s,G}\} \tag{25}$$

R_G is an interaction object sequence of the bacteria. Its set of elements is a sequence of random integers that are not repeated in the range of $[1, S]$. It is generated once per iteration. Therefore, it means that the i th bacterium has a sole interaction object (bacterium) in the generation G .

Path Interaction Strategy. After differential tumbling, the path interaction strategy is introduced to change the motion track of bacteria:

$$x_{i,Ns,G}^j = \begin{cases} x_{r_i,G,Ns,G}^j & \text{if } (rand_i^j \leq CR) \\ x_{i,Ns,G}^j & \text{otherwise} \end{cases} \quad j = 1, 2, \dots, D \tag{26}$$

where CR is a constant value between 0 to 1; $rand_i^j$ is a random number between 0 to 1. $X_{i,Ns,G} = \{x_{i,Ns,G}^1, x_{i,Ns,G}^2, \dots, x_{i,Ns,G}^D\}$ is the position vector of i th bacterium at its Ns -th swimming step.

After generating the cruising path, each bacterium may change one dimension of its position vector if it meets the condition (i.e. $rand_i^j \leq CR$). Meanwhile, this transformation is random. Therefore, by replacing the value of one dimension from its interaction object, the bacterium can explore round and fully instead of swimming straightly. An example that two bacteria interact in two-dimension space is shown in Fig. 1.

As shown in Fig. 1, the solid and dashed lines represent the two-dimension space of two bacteria. If there is no information interaction between the two bacteria, they will keep their original direction (i.e. the left of Fig. 1). If the two bacteria have path information interaction, the direction of the two bacteria will be changed (i.e. the right of Fig. 1). Obviously, this strategy can make individuals search for food in a reasonable domain and make them easier to find the optimal solution.

Differential tumbling label and path interaction strategies are employed in the chemotaxis step. In PIBFO, reproduction, and elimination & dispersal steps are also

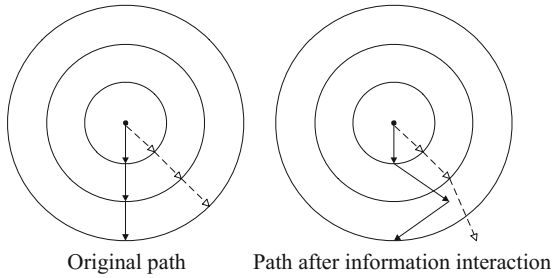


Fig. 1. The motion tracks of two bacteria in the strategy of path interaction

used to update the generation. The flowchart of the original BFO and PIBFO are compared in Fig. 2. Though the chemotaxis structure of PIBFO is complex, the strategies of differential tumbling label and path interaction will make it easy to jump out the local optimal solution, which significantly improves drawback of the original algorithm.

3.3 The Encoding Method and the Fitness Evaluation

The problem of YTSSAP is essentially composed of three sub-scheduling problems, including job scheduling order sub-problem, yard truck scheduling sub-question and containers allocation sub-problem [6, 14]. The encoding method is shown as follows referred to [14]. Each problem is denoted as $[N^+, N^-, L, M, T]$ which is characterized by the number of loading jobs (N^+), the number of discharging jobs (N^-), the number of storage locations (L), the number of trucks (M) and the number of container types (T). Each bacterium can be seen as a scheduling scheme, i.e. Eq. (27).

$$\pi_i = [\pi_{i1}, \pi_{i2}, \dots, \pi_{i(N^+ + N^- + L + M)}] \tag{27}$$

The fitness function is described as Eq. (28), where ST_i $i = 1, 2, \dots, k$ is the i^{th} constraint and k is the number of constraints. If the i^{th} constraint is satisfied, the value $\overline{bool(ST_i)}$ is zero. Otherwise, it will be infinity. If constraints (Eqs. (2)–(22)) are satisfied, it means that $\sum_{i=1}^k \overline{bool(ST_i)}$ is zero. The scheduling scheme is a candidate scheme. Otherwise, the scheduling scheme should be updated.

$$fit = MinimizeZ + \sum_{i=1}^k \overline{bool(ST_i)} \tag{28}$$

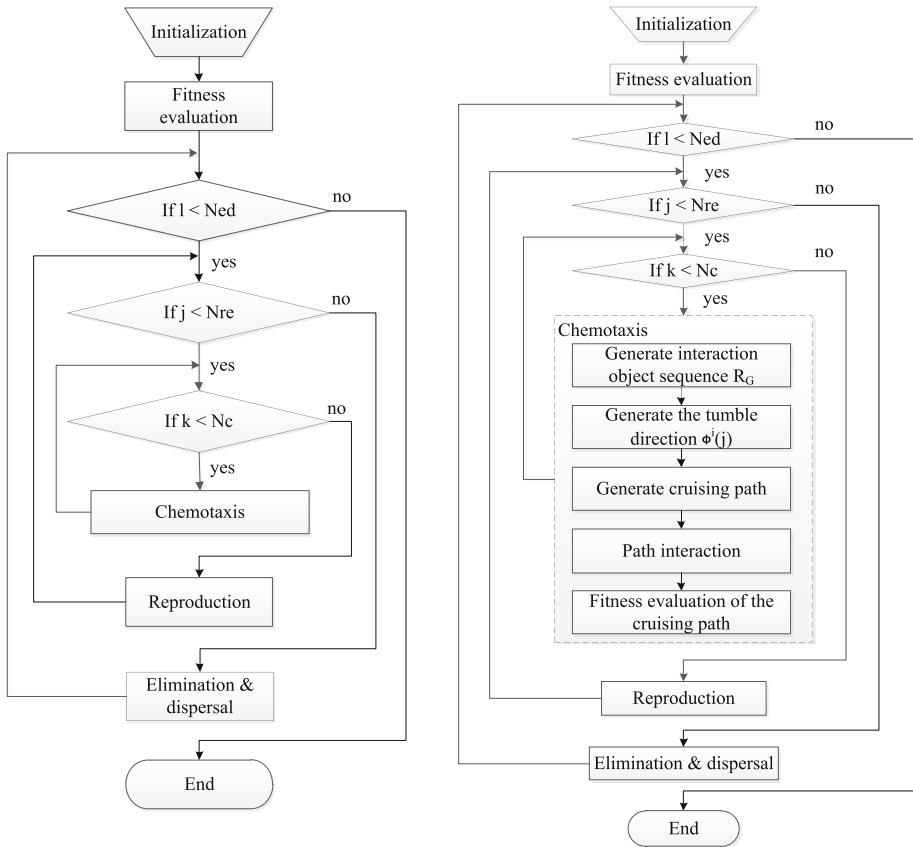


Fig. 2. The flowchart of BFO and PIBFO

4 Experimental Results

4.1 Parameter Settings

Compared with a new improved BFO algorithm—CSRBFO and two well-known optimization algorithms—CLPSO and GA, the performance of PIBFO was evaluated on six test problems. The same parameters of all algorithms are described as follows: The number of swarm $S = 100$ and the maximum number of iteration is 100. Other parameter settings of PIBFO are listed as follows: Nre (reproduction times) = 2; Ns (swimming times) = 5; C (size of swimming step) = 0.5; Ted (migration times) = 4; Nc (chemotaxis times) = Ted/Nre ; Ped (migration probability) = 0.1; $CR = 0.05$. The parameters involved in CSRBFO were set to be the same as literature [17]. The parameters of CLPSO and GA are referred to [18] and [19], respectively. Additionally, Partial parameters of the model parameters referred to [14] are shown as follows. The weight coefficients $\alpha_1 = 0.1$ and $\alpha_2 = 0.9$. The velocity of loaded truck v_{load} is 5 m/s and the velocity of empty truck v_{empty} is 11 m/s.

4.2 Results and Discussion

Table 1 presents the mean values and variances of the results. Figure 3 shows the convergence characteristics in terms of the best fitness value of PIBFO and three compared algorithms on different scale problems.

According to Table 1, PIBFO surpasses all other algorithms for each instance in terms of convergence performance. The results of CLPSO are the worst when using it to solve YTSSAP. The performance of CSRBF0 is better than GA on instance 1, 3, 4 and 6. Therefore, compared to a new improved BFO algorithm and two well-known optimization algorithms, our proposed algorithm—PIBFO is more efficient in addressing YTSSAP.

Table 1. Comparison with different algorithms for six test instances

No.	[N ⁺ , N ⁻ , L, M, T]	Results	PIBFO	CSRBF0	CLPSO	GA
1	[40, 20, 20, 1, 1]	Mean	3.1879E+04	3.3828E+04	3.5193E+04	3.4888E+04
		Var	1.9850E+05	3.2690E+05	2.0429E+05	2.7824E+05
2	[40, 30, 30, 3, 2]	Mean	1.1509E+04	1.3039E+04	1.3536E+04	1.2895E+04
		Var	1.3017E+05	5.7112E+04	1.0163E+05	9.7787E+04
3	[80, 40, 40, 4, 2]	Mean	3.8329E+04	4.0331E+04	4.2700E+04	4.0686E+04
		Var	1.3264E+06	4.0760E+05	6.0638E+05	5.6713E+05
4	[100, 50, 50, 5, 2]	Mean	4.9544E+04	5.0986E+04	5.3581E+04	5.1181E+04
		Var	5.9786E+05	7.0369E+05	3.6917E+05	7.0282E+05
5	[200, 200, 200, 10, 4]	Mean	8.6091E+04	8.6905E+04	9.5030E+04	8.5967E+04
		Var	6.5104E+05	3.9787E+05	2.4298E+06	1.1575E+06
6	[300, 100, 200, 10, 4]	Mean	2.2016E+05	2.2075E+05	2.3718E+05	2.2128E+05
		Var	1.5591E+06	3.5787E+06	6.4030E+06	5.7848E+06

From Fig. 3, it can be seen that PIBFO performs much better than other three algorithms (especially CLPSO) on instance 1, 2, 3 and 4. In addition, the instances are divided into three scales: small (1 and 2), medium (3 and 4) and large (5 and 6) by their complexity. The variable dimensions of six instances are 61, 73, 124, 155, 410, and 510 in turn. Therefore, the complexity of scale problems increases in multiple. As the dimension increases to more than 400 (e.g. instance 5 and 6), all the algorithms are difficult to obtain better results. Though the performances of PIBFO are not very prominent on large-scale problems, it still can achieve better results than the other compared algorithms. Furthermore, the solutions obtained by CSRBF0 and GA can comparable with each other, and CLPSO get worst results in terms of performance and convergence rate when solving YTSSAP.

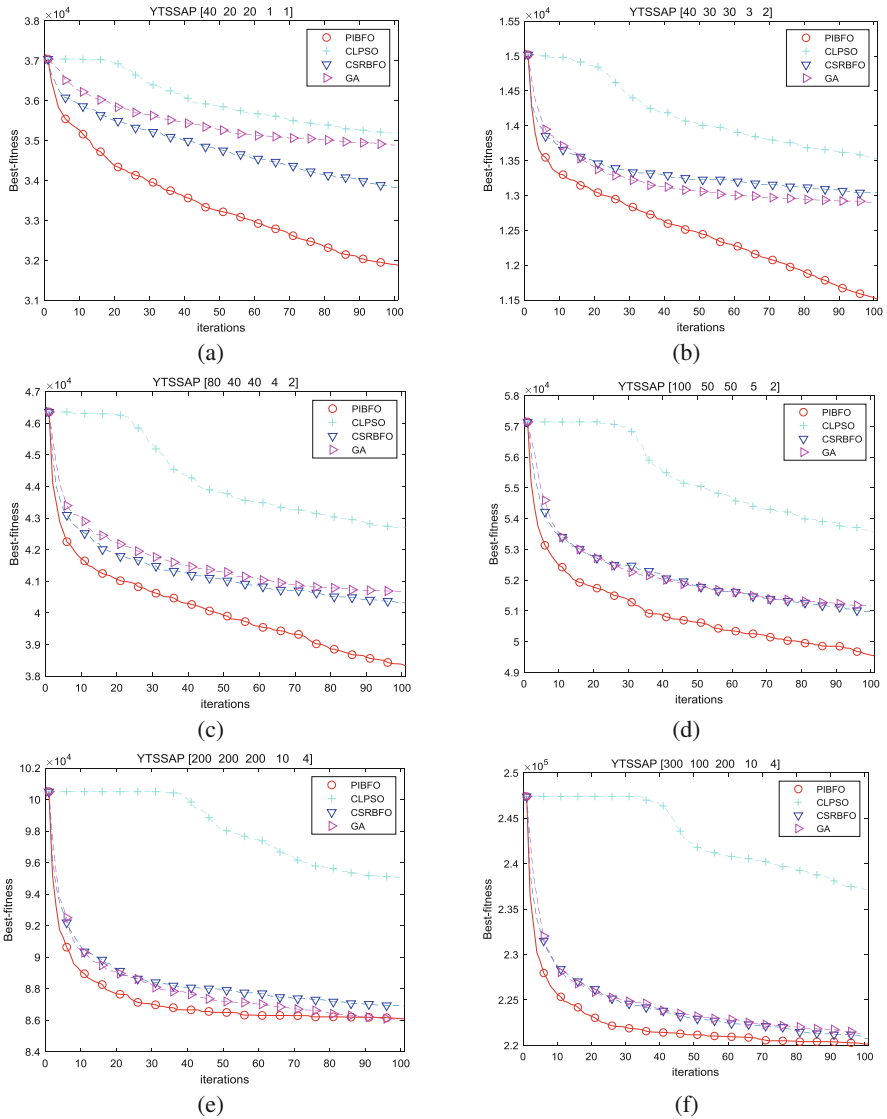


Fig. 3. The convergence characteristics comparisons for different algorithms on YTSSAP

5 Conclusion and Future Work

In this paper, we propose a “double trailer” based Yard Truck Scheduling and Storage Allocation Problem (YTSSAP) to construct a more reasonable model. Based on analyzing the process of port production, the current situation that port transportation concludes double trailer is introduced into the dispatching problem of container trucks. Therefore, our model is closer to the reality of port transportation. It also provides

theoretical support to efficiency improvement and cost reduction of the container port by introducing the “double trailer”.

In addition, we design two strategies—differential tumbling label and path interaction to improve the original BFO. These new strategies can make our proposed algorithm—PIBFO reflect better characteristics of optimization, which can be observed in the experimental results.

Finally, we apply PIBFO to solve the mathematical model of YTSSAP. In this paper, one new improved BFO algorithm—CSRBFO and two well-known optimization algorithms—CLPSO and GA are compared with PIBFO. When considering the solution quality and convergence rate comprehensively, PIBFO is the best choice. Therefore, the results show the possibility of application of PIBFO in the field of YTS-SAP. In the future, our YTSSAP model will consider more realistic factors, such as multi-layers of containers located on the yard, mixed scheduling and so on.

Acknowledgment. This work is partially supported by The National Natural Science Foundation of China (Grants No. 61472257, 71701134), Natural Science Foundation of Guangdong Province (2016A030310074, 2017A030310427), The HD Video R & D Platform for Intelligent Analysis and Processing in Guangdong Engineering Technology Research Centre of Colleges and Universities (No.GCZX-A1409), The Postgraduate Innovation Development Fund Project of Shenzhen University (PIDFP-RW2018015).

References

1. Stahlbock, R., Voß, S.: Operations research at container terminals: a literature update. *OR Spectr.* **30**, 1–52 (2008)
2. Steenken, D., Voß, S., Stahlbock, R.: Container terminal operation and operations research - a classification and literature review. *OR Spectr.* **26**(1), 3–49 (2004)
3. Carlo, H.J., Vis, I.F.A., Roodbergen, K.J.: Transport operations in container terminals: literature overview, trends, research directions and classification scheme. *Eur. J. Oper. Res.* **236**(1), 1–13 (2014)
4. Ma, H.L., Chan, F.T.S., Chung, S.H., Niu, B.: Minimizing port staying time for container terminal with position based handling time. In: 2013 IEEE International Conference on Industrial Engineering and Engineering Management, pp. 1339–1343, Bangkok, Thailand (2014)
5. Emde, S., Boysen, N.: Berth allocation in container terminals that service feeder ships and deep-sea vessels. *J. Oper. Res. Soc.* **67**(4), 551–563 (2016)
6. Niu, B., Xie, T., Tan, L., Bi, Y., Wang, Z.: Swarm intelligence algorithms for yard truck scheduling and storage allocation problems. *Neurocomputing* **188**, 284–293 (2016)
7. Al-Dhaheri, N., Jebali, A., Diabat, A.: A simulation-based genetic algorithm approach for the quay crane scheduling under uncertainty. *Simul. Model. Pract. Theory* **66**, 122–138 (2016)
8. Gao, X.M., Yang, Y., Wu, Z.H.: Genetic algorithm for scheduling double different size crane system with different truck ready times. In: 2016 IEEE International Conference on Industrial Engineering and Engineering Management, pp. 447–451, Tehran, Iran (2016)
9. Zhang, F., Li, L., Liu, J., Chu, X.: Artificial Bee colony optimization for yard truck scheduling and storage allocation problem. In: Huang, D.-S., Jo, K.-H. (eds.) *ICIC 2016*. LNCS, vol. 9772, pp. 908–917. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-42294-7_81

10. Nanda, J., Mishra, S., Saikia, L.C.: Maiden application of bacterial foraging-based optimization technique in multiarea automatic generation control. *IEEE Trans. Power Syst.* **24**(2), 602–609 (2009)
11. Chen, Y.P., Li, Y., Wang, G., Zheng, Y.F., Xu, Q., Fan, J.H., et al.: A novel bacterial foraging optimization algorithm for feature selection. *Expert Syst. Appl. Int. J.* **83**, 1–17 (2017)
12. Panda, R., Naik, M.K.: A novel adaptive crossover bacterial foraging optimization algorithm for linear discriminant analysis based face recognition. *Appl. Soft Comput.* **30**, 722–736 (2015)
13. Tan, L., Lin, F., Wang, H.: Adaptive comprehensive learning bacterial foraging optimization and its application on vehicle routing problem with time windows. *Nat. Comput.* **151**(3), 1208–1215 (2015)
14. Liu, J.: A study on yard truck scheduling and storage allocation using modified brain storm optimization algorithms. Unpublished Master's thesis. Shenzhen University, Shenzhen, China (2018)
15. Passino, K.M.: Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Syst.* **22**(3), 52–67 (2002)
16. Xiao, L., Chen, J., Zuo, L., Wang, H., Tan, L.: Differential structure-redesigned-based bacterial foraging optimization. In: Tan, Y., Shi, Y., Tang, Q. (eds.) *ICSI 2018*. LNCS, vol. 10941, pp. 295–303. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93815-8_29
17. Niu, B., Liu, J., Wu, T., Chu, X.H., Wang, Z.X., Liu, Y.M.: Coevolutionary structure-redesigned-based bacterial foraging optimization. *IEEE/ACM Trans. Comput. Biol. Bioinf.* (2017). <https://doi.org/10.1109/TCBB.2017.2742946>
18. Liang, J.J., Qin, A.K., Suganthan, P.N., Baskar, S.: Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Trans. Evol. Comput.* **10**(3), 281–295 (2006)
19. El-Abd, M.: Performance assessment of foraging algorithms vs. evolutionary algorithms. *Inf. Sci.* **182**(1), 243–263 (2012)



Research on Optimization of Warehouse Allocation Problem Based on Improved Genetic Algorithm

Ding Ning, Wang Li^(✉), Teng Wei, and Zhao Yue

School of Electronic and Information Engineering,
University of Science and Technology Liaoning, Anshan 114051, China
Wangli19966@163.com

Abstract. In this paper, a mathematical model is established for the distribution of cargo warehouses in a three-dimensional warehouse. This model is a multi-objective optimization problem which considers three factors: shelf stability, time of delivery, and association rules among goods. This paper uses the simulated annealing algorithm to solve the problem that the traditional genetic algorithm “easily falls into the local optimal solution” in the search problem, and combines the improved genetic algorithm and the objective function to distribute the goods in the goods. The experimental results show that the improved genetic algorithm is better than the traditional genetic algorithm in time and the optimization of the objective function, which improves the efficiency of the whole warehouse and reduces the operation cost of the enterprise.

Keywords: Location Assignment · Genetic algorithm
Simulated annealing algorithm

1 Introduction

Storage Location Assignment Problem means to ensure the maximum utilization rate of warehouse space according to a certain strategy and principle to reduce the storage cost. According to the existing distribution state of the goods, the goods in the warehouse are placed in the corresponding position [1]. The optimization of warehousing allocation is to plan the cargo space, take into account the characteristics of the goods, the actual demand, the nature of the storage equipment, and so on, to improve the efficiency of the automated warehouse and reduce the cost of storage [2, 3]. This paper mainly considers the shelf stability, the outgoing time and the principle of association between the goods, and uses the improved genetic algorithm which is improved by the simulated annealing algorithm, and carries out the distribution of different kinds of goods in the warehouse.

2 Models of Storage Location Assignment

2.1 Constraint Condition

The experimental data needed are randomly generated and fixed freight allocation strategy is adopted. To restrict each cargo location, only one item must be placed, and

the same goods can be stored on multiple cargo locations. That is, only one kind of goods is stored in a certain cargo position, which cannot be stored in the position of two or more than two goods. If the cargo position is not full, the goods will be loaded into the position. If the cargo position is full, then the corresponding optimization of the cargo position is carried out with the algorithm, and a new position is found. The quantity of goods stored in each cargo location must be integer.

2.2 Model Hypothesis

Based on the premise of the above constraints, the optimization model of this paper is assumed as follows:

1. The shape and volume of all goods placed on the shelf are standard, and the quality is evenly distributed.
2. In each simulation process, the turnover rate, quality and type of goods are known (randomly generated before experiment).
3. Only one kind of goods can be stored in each position.
4. The stereoscopic warehouse studied in this paper uses the way of single port and out warehouse.
5. The length and width of each position are set to 10.
6. The time of starting and braking of stacking machines is not counted, and the time spent on placing goods to corresponding cargo locations is also negligible.
7. The shelf length (L) and width (H) of this paper are all set to 10. The weight of the cargo location can be set to 1.5 times the maximum quality of the goods produced randomly.

2.3 The Establishment of a Mathematical Model

This topic mainly studies the distribution of warehouses in warehouses. The mathematical model is based on three principles of distribution of goods and bits. This paper considers the stability of the shelves, the time of taking the goods, and the relevance between the goods.

The shelf of the three-dimensional warehouse in this article is p layer q column, i represents the column number, j representative layer number, the shelves near the ground are first layers, the most close to the entry and exit table are first columns, using (i, j) to represent the corresponding position coordinates of the goods. In the entire shelf system, the correlation coefficient r between the goods is generated randomly, the goods are randomly generated, the goods are randomly generated, the goods are randomly generated, goods, The quality m is generated randomly for the algorithm, and the center of gravity of each cargo location is located at the center of the cargo location. Under normal circumstances, the maximum number of goods and the corresponding weight is inversely proportional to the relationship. Figure 1 is a schematic diagram of a three-dimensional warehouse shelf.

Shelf Based Stability. In the management of the warehouse, the stability of the shelf is the first problem to be considered. Any shelf has its own maximum bearing range. In the process of placing the goods on the shelf, the principle of “upper and lower weight”

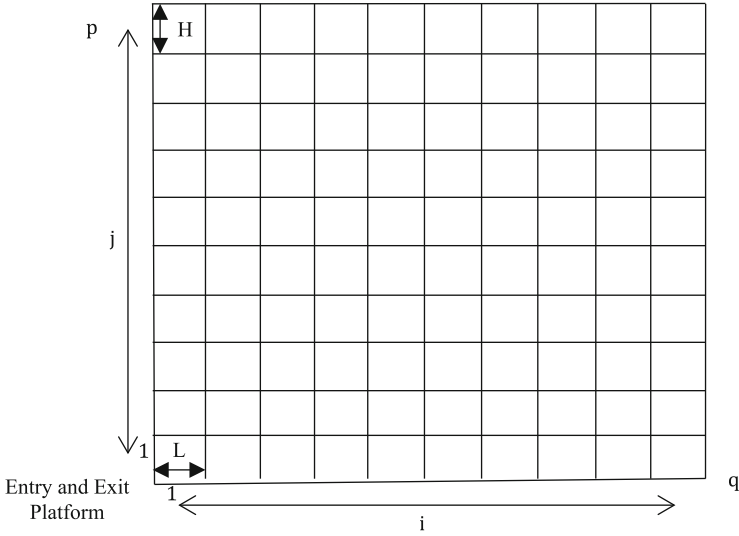


Fig. 1. Diagram of warehouse shelf

should be followed to ensure the stability of the shelf and the uniform shelf life. The quality of goods ensures the safety and economic [4] of the whole warehouse. The mathematical model based on the stability is as follows:

$$\min S = \sum_{i=1}^p \sum_{j=1}^q m_{ij} \cdot n_{ij} \cdot j \cdot x_{ij} \tag{1}$$

Among them, S represents the stability, m_{ij} is the quality of the goods stored in the (i, j) cargo position; n_{ij} is the number of goods stored on the (i, j) position; x_{ij} 's value is 0 or 1. When x_{ij} is 0, there is no goods on the (i, j) cargo position, not counting, and when x_{ij} is 1, there are goods on behalf of the goods. The mass of the quality m and the number of n can get the total weight of the corresponding goods, and then multiplied with the corresponding height j , the relative stability S can be obtained. The smaller the S the more the stability is, the more the stability of the shelf is met.

Based on the Shortest Delivery Time. How to shorten the walking time of the stacker is the key to improve the operating efficiency of the warehouse, which is mainly controlled by the path distance of the stacker and the speed control of the stacker and pallet [5]. If the location coordinates of a cargo location are (i, j) , the following is a mathematical model based on the shortest delivery time:

$$\min T = \sum_{i=1}^p \sum_{j=1}^q t_{ij} \cdot x_{ij} \tag{2}$$

$$t_{ij} = \max\left(\frac{(i - 0.5) \cdot L}{V_x}, \frac{(j - 0.5) \cdot H}{V_y}\right) \tag{3}$$

T indicates the time required by the stacker to take out the coordinates of the cargo in (i, j) position. In t_{ij} , the L indicates the length of the cargo position, the H indicates the height of the cargo position, and V_x indicates the speed of the stacker in the horizontal direction. V_y represents the speed of the palletizing tray in the vertical direction of the rack, and (i - 0.5) represents the cargo at the X axis. The product of the center of gravity and the unit length of the shelf represents the distance from the X table to the outlet, and the Y axis is the same. When the stacker is running horizontally, the tray will also work vertically and vertically, so the maximum time required for t to take the X axis and the Y axis is obtained. T multiplies with the corresponding cargo location information x_{ij} , that is, the time of delivery is T.

Association Rules Based on Association Rules. In the actual problem of distribution of goods, A and B are often encountered at the same time. Therefore, the idea of association rules is introduced in this paper, which is placed in a similar position to facilitate the picking behavior of the stacker, thus improving the operation efficiency of the warehouse whole. The mathematical model based on the association rules is as follows:

$$\min D = \sum_{a=1}^N \sum_{b=1}^N r_{ab} \cdot d_{ab} \tag{4}$$

Among them, D is calculated by correlation coefficient and placement distance between different kinds of goods. N is the number of goods, r_{ab} represents the correlation coefficient between goods a and B, which can be obtained by correlation coefficient matrix, d_{ab} represents the distance between cargo a and cargo B, and d_{ab} is calculated by the Pythagorean Theorem, and the coordinates of two different cargo positions are calculated. The corresponding distance can be obtained by making a difference. The smaller the D, the closer the goods on behalf of the associated goods are placed.

2.3.1 Objective Function

The objective function of this paper consists of three parts: shelf stability, picking time and association rules. The final objective function F is the sum of the three.

$$F = \min S + \min T + \min D \tag{5}$$

3 Improved Genetic Algorithm

In this paper, the idea of genetic algorithm and simulated annealing algorithm is integrated, and a new optimization algorithm is obtained. This algorithm not only preserves the advantages of the genetic algorithm, but also overcomes the disadvantage

of the genetic algorithm being easily trapped in the local optimal solution. The algorithm has the characteristics of strong search ability, wide application range, strong expansibility and a certain probability to jump out of the local optimal solution.

The basic process of improved genetic algorithm: the initial population is generated, the coding method is selected to code the chromosome, the parameters are set, the initial solution is set, the new solution set is optimized by the genetic algorithm, then the genetic algorithm is improved by the simulated annealing algorithm, when the algorithm meets the termination of the simulated annealing algorithm. When the condition is terminated, the whole algorithm is terminated and the optimized result is obtained. The specific process is shown in Fig. 2.

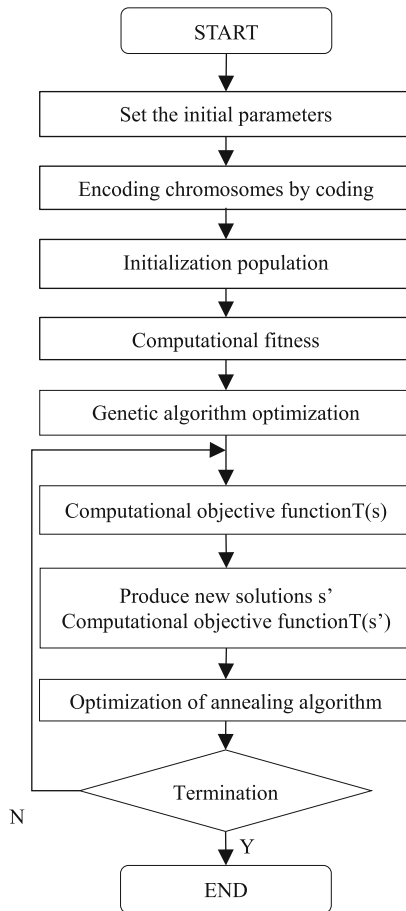


Fig. 2. Flow chart of improved genetic algorithm

4 Analogue Simulation

A total of three purchase operations are simulated, MATLAB software is used to simulate the distribution of cargo space. The fitness curve is compared with the traditional genetic algorithm. In order to facilitate the study, the speed of the X and Y axes of the stacker is set to 0.5. Only one container is sampled to simulate the storage space. The storage space of the container is set to 10*10, and the length and width of the container are all set to 1 m. It is assumed that there are 6 kinds of goods that need to be stored. The improved genetic algorithm introduced in this paper has the size of the initial population, which is set to 200. The evolutionary algebra is the terminating condition, which is set to 1000. The cross probability is usually larger to ensure the diversity of the population, often between 0.4 and 0.9. The cross probability is $P_c = 0.7$; the mutation probability is usually the same, avoid the degradation of the ambassador algorithm, from 0.001 to 0.2. Here, the mutation probability is $P_m = 0.05$, the initial temperature is 100, and the temperature reduction parameter is set to 0.98. There are six kinds of goods known in this article. A total of 3 purchases are simulated, and each incoming data is as follows:

- First purchase, goods 4, goods 2, single cargo weight 12.
- Second purchase, goods 3, goods 5, single cargo weight 27.
- Third purchase, goods 3, goods 6, single cargo weight 27.

4.1 First Time Simulation

The first time to carry out the cargo storage operation, the random distribution of the goods position, fourth kinds of goods optimization of the storage allocation before the optimization as shown in Fig. 3, the improved genetic algorithm optimization of the distribution of cargo map as shown in Fig. 4.

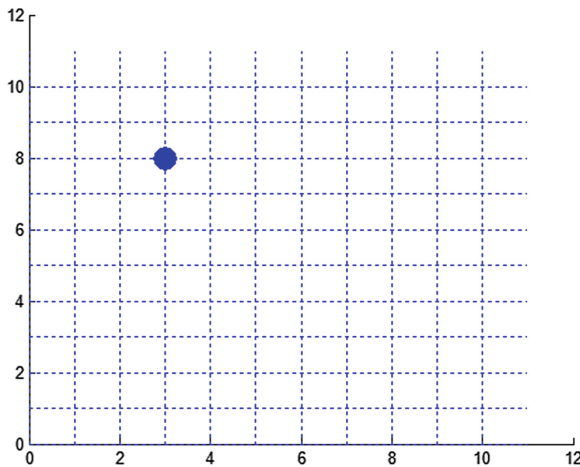


Fig. 3. The first time storage distribution before optimization

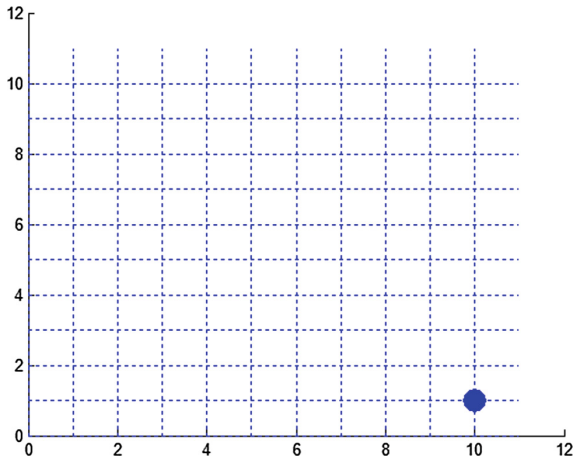


Fig. 4. The first time storage distribution after optimization

After the first cargo was put into storage, the improved genetic algorithm was used to optimize it. The objective function values before and after optimization were calculated respectively, and the results were as follows:

After optimization, $S = 168.000000$, $D = 0.053698$, $T = 410.000000$

Pre optimization $S = 770.000000$, $D = 0.109875$, $T = 1190.000000$

It can be seen from the calculation results that S , T and D have been reduced. As a result, after optimization, the stability of the shelves has been improved and the goods with greater relevance are placed near. At the same time, the allocation of the optimized warehouses will also greatly shorten the time of taking the goods and facilitate the subsequent outgoing operation of the goods.

As shown in Fig. 5, the red line in the graph is the fitness curve of the traditional genetic algorithm, the green line is an improved genetic algorithm curve, and we can get the conclusion from the curve. The convergence rate of the fitness curve obtained by the traditional genetic algorithm is not as good as the improved genetic algorithm, although the last two converges to the same one. When the improved genetic algorithm is used to optimize the convergence speed, the speed of convergence is obviously improved. Thus, we can see that the improved genetic algorithm can improve the efficiency of the storage system when the actual distribution of goods is allocated to the warehouse.

4.2 Second Times Simulation

When the goods are carried out in the second operation, the kind of goods is randomly generated. This simulation simulated random generation of third kinds of goods, which is based on the result of the optimization of the first storage allocation. The location of the random distribution of the goods after the entry is shown as shown in Fig. 6, and

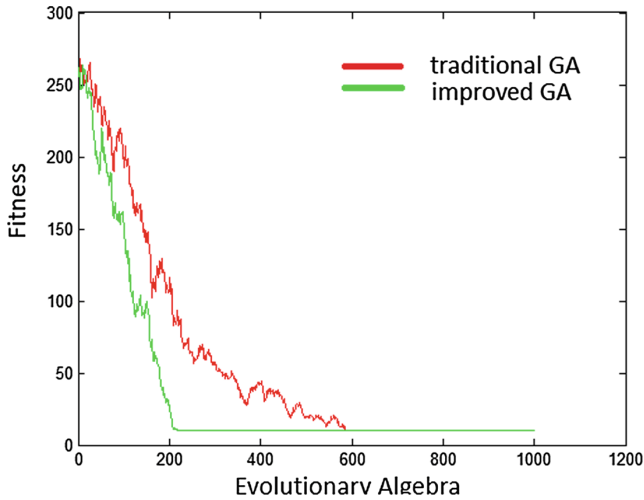


Fig. 5. The first time comparison chart of two optimization algorithms fitness curve (Color figure online)

the improved genetic algorithm is optimized. The distribution map of the cargo space is shown in Fig. 7.

After the second goods were put into storage, the improved genetic algorithm was used to optimize the objective function values before and after optimization. The results were as follows:

After optimization, $S = 27.000000$, $D = 0.111294$, $T = 200.000000$

Pre optimization $S = 1188.000000$, $D = 0.557303$, $T = 740.000000$

It can be seen from the calculation results that S , D and T have been reduced. It is proved that after optimization, the stability of the shelves is improved and the goods with greater relevance are placed near. At the same time, the distribution of the optimized cargo space also greatly shortens the time of picking up the goods and facilitates the subsequent outgoing operation of the goods.

As shown in Fig. 8, the red line in the graph is the fitness convergence curve of the traditional genetic algorithm and the green line is the convergence curve of the genetic algorithm improved by the simulated annealing algorithm. It can be seen from the image that the convergence speed of the two algorithms after the second operation is basically the same, but the traditional genetic algorithm is in the evolutionary generation of the population. When the number reaches 300 generations, it will no longer converge and fall into the state of "local optimal solution". When the genetic algorithm improved by simulated annealing algorithm is optimized, the convergence is stopped when the evolutionary algebra reaches about 400 generations, and the local optimal state has been leapt out and the desired effect has been achieved. It can be seen that the improved genetic algorithm can achieve better results and better location allocation.

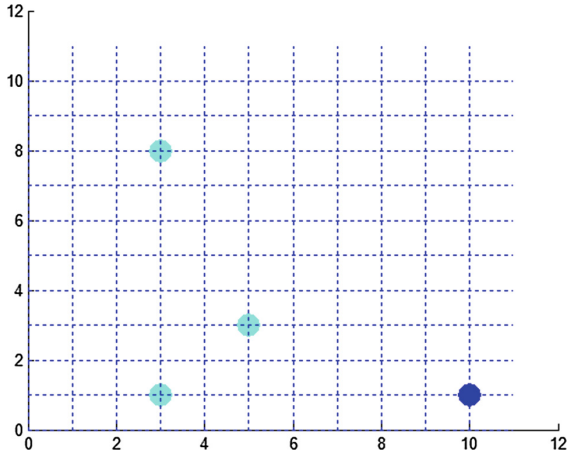


Fig. 6. The second time storage distribution before optimization

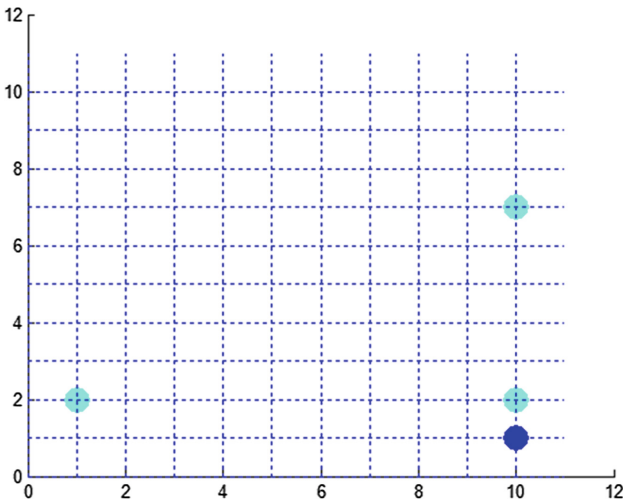


Fig. 7. The second time storage distribution after optimization

4.3 Three Times Simulation

When the goods are carried out in the third operation, the kind of goods is randomly generated. This simulation simulated random generation of third kinds of goods, and on the basis of the optimization of the distribution of the second goods after the distribution of the goods, the location of the random distribution of the goods is shown in Fig. 9. The improved genetic algorithm is superior to the improved genetic algorithm. The assigned map is shown in Fig. 10.

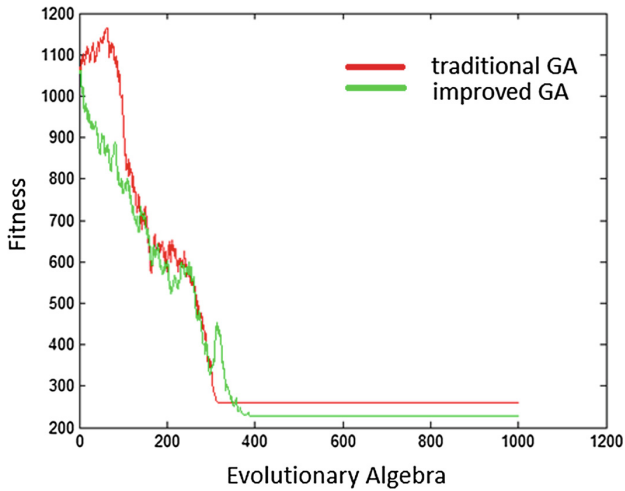


Fig. 8. The second time comparison chart of two optimization algorithms fitness curve (Color figure online)

After the third goods were put into storage, the improved genetic algorithm was used to optimize the objective function values before and after optimization. The results were as follows:

After optimization, $S = 81.000000$, $D = 20.515248$, $T = 550.000000$
 Pre optimization $S = 1296.000000$, $D = 52.836883$, $T = 1050.000000$

From the calculation results, it can be seen that S, D and T have different degrees of reduction. The experimental results show that after optimization, the stability of the shelves is improved and the goods with greater relevance are placed near. At the same

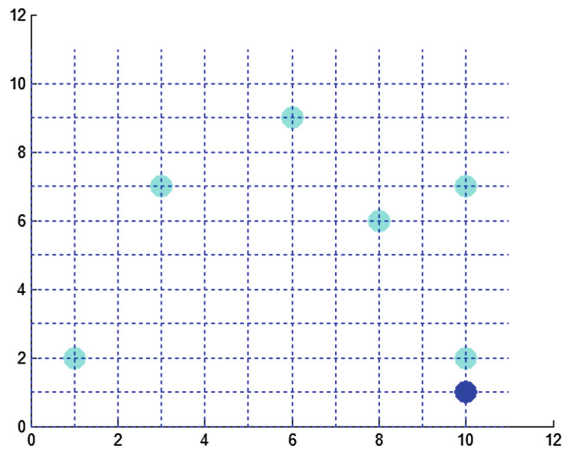


Fig. 9. The third time storage distribution before optimization

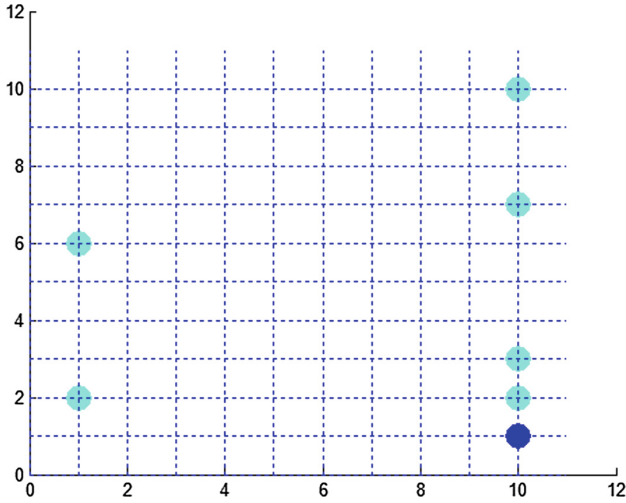


Fig. 10. The third time storage distribution after optimization

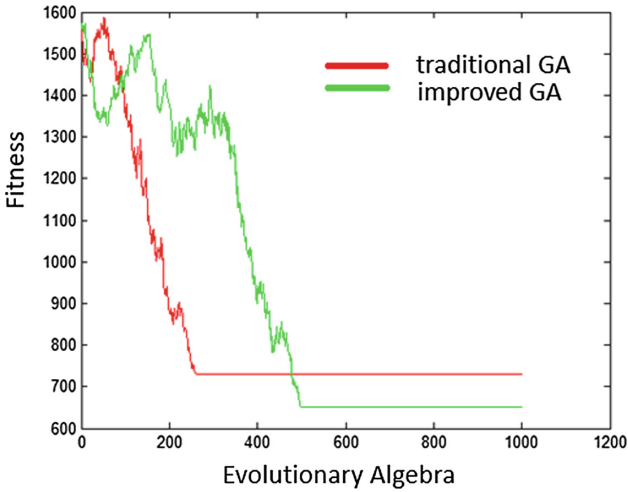


Fig. 11. The third time comparison chart of two optimization algorithms fitness curve at the end of the experiment (Color figure online)

time, the allocation of the stored goods and places after the optimization has also greatly shortened the time of taking the goods.

As shown in Fig. 11, the red line in the graph is the fitness convergence curve of the traditional genetic algorithm, and the green line is the convergence curve of the genetic algorithm improved by the simulated annealing algorithm. It can be seen from the image that the traditional genetic algorithm quickly presents the convergence trend

when the population has just evolved at the beginning of the third operation. When evolutionary algebra reaches about 250 generations, it will no longer converge and fall into the state of “local optimal solution”. When the genetic algorithm improved by simulated annealing algorithm is optimized, it converges quickly when the evolutionary algebra reaches about 300 generations, and does not converge when the evolutionary algebra reaches about 500 generations. The image shows, the optimized algorithm jumps out. The local optimal state overcomes the shortcoming of the traditional genetic algorithm which converges quickly at the beginning and achieves the desired effect. It can be seen from the image that the improved genetic algorithm can get better results for the distribution of goods in storage and make the allocation of freight more reasonable.

5 Conclusion

After three simulation experiments, simulates the operation of three cargo warehousing. The S, T and D values before and after the optimization are obtained respectively. The results of the simulated S value after the simulation show that the optimized storage position allocation scheme can make the shelf life more stable and ensure the shelf life. The safety and economy of the goods are put in order to ensure the safety and economy of the whole warehouse. The comparison of the results before and after the T value shows that the optimized distribution scheme can shorten the walking time of the stacker when the goods are out of the warehouse, and thus improve the operating efficiency of the whole warehouse. The comparison of the results of the D value before and after shows that the goods are optimized by the distribution of the goods. After that, the goods with large relative coefficient are placed in a similar position, which can facilitate the subsequent picking behavior of the stacker and improve the overall efficiency of the warehouse. Simulation shows that the increase of cargo location and the increase of cargo types will increase the running time of the algorithm.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (Project Number: 71472081).

References

1. Enrico, M., Giacomo, N., Dimitri, T.: Optimizing allocation in a warehouse network. *Electron. Notes Discrete Math.* **64**, 195–204 (2018)
2. Mourad, M., Paul, R., Karine, E., Samuel, V., Botta, G., Thibard, M.: Pooled warehouse management: an empirical study. *Comput. Ind. Eng.* **112**, 526–536 (2017)
3. Lu, C., Lu, Z.: The storage location assignment problem for outbound containers in a maritime terminal. *Int. J. Prod. Econ.* **135**(1), 73–80 (2012)
4. Li, J., Yang, G., Chen, F.: Research on location assignment of retail e-commerce storage center. *Ind. Eng. Manag.* **4**, 102–108 (2013)
5. Faraz, R., Jennifer, A.: Analytical models for an automated storage and retrieval system with multiple in-the-aisle pick positions. *IIE Trans.* **46**(9), 968–986 (2014)
6. SAP Homepage. <https://www.saponlinetutorials.com>. Accessed 21 Dec 2017



An Expert System for Diagnosis and Treatment of Hypertension Based on Ontology

Wang Jie¹(✉), Peng Yan¹, Ren Xiaoxiao², and Qiao Yixuan¹

¹ Capital Normal University, Beijing 100048, China
cnu_wangjie@126.com

² WuHan University, Wuhan 430072, China

Abstract. The ontology theory is widely used in various fields including medical research. In addition, the number of patients with hypertension in China has been increasing presently, and hypertension is the main risk factor for the occurrence and even death of cardiovascular and cerebrovascular diseases. This paper designs and implements an expert system for diagnosis and treatment of hypertension based on ontology theory. The system builds a diagnostic knowledge base referred to authoritative literature firstly, constructs hypertension ontology by Protégé, uses SWRL semantic web rule language to edit the inference rules, and then reasons out the patient's hypertension levels and drug risk hierarchy and the corresponding drug using strategy by using Jess reasoning machine. The hypertension diagnosis and treatment expert system can accumulate cases and store them in knowledge library to realize knowledge reuse and improve diagnostic efficiency.

Keywords: Ontology · Hypertension · Expert system · Knowledge base

1 Introduction

Hypertension is both the most common cardiovascular disease and the most important health problem. At present, the estimated number of hypertensive patients in China is up to 200 million and this poses a serious challenge to human public health problems. With the rise of “Internet +” and “Semantic Web” in recent years, the research of expert system has been greatly promoted, and expert system has great advantages in auxiliary diagnosis and disease diagnosis of doctors. Therefore, the research on expert system of hypertension diagnosis and treatment has very important significance.

In this paper, an ontology-based expert system for the diagnosis and treatment of hypertension is developed. Based on the stratification of hypertension, risk levels and medication recommendations in the *2010 Chinese guidelines for the management of hypertension* [1], and use ontology construction and semantic web rules and reasoning machine theory, the diagnosis knowledge base was established. This paper links hypertensive disease to ontology and tries to establish a knowledge base of hypertension diagnosis that can be reasoned automatically. We expect that the final expert system can accumulate cases and store them in the knowledge base to reuse the knowledge and improve the diagnostic efficiency.

2 Related Research

The medical treatment expert system is a computer system, which uses the expert system theory and knowledge to simulate the way in which human experts deduce the diagnosis and treatment of related diseases. It can help doctors analyze some complicated medical symptoms, and it plays a significant role in human medical treatment construction [2].

The first successful practice of computer-aided diagnosis and treatment was in the late 1950s. The United States combined the mathematical models based on Boolean algebra theory and Bayesian axiom with medical clinical experience to diagnose a lung cancer case accurately [3]. In 2011, the GDA-LS-SVM lung cancer diagnostic system was developed by Firat University [4]. Medical experts in the continuous development of the international system, gradually becoming more sophisticated and intelligent, greatly promoted the development of medical science.

In China, researchers in various fields have developed different kinds of disease diagnosis and treatment expert system in various medical fields [5]. In 2010, Liu had developed a prototype system and knowledge sharing management platform for departmental diagnosis and treatment services, case management of knowledge base, knowledge acquisition module and integrated diagnosis and treatment of knowledge and other subsystems [6]. In 2013, Cui applied ontology modeling method and ontology knowledge model that accord with open knowledge management to CDSS medical field [7]. In 2016, Chen and other scholars introduced two kinds of artificial intelligence technologies, RBR and CBR, into the design of diabetes diagnosis and treatment system [8].

3 Construction of Hypertension Diagnosis and Treatment Knowledge Base

Knowledge base includes ontology base and rule base. According to the definition in guideline [1] and the relationship between the constraints, hypertension ontology base complete the formal expression of knowledge in the field of hypertension in the form; and the rules of high blood pressure rules base construction is completed according to the guidelines. Finally, we use jess-reasoning machine to realize ontology reasoning.

3.1 The Principles of Constructing Ontology and the Process of Constructing Ontology

We use Protégé, a free visual ontology software developed by Stan Stanford University, to build the ontology. The OWL language is used to describe ontology. OWL describes knowledge by object-oriented model. The description of objects is achieved through classes and attributes.

The first task of constructing ontology base is to have a basic understanding of the content of the ontology. Firstly, we extract the various concepts in the whole field of hypertension, comprehensively list the important terms and key concepts. Then a high conceptual framework of blood pressure is established. According to the framework,

we fill it with ontology content. The establishment of the framework also needs to be constantly adjusted to select the optimal framework. Finally, according to the relationship between different concepts define the class and attributes which is the relationship between classes. The definition of the attributes needs to think ahead and sort out to reduce the difficulty of the process of reasoning. Finally, a patient instance is established. The building process follows five guidelines Gruber’s proposed for building ontologies [9] and is described in Fig. 1.

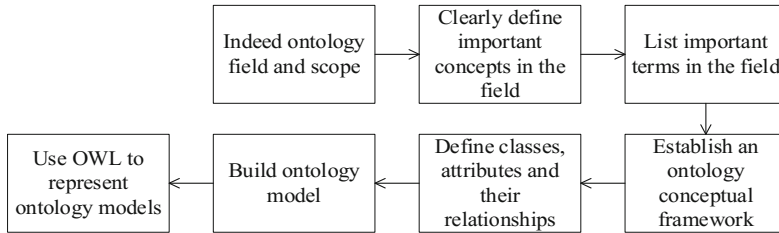


Fig. 1. Ontology of the building process

3.2 Method for Constructing Hypertension Diagnosis Ontology

By contrast, we choose the “skeleton method” to guide the process of building the ontology. Skeleton method is roughly divided into five steps, which is described in Fig. 2.

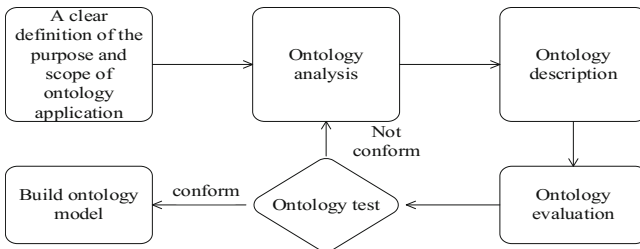


Fig. 2. Skeleton method to build ontology flow chart

Skeleton method to build ontology includes the following five steps. (1) Confirm the definition of the purpose and the scope of ontology application. (2) Ontology analysis. The ontology of the key terms and their relationship between the integration, a careful analysis of the relationship between them, detailing the content of hypertension diagnosis and treatment. (3) Ontology description. Use Protégé to build classes and their relationships and constraints between properties and classes. (4) Ontology evaluation. The definition cannot have other meaning, and should be complete, which can not only cover all the contents of hypertension diagnosis, but also have scalability. (5) Ontology establishing. Construct a perfect ontology through the above method.

3.3 Implementation of Hypertension Diagnosis Ontology

Confirm the main concepts in the diagnosis of hypertension, and define the ontology class, attributes, and examples. Class is an important concept and terms in the prevention and treatment of hypertension guidelines, which is the smallest organization of the ontology.

Thing is the parent class for all classes of OWL files. There are six subcategories of Thing including stratification of hypertension levels, examination, diagnosis, cardiovascular risk stratification, and patient. Figure 3 shows a hierarchy of hypertension diagnoses.

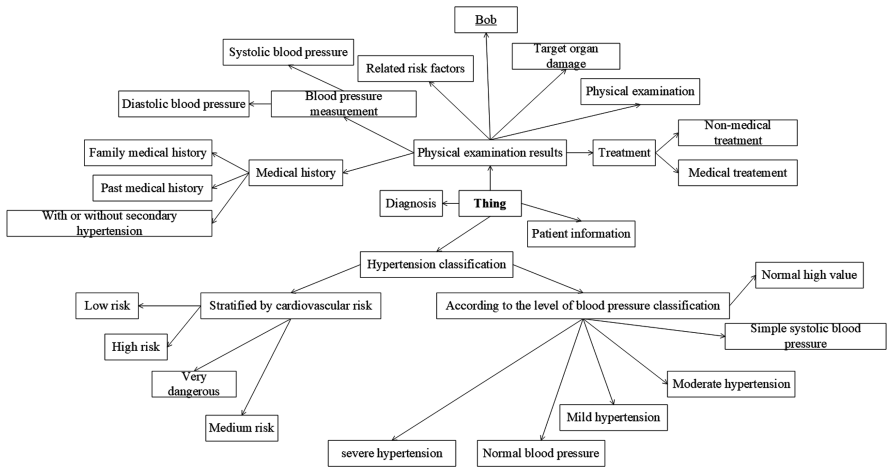


Fig. 3. The Fig. of hypertension diagnosis class hierarchy

The attributes of the diagnosis ontology of hypertension include the history, the target organ damage, the value of systolic blood pressure, the value of diastolic blood pressure, the age, gender, the number of risk factors, etc. These attributes belong to the data type and the data type can define the value of the class and its type. As well, risk stratification and level belonging to the object type attribute, and the object type attribute can define the relationship between classes and classes, including inheritance. Figure 4 shows the definition of object properties and data properties and their constraints settings during ontology building.

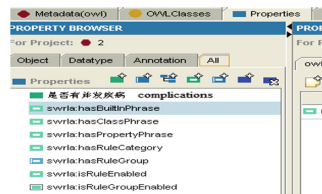


Fig. 4. Definition of object and data attributes and their constraints

Hypertension prevention and treatment guidelines include a large number of examples; we need to instantiate the ontology class instance combined with content of the guidelines. Instances are the most basic concepts that exist dynamically in the ontology.

Examples include age, sex, systolic blood pressure, diastolic blood pressure, history of hypertension, number of risk factors, and target organ damage. At this point, the object attribute “Suffering” which is the content of the disease situation is not defined. After the implementation of jess reasoning machine, the object attribute “suffering” content will be displayed. Diagnostic results are intended to show blood pressure levels and risk levels as well as medication recommendations and life advice. Figure 5 shows an example of a class of hypertension diagnosis.

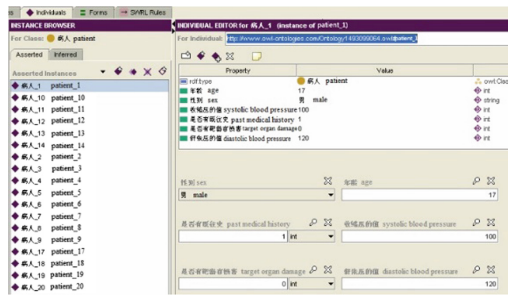


Fig. 5. The creation of a patient instance

3.4 Description of Ontology of Hypertension Diagnosis

The description of the ontology utilize OWL language, and OWL document can quote other OWL documents or being quoted by other OWL documents in order to achieve ontology reusability. OWL uses object-oriented way to describe knowledge. Object description is realized through classes and attributes. Then we use axioms to describe the characteristics and relationships that classes and attributes have. OWL document mainly consists of four parts [10]:

- (1) The definition of the ontology header, especially the document metadata [11], generally contains version information and some compatible information.

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:xsp="http://www.owl-ontologies.com/2005/08/07/xsp.owl#"
  .....
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:sqwrl="http://sqwrl.stanford.edu/ontologies/built-ins/3.4/sqwrl.owl#"
  xml:base="http://www.owl-ontologies.com/2.owl">
```

- (2) The definition of a class, including the description of the relationship between the class and its subclasses.
- (3) Property definition. There are two types attributes of OWL, called Object-Property and DatatypeProperty. Object attribute describes the relationship between two classes, and the data type attribute describes the value of the class and the relationship between other data types.
For example, “patient” class and its object type attribute “name” and the data type attribute “systolic blood pressure” is defined in owl as follows:

```
<owl:Class rdf:ID="patient"/>
  <owl:Object Property rdf:ID="name">
    <rdfs:domain rdf:resource="#patient"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:Object Property>
  <owl:Datatype Property rdf:ID=" systolic blood pressure ">
    <rdfs:domain rdf:resource="#patient"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:Datatype Property>
```

- (4) Instance definition. Instances are individuals of a particular class, associated with the properties of the class. Usually instances are regarded as test sets to verify the accuracy of ontology inference systems.
There is an instance of a Patient_1 in “patient” class in Hypertension Diagnostic Ontology, which includes all the attributes in the “patient” class definition, especially the patient’s examination and identity information. After completing the construction of the ontology knowledge base, the patient’s condition can also be diagnosed, and the diagnosis result will also be written into the owl document.
The specific OWL language is as follows:

```
<patient rdf:about="http://www.owl-ontologies.com/Ontology1493099064.owl# patient_1">
  < risk factors rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>1</ risk factors >
  < diastolic blood pressure rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>95</ diastolic blood pressure >
  < systolic blood pressure rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>170</ systolic blood pressure >
  < target organ damage rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>0</ target organ damage >
  < past medical history rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>0</ past medical history >
  < hypertension classification
rdf:resource="http://www.owl-ontologies.com/Ontology1493099064.owl# hyperten-
sion classification "/>
  < risk stratification
rdf:resource="http://www.owl-ontologies.com/Ontology1493099064.owl# moderate
risk "/>
</patient>
```

4 Construction of Hypertension Diagnosis and Treatment Knowledge Base

4.1 The Construction of Hypertension Rule Base

The construction of the rules requires the installation of the SWRLTab plug-in in Protégé. The plug-in opens in the Reasoning → Open SWRL Tab and then enters the SWRL Rules. The rules before the preparation of the rules must list Atom first, understand the meaning of the expression for the follow-up rules to provide the basis for the preparation.

Establish SWRL rules in Imp, and do condition judgment in Atom. There are part of the rules and their corresponding explanation in Atom: the patient (?p) that P is a patient; is there any target organ damage (?p, boolean) indicates that the patient p has target organ damage; SWRLb: greater Than Or Equal y, num) indicates that Y is greater than or equal to num [12] (Table 1).

Table 1. Atom list in SWRL rules for hypertension

Atom	Explanation	Atom	Explanation
patient (?p)	P is a patient	SWRLb:greater Than Or Equal (?y, num)	Y greater than or equal to num
diastolic blood pressure (?p, ?dia)	The value of the patient P's diastolic blood pressure	SWRLb:less Than Or Equal (?y, num)	Y less than or equal to num
systolic blood pressure (?p, ?sys)	The value of the patient P's systolic blood pressure	risk stratification (?p, low risk)	The patient P's cardiovascular risk is low risk
risk factors (?p, num)	The number of patient P's risk factors	risk stratification (?p, moderate risk)	The patient P's cardiovascular risk is moderate risk
target organ damage (?p, boolean)	Whether there is a target organ damage with patient P	risk stratification (?p, high risk)	The patient P's cardiovascular risk is high risk
past medical history (?p, boolean)	Whether there is a past medical history of hypertension with patient P	hypertension classification (?p, mild hypertension)	Patient p has a mild hypertension
complications (?p, boolean)	Whether there are complications with patient P	hypertension classification (?p, moderate hypertension)	Patient p has a moderate hypertension
suggestions (?con)	Suggested content	hypertension classification (?p, severe hypertension)	Patient p has a severe hypertension

Hypertension diagnostic rules are based on the guidelines for prevention and treatment of hypertension. According to the guidelines we summarize the diagnosis of hypertension rules and build Imp on the basis of these restrictions. Table 2 list the specific rules based on the guidelines for prevention and treatment of hypertension sort by the level of blood pressure.

Table 2. Blood pressure level classification rules

Blood pressure level	Knowledge representation	Knowledge explanation
Have hypertension	patient (?p) \wedge past medical history (?x, true) \rightarrow have (?p, ?x)	Patient P is diagnosed with hypertension if patient P has a history of hypertension
Mild hypertension	patient (?p) \wedge diastolic blood pressure (?p, ?dia) \wedge SWRLb:greater Than Or Equal (?dia, 90) \wedge SWRLb:less Than Or Equal (?dia, 104) \rightarrow have (?p, mild hypertension)	Patient P is diagnosed as having mild hypertension if patient P has a diastolic blood pressure (dia)between 90 and 104
Moderate hypertension	patient (?p) \wedge diastolic blood pressure (?p, ?dia) \wedge SWRLb:greater Than Or Equal (?dia, 105) \wedge SWRLb:less Than Or Equal (?dia, 114) \rightarrow have (?p, moderate hypertension)	Patient P is diagnosed as having moderate hypertension if patient P has a diastolic blood pressure (dia)between 105 and 114
Severe hypertension	patient (?p) \wedge diastolic blood pressure (?p, ?dia) \wedge SWRLb:greater Than Or Equal (?dia, 114) \rightarrow have (?p, severe hypertension)	Patient P is diagnosed with severe hypertension if the diastolic blood pressure of patient P is greater than 114

4.2 Inference Engine for Hypertension Diagnosis and Treatment

Hypertension diagnosis and treatment knowledge base use guideline [1] as the basis, through the construction of ontology and the preparation of SWRL rules to build a knowledge base, and the rules engine Jess implement the ontology reasoning.

The diagnosis and treatment of hypertension ontology is the most basic content in the reasoning proceed which is necessary to proceed with reasoning. The content of the ontology was completely extracted according to the authoritative guide for Chinese prevention and treatment, which ensured the ontology’s integrity and laid a solid foundation for the smooth progress of the later inference. Using the Semantic Web Rule Language SWRL is based on the setting rules of hypertensive medical ontology, the rules are an important part of Jess’s reasoning. Facing such a complex disease like hypertension, it is necessary to continuously increase the clinical experience of the corresponding experts. Moreover, ontology’s scalability ensure the addition and modification of the late rules. OWL files can transform ontologies and rules into jess-aware

repositories and rule bases that cannot be deduced without mapping from the owl file to the jass system. Proteus jess plugin is a good implementation of this point, as long as the appropriate plug-in installed and started, there will be a button that transfer owl file to jess conversion which is easy to achieve the conversion function, in order to achieve reasoning (Fig. 6).

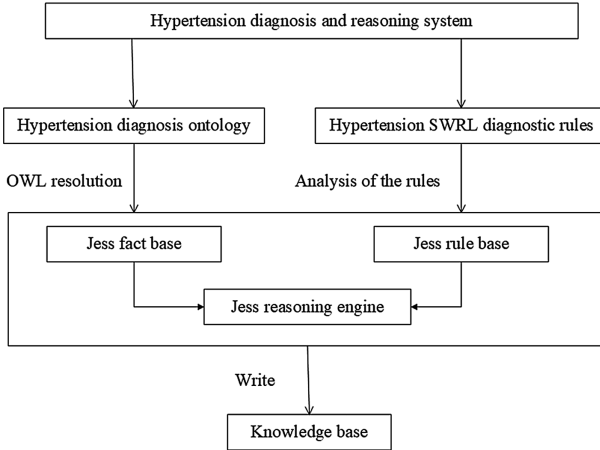


Fig. 6. The framework of ontology reasoning system

Throughout the reasoning system, the ontology of hypertension diagnosis and treatment is the foundation, which is the basis of the reasoning system. Using the Semantic Web Rule Language SWRL is based on the setting rules of hypertensive medical ontology which are an important part of Jess’s reasoning. The OWL parser transforms the information contained in the ontology’s classes, properties, and instances into facts that the Jess inference engine can recognize. The role of the SWRL parser is to implement mapping transformations from the SWRL rules to Jess’s recognizable rule format, which implements reasoning functions based on facts and rules that they can recognize. Through the Jess system, OWL files generated by reasoning are written into the knowledge base, which contains both the ontology and the rule of hypertension diagnosis, and the knowledge base is built with diagnostic inference function.

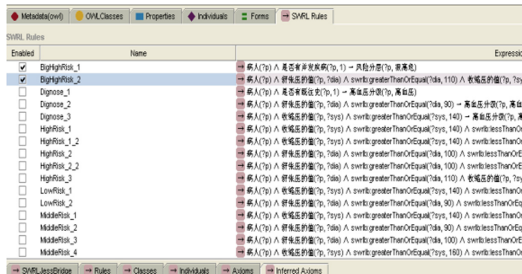


Fig. 7. The framework of ontology reasoning system

An example of diagnosis and treatment for hypertensive patients is shown in Fig. 7, in the patient's newly entered attribute, the "suffering" column is empty. When the reasoning engine finishes SWRL rule reasoning, the patient's corresponding level of hypertension is displayed in the margin. In the meantime, Rules of Reasoning page shows rules for running, Axioms shows the axioms numbers and the results of the run.

5 Conclusion

In this paper, based on the research and analysis of existing expert system, combined with semantic web and ontology and other modern technologies, the main function and implementation of hypertension diagnosis and treatment system are considered and conceived. Based on the guideline [1], the knowledge of hypertension is extracted, which includes the classification of hypertension, the classification of cardiovascular and cerebrovascular risk in patients with hypertension, and the medication recommendation and life advice of patients with hypertension. We define the class and the ontology knowledge domain; defines the interrelationship of core concepts and concepts with object and data attributes; and the OWL parser transforms the information contained in ontology classes, properties, and instances into facts recognizable by the Jess inference engine. The SWRL parser implements mapping transformations from the SWRL rules to Jess's recognizable rule format, which is implemented by the Jess inference system using the facts and rules identified above. Through the Jess system, OWL files generated by reasoning are written into the knowledge base, which contains both the ontology and the rule of hypertension diagnosis, and the knowledge base is built with diagnostic inference function.

Finally, in our current work, the system's knowledge base only draws on the diagnosis of guidelines [1]. While hypertension itself is a very complex condition, including the patient's daily diet, life-style, whether or not suffering from other diseases and many other factors affect the diagnosis. Therefore, experts in the Guide to Prevention and Treatment of Hypertension in China is the only part of the experience, and it still needs to constantly update the knowledge base to increase the diagnostic accuracy.

Acknowledgment. This work is supported by the Key Laboratory of machine intelligence and advanced computing (No. MSC-201707A), Project of Science Innovation Platform of Beijing Education Commission (No. 025185305000/035) and Project of interdisciplinary research project of Beijing Education Commission (No. 112175311500).

References

1. Liu, L.: Guidelines for the prevention and treatment of hypertension in China 2010. *Chin. J. Med. Front. (Electron. Ed.)* **3**(5), 42–93 (2011)
2. Li, F., Zhuang, J., Liu, K., et al.: The present situation and the trend of medical expert system. *Med. Inf.* **20**(4), 527–529 (2007)
3. Gallant, S.I.: Connectionist expert systems. *Commun. ACM* **31**(2), 152–169 (1988)

4. Avci, E.: A new expert system for diagnosis of lung cancer: GDA-LS_SVM. *J. Med. Syst.* **36**(3), 2005–2009 (2012)
5. Zhao, K., Ling, J.: An approach to building expert system based on neural network. *Pattern Recognit. Artif. Intell.* **8**(04), 320–325 (1995)
6. Liu, C.: Research on theory and technology of diagnosis and curing knowledge service based on knowledge flow. Shanghai Jiao Tong University (2010)
7. Cui, C.: Research and implementation of opening modeling tools for clinical knowledge based on ontology. Xi'an University of Electronic Science and Technology (2013)
8. Chen, G., Wang, J., Yang, Z.: Research on expert system of diabetes diagnosis and treatment based on combination of rule based reasoning and case based reasoning of ontology. *J. Chang. Univ.* **26**(6), 19–25 (2006)
9. Wu, H., Xie, H.: Research on hypertension diagnosis and treatment system on ontology and CBR. *Comput. Appl. Softw.* **30**(12), 155–159 (2013)
10. Guo, W.: SWRL based semantic relevant discovery and its application on ontology mapping and integration. Zhejiang University (2006)
11. Han, L.: Study on the construction of ontology and reasoning mechanism for the knowledge base system of information security management. Shandong University of Technology (2008)
12. Wang, J.: Research on hypertension diagnosis and treatment system based on association rules and ontology. Taiyuan University of Technology (2011)



A Three Input Look-Up-Table Design Based on Memristor-CMOS

Junwei Sun^{1,2}, Xingtong Zhao^{1,2}, and Yanfeng Wang^{1,2}(✉)

¹ Henan Key Lab of Information-Based Electrical Appliances,
Zhengzhou University of Light Industry, Zhengzhou 450002, China
yanfengwang@yeah.net

² School of Electrical and Information Engineering,
Zhengzhou University of Light Industry, Zhengzhou 450002, China

Abstract. The logic block of field programmable gate array (FPGA) basic unit is mainly composed of look-up-table (LUT). The conventional LUT using the static random access memory (SRAM), which leads to FPGA almost reach the limitation in term of the density, speed, and configuration overhead. In this paper, a new scheme of improved memristor-based look-up-table (MLUT) is proposed. The MLUT circuit, which is compatible with the mainstream circuit in FPGA. The MLUT effectively solving the limitations of field FPGA and MLUT is more efficient in data transmission than traditional LUT. In addition, the proposed circuit can achieve any combination logic function in MLUT by specific configuration. As a case study, a three-input LUT circuit based on memristor is designed and the correctness of the results is simulated in PSPICE software. The MLUT can replace traditional SRAM-based LUTs and further improve FPGA performance.

Keywords: Memristor · Look-Up-Table (LUT)
Field programmable gate array (FPGA)

1 Introduction

As a missing circuit component, the memristor was proposed by the scientist Chua in 1971 [1]. HP laboratory researchers in the United States successfully developed the first working memristor in 2008 [2], which verified the existence of the memristor. Memristor arouses extensive interest of industry and academia owing the advantages of nano-scale dimension, nonvolatility, fast access, and high density in comparison to the CMOS technology. The resistance of the memristor depends on the history of the applied electronic signal (voltage or current), which can be tuned to any arbitrary resistance state within the allowable range by appropriate applied voltage and current. The nonvolatile memristor makes it a competitive substitute for storage elements. The latent path current in the pure memristor crossover array is inevitable [3]. The memristor-CMOS hybrid architecture is an effective way to overcome the latent path defects [4], which shows the compatibility with the current mainstream CMOS and can easily access [5, 6]. On the other hand, memristors show good performance in logic computation [7–11]. Field programmable gate array (FPGA) is a powerful tool for

complex computing and high-speed digital signal processing, and has been widely used in information technology.

Field programmable gate array (FPGA) is a powerful tool for complex computing and high-speed digital signal processing. The FPGA has been widely used in information technology. The internal includes three parts: the configurable logic module CLB, the input and output module IOB, and the internal connection (Interconnect). As we all know, the key elements of configurable logic block (CLB) in FPGA are LUT, D flip flops and carry control logic [12]. Previous research on LUT has focused on how the size of the lookup table (LUT) affects the performance of FPGA [13, 14]. In this paper, FPGA and memristor technology are combined to design an improved memristor based LUT. Since the memristor has the ability to store information, the improved LUT does not need to download the configuration information from the external memory (the configuration information can be stored in the memristor). That is to say, traditional ROM and LUT have been replaced by MLUT. The improved LUT takes up smaller area overhead and shows more effective data transmission. The proposed MLUT pays more attention to detailed circuit implementation and performance analysis, and the circuit structure of a more compact decoder is simpler than that of [15]. In [16–18], the improvement of the whole FPGA architecture by memristor is emphasized. However, there is not much research on the design and implementation of MLUT circuits. On the other hand, compared with [15, 19, 20] (3T1 M), we simplify the required basic cell structure (2T1 M).

In this paper, a three-input MLUT is designed. Through the combination of a memristor and CMOS, the CLB architecture can be made more dense and more efficient. The rest of this paper is organized as follows: Sect. 2 briefly describes the memristor and its integration with CMOS technology, and introduces traditional configurable logic blocks. A memristor-based look-up table circuit is designed in Sect. 3. The Sect. 4 discusses the results and analysis of MLUT circuit simulation. Finally, this article is concluded in Sect. 5.

2 Background and Related Work

2.1 Memristor Character

The TiO₂-based thin film memristor fabricated successfully by HP laboratory [2] is one of the most popular representatives in many types of memristor. The physical model and the device symbol used in this paper are shown in Fig. 1. The memristor has a doped region with TiO_{2-x} and an undoped region with TiO₂ that are isolated by a thin film. The doped region has high conductivity, and the undoped region has low conductivity. When current forward flows past the memristor, the doped region extends and the undoped region shrinks, and as a result, the memristance will decrease. Otherwise, the memristance will increase.

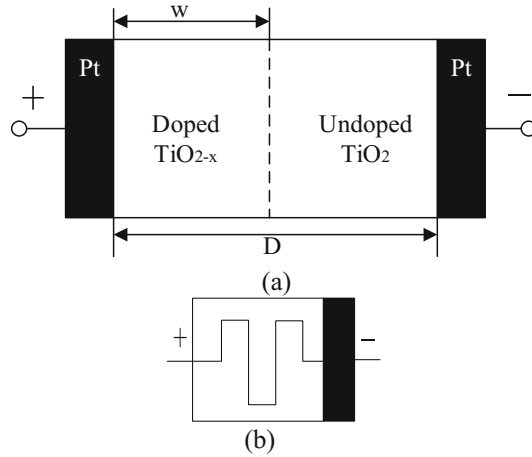


Fig. 1. Memristor diagrams. The end with “+” is the positive end and that with “-” is the negative end. (a) Construction of the HP memristor model and (b) device symbol of memristor used in this paper.

The memristance is defined as

$$M(t) = R_{on}x(t) + R_{off}(1 - x(t)) \tag{1}$$

$$\frac{dx(t)}{dt} = kf(x)i(t) \tag{2}$$

In Eqs. (1) and (2)

$$x(t) = \frac{\omega(t)}{D} \tag{3}$$

$$k = \frac{\mu_v R_{on}}{D^2} \tag{4}$$

Where $\omega(t)$ is the width of the doped region, D is the width of the total region, μ_v is the average ion mobility. According to [6], ionic mobility μ_v is set to 10^{-7} , control the memristor programming time scale. Obviously, $0 \leq (\omega(t)/D) \leq 1$, so that, if $\omega(t) = 0$, $M(t) = R_{off}$ and $\omega(t) = 1$, $M(t) = R_{on}$.

The threshold character of memristor is considered [3], when the absolute value of voltage is less than the threshold, the memristance will not change in the computing circuit. Supposing the absolute value of positive and negative threshold voltages both are 1 V, we utilize the specific small voltage under threshold to work as computing mode. When the voltage is larger than the threshold, the memristance will change depending on the applied voltage [21–24].

2.2 Configurable Logic Block

The programmable logic module CLB stands for configurable logic block. For detailed information and various functions of CLB, referred to the Spartan-6 FPGA CLB User Manual [12]. The configurable logic blocks are made up of 4 interconnected slice and additional logic. And the CLB can implement the corresponding logic, timing functions and have certain computing power. Due to the large number of CLBs in the FPGA, most of the functions of the FPGA can be constructed by using cascades. Its topology is shown in Fig. 2(a). Each pair of slices is distributed in the same column and has a separate carry chain.

Each CLB contains a pair of slices and a slice contains four logic function generators (LUTs) and eight storage elements. The four storage elements in the slice can be configured as edge-triggered D-type flip-flops or level-sensitive latches for implementing sequential logic. In addition, each function generator can implement a 64-bit ROM. The function generator in the Spartan-6 FPGA is the 6-input look up table (LUT). There are four for each slice. The LUT has six independent inputs (A_1 - A_6) and two independent outputs (O_5 - O_6), which is shown in Fig. 2(b). The function generator

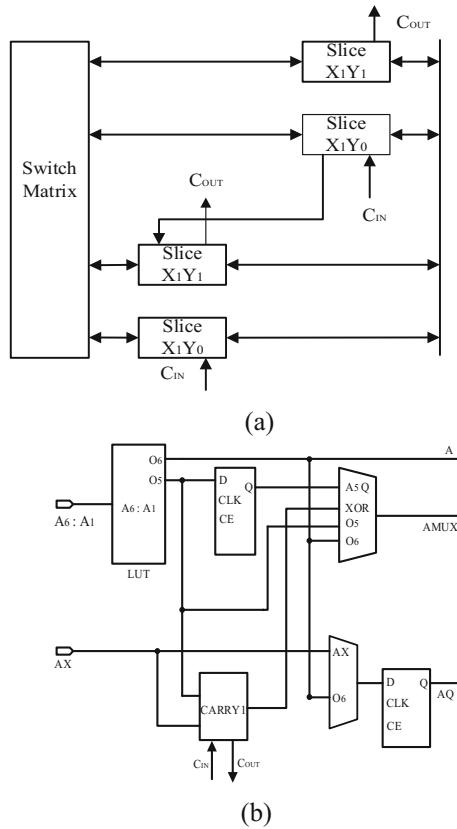


Fig. 2. (a) Schematic diagram of CLB. (b) Quarter of a slice.

can implement an arbitrary six-input Boolean function and two arbitrary 5-input Boolean functions. In addition to the basic LUT, the slice contains multiple multiplexers. These multiplexers are used to combine up to four function generators to provide any function of the eight inputs in the slice.

3 Memristor-Based Look-up-Table Design

In this section, the basic reconfiguration logic design is given in [15]. Memristor-CMOS hybrid structure can provide a complete combination logic function. A digital binary decoder must be used to select the correct single output from the specific input signal combination in the structure. Correspondingly, the digital $n-2n$ decoder is a necessity in the n -input LUT, which selects a single output according to the special logic function. The detailed schematic circuit has two modes of operation: (1) computing mode and (2) configuration mode.

3.1 Computing Mode and Configuring Mode

The main part of the calculation module circuit: nMOS and pMOS field effect transistors, resistors, a decoder, memristors, and a comparator. The computing mode circuit working process is shown in Fig. 3. pMOS transistor S_1 is connected to the voltage V_D through the resistor R_1 . The voltage V_D , which is less than the threshold voltage, supplies a power for the circuit. The voltage V_P is connected to the nMOS transistor, and can be used to apply the above-threshold voltage. Set the memristor device to any continuous analog state with proper continuous time. And the voltage V_P , which is larger than the threshold the voltage, delivers required to switch memristors M_i between R_{on} and R_{off} . In addition, when node H selects logic 0, the transistor switch is turned on, then computing the reconfigurable logic state. When node H selects logic 1, the transistor switch is turned off, then to configure the reconfigurable logic state.

When the node H is connected to logic 0, the gate of the pMOS transistor S_1 and S_3 will be in the ON state. The gate electrodes of nMOS transistors S_2 and S_4 will be in the OFF state. Therefore, the V_D will be connected in series with the drain terminals of the resistor R_1 and nMOS transistor S_{19} . In addition, the memristor will be grounded through the pMOS transistor S_3 . The circuit current flow of this process is shown in Fig. 3(a), indicated by the dotted line.

The configuring mode circuit working process is shown in Fig. 3(b). The node H is set to logic one. Thus, the gate electrodes of nMOS transistors S_2 and S_4 be in the ON state, while the gate electrodes of pMOS transistors S_1 and S_3 will be in the OFF state. Therefore a direct path from node V_P to ground is now established and node V_D is isolated. Only one memristor can be configured at a time and the memristor can be reconfigured corresponding to the output port selected by the decoder. Thus, any particular logic function can be implemented by configuring one memristor at a time through several steps. The step count is equal to the amount of the memristors used in the MLUT.

We assuming that the channel resistance of all transistors is negligible, the voltage on the selected memristor $M_i (i = 1, 2 \dots 7, 8)$ is given by the following:

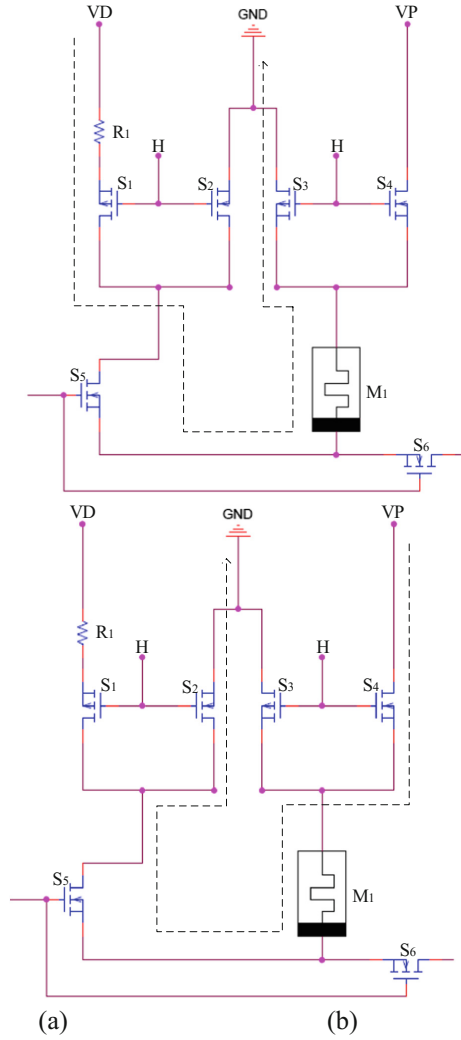


Fig. 3. (a) Schematic of the computing mode circuit current flow ($H = 0$). (b) Schematic of the configuring mode circuit current flow ($H = 1$).

$$V(M_i) = \begin{cases} VD \frac{M_i}{R_1 + M_i}, & Y_i = 1 \\ 0, & Y_i = 0 \end{cases} \quad (5)$$

$M_i (i = 1, 2 \dots 7, 8)$ and Y_i are the i th memristor and i th decoder output in Fig. 4, respectively. VD is the voltage source, whose value is set to 0.9 V. The parameters of memristor model mentioned in Sect. 2.1, Memristor internal parameters: $R_{on} = 1 \text{ k}\Omega$, $R_{off} = 100 \text{ k}\Omega$, $D = 10$, $\mu_v = 100 \text{ n}$, $p = 10$, $V_T = 1 \text{ V}$. To achieve the maximum read sensing margin, the resistance R_1 value is the geometric mean of R_{on} and R_{off} , i.e.,

$R_1 = 100\text{ K}\Omega$ [21]. Then by Eq. 1, the $V(M_i)$ can vary from $V(M_i = R_{on}) = 0.082\text{ V}$ up to $V(M_i = R_{off}) = 0.818\text{ V}$. When $Y_7 = 1$, the nMOS transistor S20 will be in the ON state. This will deliver the voltage potential $V(M_i)$ to the input of the voltage tuning block circuit (a comparator). The function of the voltage tuning block is to set the input voltage to the TTL voltage. That is to say, the output port will be 5 V if the input of the block $V(M_i)$ is larger than 0.5 V. While the output port will be 0 V if $V(M_i)$ is lesser than 0.5 V.

Note that there are one memristor and two transistors for each decoder output. The output of MLUT directly depends on the state of the memristor connected to it. Thus the logic of the overall output circuit only depends on the selected single memristor device decoder, while all other memristor states can be ignored. In addition, decoupling transistors can be omitted compared with [15], because the output voltage of each unselected decoder is 0 V, which does not affect the selected cell. And the exclusive OR (XOR) function is designed in the MLUT since the specific logic function will be applied in the following design in the CLB of FPGA. The truth table is also showed in Table 1. Here, all memristors are configured well beforehand using the configuring mode circuit.

3.2 MLUT Circuit

The MLUT circuit is designed in this paper, which is made up of: MOS field effect transistors, inverters, memristors, resistors, a decoder, and a comparator. T_1 - T_8 are high speed silicon gate CMOS devices 74HC04, the inverter is used for inverting logic operations, it is a six-inverter, that is, there are six inverters on one integrated block. The input of the inverter is high level, the output is low level; the input is low level, the output is high level. When the decoder inputs A_1 , A_2 and A_3 both are equal to logic one, the output port Y_7 of the decoder is logic 1, while other ports Y_0 , Y_1 , Y_2 , Y_3 , Y_4 , Y_5 and Y_6 are logic 0. Under this circumstance, the gate electrode of nMOS transistor S_{19} is in the ON state, the gate electrodes of nMOS transistors S_5 , S_7 , S_9 , S_{11} , S_{13} , S_{15} , and S_{17} remain in the OFF state.

Table 1. Truth table of three-input computing logic: XOR

Input value			Decoder out selection	Memristor state configuration	Output value
In 1	In 2	In 3			
0	0	0	Y_0	$M_1 = R_{on}$	0
0	0	1	Y_1	$M_2 = R_{off}$	1
0	1	0	Y_2	$M_3 = R_{off}$	1
0	1	1	Y_3	$M_4 = R_{on}$	0
1	0	0	Y_4	$M_5 = R_{off}$	1
1	0	1	Y_5	$M_6 = R_{on}$	0
1	1	0	Y_6	$M_7 = R_{on}$	0
1	1	1	Y_7	$M_8 = R_{off}$	1

When the cell is working on the computing mode, in other words, $H = -5\text{ V}$, $V_D = 0.9\text{ V}$, $Y_0 = 5\text{ V}$, then pMOS transistors S_1 , S_3 , and nMOS transistor S_5 are in the ON state. Since the boundary effect of memristor affects the performance of the circuit, in order to avoid the boundary effect of memristor, we assume that $R'_{off} = 90\text{ k}\Omega$. By the Eq. (3), $V(M_i)$ is calculated as $90/(10 + 90)*V_D = 0.81\text{ V}$. The simulation result is 801.8 mV . The difference between the calculation and the simulation result is just 1.01% , which is acceptable in digital logic. Similarly, we assume $R'_{on} = 2\text{ k}\Omega$. $V(M_i)$ is calculated as $2/(10 + 2)*V_D = 0.15\text{ V}$ according to Eq. (3). The simulation result is 148.2 mV . Thus, the difference is about 1.2% .

When the cell is working in the configuration mode, that is, $H = 5\text{ V}$, $V_P = 5\text{ V}$, the nMOS transistor S_2 , S_4 and S_5 are in ON state, S_2 and S_4 are working in the

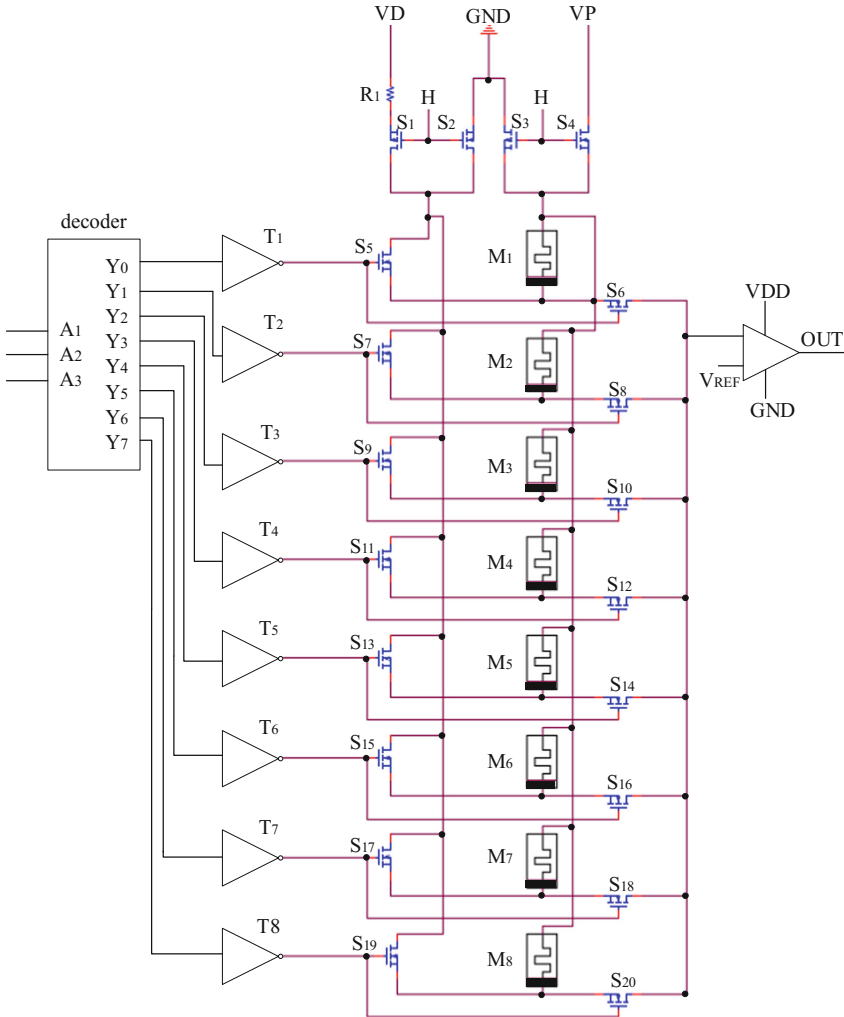


Fig. 4. Memristor-based three-input LUTS

saturated zone and running well. However, S5 is working on the nonlinear regions. So it doesn't work in a perfect environment, but the configuration is still right. when $VP = -5$ V, the memristor of the selected memristor can be adjusted from low resistance to high resistance. Similarly, memristor can be resisted from high resistance to low resistance when $VP = 5$ V. That is to say, the configuration mode circuit can work normally.

Compared with [15], the reconfigurable architecture is improved in this paper. The improved circuit can save one transistor per basic unit. Because there are millions of basic units in the FPGA chip, the use of more compact units can save a lot of total area. And compared with pure memristor crossover array, another advantage of this design is that there is almost no sneak path current [4]. The memory of the unselected memristor does not change very much, almost without verifying the potential path current in the structure.

In Fig. 4, a schematic diagram of the three-input MLUT is given. MLUT in [13] uses relatively complex and redundant decoders, resulting in greater regional overhead without considering cascading. And the conventional three-input LUT at least needs eight storage elements ($6S * 8$) and a decoder. In addition, the information stored in LUT needs extra steps to download from external ROM. However, the MLUT proposed in this paper needs only $8M + 20S$, the voltage tuning block and the decoder. In addition, the proposed MLUT saves at least $3 * 2^n$ transistors in n input LUT. Moreover, because of the nonvolatile memory resistance, the proposed design can keep the stored information in the memristor unit when the power is closed, which is not easy to implement in the traditional LUT design. Based on the above two advantages, it ensures that the design proposed in this paper will contribute to the design and development of FPGA in the future.

4 Simulation Results

The proposed MLUT circuit design combines the computing logic and configuring logic method. Because the logic of CLB includes a logic gate, a universal full adder can be designed and analyzed in detail by using the MLUT circuit. The simulation of the circuit using PSPICE software in CADENCE environment. We use the modified SPICE netlist in [21] for the memristor, and the type of CMOS chips 74HC138, IRF250, IRF9130 and OP-07. The traditional CLB uses SRAM based LUT, carry control logic and D trigger to realize the function of n -bit adder. The memristor-based three-input LUTs simulation is shown in Fig. 5.

In this simulation result, all horizontal coordinate units are time, and vertical coordinate units are volts. The A1, A2 and A3 are inputs, a-XOR is the analog output of MLUT with XOR logic function, and a-XOR is also the voltage before the comparator. The b-XOR is the digital output of the specific MLUT, also is the voltage after the comparator. S and C is the output. The MLUT used in Fig. 4 has been configured well to implement XOR function shown in Table 1.

Note: Due to the influence of the selected components and the performance of the circuit itself, the simulation will cause certain fluctuations and errors, but it will not affect the entire simulation results. The acceptable range of these fluctuations and errors is $\pm 2\%$.

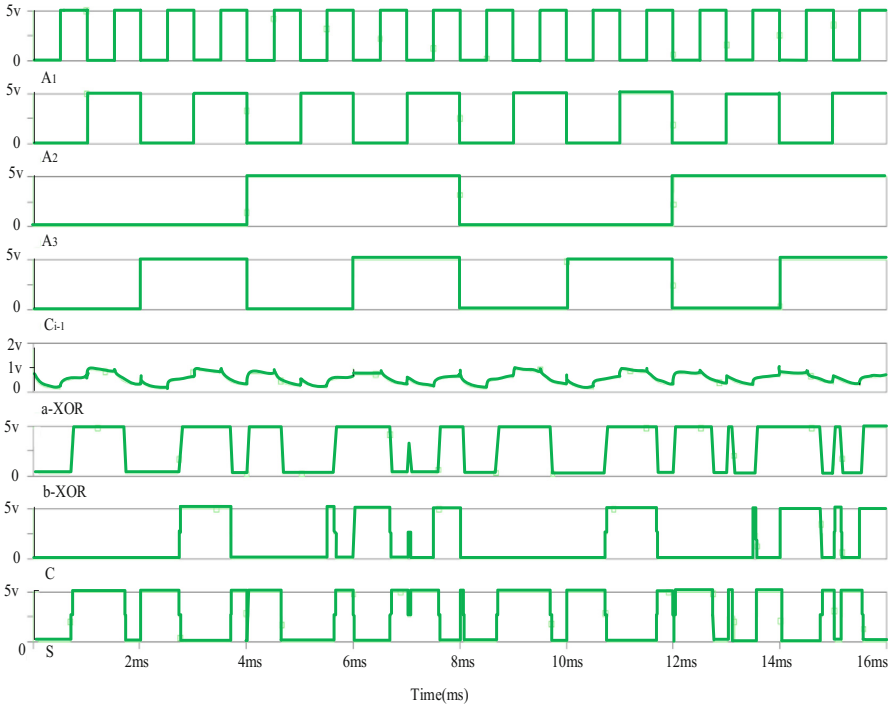


Fig. 5. Simulation results of three input MLUT

5 Conclusion

This paper presents a detailed design of MLUT. The performance of the circuit analyzed, and the correctness of the design verified by simulation software. Compared with previous literature, the design shows the advantages of taking up a small area of overhead and using the memory capability of the memristor to eliminate the downloading of configuration information from external memory. We believe that this research will help FPGA technology development. Future research will focus on memristor simulation, aiming at achieving greater improvement in FPGA design.

Acknowledgment. The work is supported by the State Key Program of National Natural Science of China (Grant No. 61632002), the National Key R&D Program of China for International S&T Cooperation Projects (No. 2017YFE0103900), the National Natural Science of China (Grant Nos. 61603348, 61775198, 61603347, 61572446, 61472372), Science and Technology Innovation Talents Henan Province (Grant No. 174200510012), Research Program of Henan Province (Grant Nos. 172102210066, 17A120005, 182102210160), Youth Talent Lifting Project of Henan Province and the Science Foundation of for Doctorate Research of Zhengzhou University of Light Industry (Grant No. 2014BSJJ044).

References

1. Chua, L.O.: Memristor—the missing circuit element. *IEEE Trans. Circuit Theory* **18**(5), 507–519 (1971)
2. Strukov, D.B., Snider, G.S., Stewart, D.R., Williams, R.S.: The missing memristor found. *Nature* **453**(7191), 80 (2008)
3. Tanachutiwat, S., Liu, M., Wang, W.: FPGA based on integration of CMOS and RRAM. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **19**(11), 2023–2032 (2011)
4. Zidan, M.A., Omran, H., Sultan, A., Fahmy, H.A., Salama, K.N.: Compensated readout for high-density MOS-gated memristor crossbar array. *IEEE Trans. Nanotechnol.* **14**(1), 3–6 (2015)
5. Mohammad, B., Homouz, D., Elgabra, H.: Robust hybrid memristor-CMOS memory: Modeling and design. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **21**(11), 2069–2079 (2013)
6. DeHon, A.: Reconfigurable architectures for general-purpose computing. Ph.D., dissertation, MIT, December 1996
7. Xu, C., Dong, X., Jouppi, N.P., Xie, Y.: Design implications of memristor-based RRAM cross-point structures. In: Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1–6. IEEE Press, New York (2011)
8. Borghetti, J., Snider, G.S., Kuekes, P.J., Yang, J.J., Stewart, D.R., Williams, R.S.: ‘Memristive’ switches enable ‘stateful’ logic operations via material implication. *Nature* **464**(7290), 873 (2010)
9. Monmasson, E., Cirstea, M.N.: FPGA design methodology for industrial control systems—a review. *IEEE Trans. Ind. Electron.* **54**(4), 1824–1842 (2007)
10. Yang, J.J., Pickett, M.D., Li, X., Ohlberg, D.A., Stewart, D.R., Williams, R.S.: Memristive switching mechanism for metal/oxide/metal nanodevices. *Nat. Nanotechnol.* **3**(7), 429 (2008)
11. Snider, G. S.: Architecture, methods for computing with reconfigurable resistor crossbar. U. S. Patent 7,203,789 (2007)
12. Bourdeauducq, S.: Time to digital converter core for spartan 6 FPGAs (2011)
13. Kumar, T.N., Almurib, H.A.F., Lombardi, F.: A novel design of a memristor-based look-up table (LUT) for FPGA. In: 2014 IEEE Asia Pacific Conference on Circuits and Systems, pp. 703–706. IEEE Press, New York (2014)
14. Cong, J., Xiao, B.: mrFPGA: a novel FPGA scheme with memristor-based reconfiguration. In: 2011 IEEE/ACM International Symposium on Nanoscale Architectures, pp. 1–8. IEEE Press, New York (2011)
15. Ahmed, E., Rose, J.: The effect of LUT and cluster size on deep-submicron FPGA performance and density. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **12**(3), 288–298 (2004)
16. Dong, C., Chen, D., Haruehanroengra, S., Wang, W.: 3-D nFPGA: a reconfigurable architecture for 3-D CMOS/nanomaterial hybrid digital circuits. *IEEE Trans. Circuits Syst. I Regul. Pap.* **54**(11), 2489–2501 (2007)
17. Kuon, I., Rose, J.: Area and delay trade-offs in the circuit and architecture design of FPGAs. In: Proceedings of International Symposium on Field Program Gate Arrays, Monterey, CA, USA (2008)
18. Bruchon, N., Torres, L., Sassatelli, G., Cambon, G.: Magnetic tunnelling junction based FPGA. In: Proceedings of the 2006 ACM/SIGDA 14th International Symposium on Field Programmable Gate Arrays, pp. 123–130. ACM, New York (2006)

19. Canis, A., Choi, J., Aldham, M., Zhang V.: LegUp: high-level synthesis for FPGA-based processor/accelerator systems. In: Proceedings of the 19th ACM/SIGDA International Symposium on Field Programmable Gate Arrays, pp. 33–36. ACM (2011)
20. Gokhale, M., Stone, J., Arnold, J., Kalinowski, M.: Stream-oriented FPGA computing in the streams-c high level language. In: 2000 IEEE Symposium on Field-Programmable Custom Computing Machines, pp. 49–56. IEEE Press, New York (2000)
21. Biolek, Z., Biolek, D., Biolkova, V.: SPICE model of memristor with nonlinear dopant drift. *Radioengineering* **18**(2), 210–214 (2009)
22. Qi, A.X., Zhang, C.L., Wang, G.Y.: Memristor oscillators and its FPGA implementation. *Adv. Mater. Res.* **383**, 6992–6997 (2012)
23. Wang, X., Chen, Y.: Spintronic memristor devices and application. In: Proceedings of the Conference on Design, Automation and Test in Europe, pp. 667–672. European Design and Automation Association (2010)
24. Homouz, D., Abid, Z., Mohammad, B.: Memristors for digital, memory and nermorphic circuits. In: 25th International Conference on Microelectronics (ICM), pp. 1–4. IEEE, Beirut (2014)



Complex Logic Circuit of Three-Input and Nine-Output by DNA Strand Displacement

Yanfeng Wang^{1,2}, Guodong Yuan^{1,2}, Chun Huang^{1,2},
and Junwei Sun^{1,2} (✉)

¹ Henan Key Lab of Information-Based Electrical Appliances,
Zhengzhou University of Light Industry, Zhengzhou 450002, China
junweisun@yeah.net

² School of Electrical and Information Engineering,
Zhengzhou University of Light Industry, Zhengzhou 450002, China

Abstract. In recent years, DNA strand displacement technology has developed rapidly with the advantages of high-performance parallel computing and large information capacity. DNA strand displacement technology is a dynamic DNA nanotechnology developed on the basis of DNA self-assembly technology. DNA strand displacement technology can realize dynamic connection between input signal and output signal, and it is a method for constructing logic gates and logic circuits. According to the basic principle of DNA strand displacement, this paper studies the specific function of the molecular logic circuit model, and constructs a logic circuit with three input and nine output. Then a biochemical logic circuit is built based on the dual-rail thought and the construction of seesaw gate. Finally, DNA strand displacement simulation platform Visual DSD is used to verify the rationality of the model. It lays the foundation for constructing complex logic circuits.

Keywords: DNA strand displacement · Logic circuit · Visual DSD

1 Introduction

DNA is a powerful data storage and information processing medium that is considered a promising molecular computational material due to its good structure as well as ideal folding pathways and conformational changes [1, 2]. DNA computing is a new field combining computer science with molecular biology [3, 4]. As a new computing tool, DNA solves many complex problems with its powerful parallel computing capabilities, such as Hamilton Path, which is the biggest team problem [5–7]. DNA computation combines many molecular manipulation techniques: self-assembly, fluorescent labeling, strand displacement, and nanochains [8]. DNA strand displacement is a technology in biocomputing in recent years, which is derived from the optimization of DNA self-assembly technology [9, 10]. DNA strand displacement technology is also a dynamic DNA nanotechnology with spontaneity, sensitivity and accuracy [11–13].

In recent years, DNA strand displacement technology has made great progress. Through the strand displacement cascade reaction [14], the dynamic connection of adjacent logic modules is realized, enabling researchers to build large-scale, complex

logic circuits. In addition, DNA strand displacement technology has the advantages of high-capacity information accumulation, high-performance parallel computing, programming and simulation [15, 16], and has been deeply studied in the fields of molecular computing, nanomachines, diagnosis and disease treatment [17, 18]. DNA computing is very proficient in solving some mathematical problems, managing nanomachines and discussing life processes. Biochemical logic is the basis of DNA computing. Therefore, mastering the design process of constructing biochemical logic circuits is an important research method based on DNA strand displacement [19].

The logic circuit with three inputs and nine outputs is constructed by studying the functions of the molecular logic circuit. The specific output results can be obtained by changing different input signals. Compared with traditional logic circuits, the logic circuit designed in this paper is more flexible, and it has great possibilities for the construction of large and complex molecular logic circuits.

This work is presented as follows: in Sect. 2, the basic principle of DNA strand displacement is introduced. Followed by, a complex logic circuit of three-input and nine-output is designed in Sect. 3. In Sect. 4, the reliability of complex circuits is simulated. Some conclusions are finally obtained in Sect. 5.

2 Principle of DNA Strand Displacement

DNA strand displacement technique is based on the Watson-Crick base pairing principle and is derived from the improvement of DNA self-assembly technology [20–23]. DNA strand displacement cascades enable dynamic linking of input and output signals and are new techniques for building logic circuits [24–28]. DNA strand displacement reaction is actually a process, in which an input single-stranded DNA molecule undergoes a base pairing reaction with a complementary partial double-stranded DNA structure, thereby finally producing a double-stranded new structure and simultaneously releasing DNA.

DNA strand displacement branch migration process is shown in Fig. 1. R-T and S-T1-R represent input and output signals, respectively, and R represents the recognition region. T denotes toehold area. T* is specifically complementary to T, and toehold is a shorter base sequence, usually containing a sequence of 4–6 bases. The initial phase of branch migration releases the output chain. The toehold T first binds to the exposed T* in the partially double-stranded complex and then replaces the bases in the same region on the double-stranded complex until all substitutions are completed. Finally, only the original binding DNA strand hanging to the right of the partially double-stranded complex with toehold will gradually fall off and become an output signal. When the input chain and the output chain have the toehold, the output of the previous logic gate can be used as the input of the next logic gate, which provides favorable conditions for constructing the cascade reaction circuit.

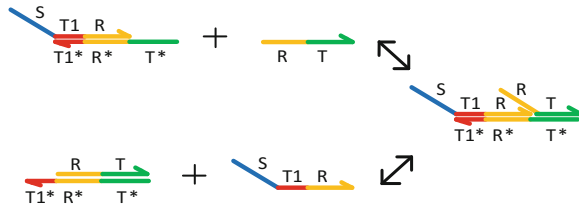


Fig. 1. DNA strand replacement branch migration process

Table 1. Truth table for complex logic circuits

Input			Output								
A	B	C	Y ₀	Y ₁	Y ₂	Y ₃	Y ₄	Y ₅	Y ₆	Y ₇	Y ₈
0	0	0	1	1	1	0	0	0	0	0	0
0	0	1	0	0	0	1	1	1	0	0	0
0	1	0	0	0	0	0	0	0	1	1	1
0	1	1	1	0	0	1	0	0	1	0	0
1	0	0	0	1	0	0	1	0	0	1	0
1	0	1	0	0	1	0	0	1	0	0	1
1	1	0	1	0	0	0	1	0	0	0	1
1	1	1	0	0	1	0	1	0	1	0	0

3 Design of Complex Logic Circuit

3.1 Digital Logic Circuit and Truth Table

Table 1 is a truth table for the three-input and nine-output logic circuits. The design of complex logic circuits in this paper is based on the relationship between input and output in the table. Eight kinds of output combinations are given in this paper. Each output combination consists of three independent outputs. For example, when the input signal is $ABC = 000$, the output signal is $Y_0Y_1Y_2 = 111$, and when the input signal is $ABC = 001$, the output signal is $Y_3Y_4Y_5 = 111$. As can be seen from the table, Y_4 is output signal when the input ABC is $001, 100, 110$ and 111 respectively. So the logical expression form $Y_4 = \bar{A}BC + A\bar{B}\bar{C} + AB\bar{C} + ABC$. The logic circuit diagram is then constructed according to the logical expression of Y_4 . The remaining eight output Y_0 – Y_8 logic diagrams are also constructed in the same way.

Logical operations have two states, “0” or “1”. The value is “1” when the input signal is received, and the value is “0” when the corresponding signal is not received. Usually in general logic circuits, the logic algorithm includes three modes of operation, namely logical AND, logical OR and logical NOT. Figure 2 is the logic diagram of the design, where A, B and C are binary inputs, and Y_0 – Y_8 are circuit outputs. The middle gate circuit and the connection of the line are constructed according to the relationship between the input and output of the two ends, and then the entire logic circuit is drawn by using the visio software.

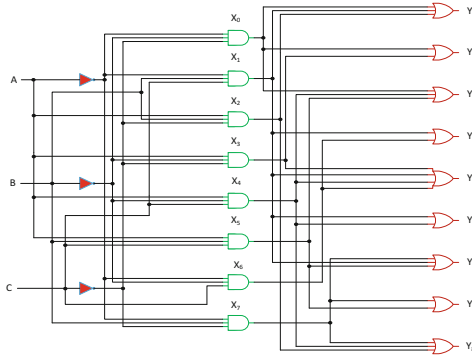


Fig. 2. Three-input and nine-output logic circuits

3.2 Dual-Rail Logic Circuit

When the input signal is logic “0” in the logic circuit, it means no signal input. However, in this case, the output signal is not an absolute logic “0” or a logic “1”, which sometimes results in an erroneous output signal. To avoid this, the method of using a dual-rail logical expression, that is, the input signal A is represented as “true” and “false”, and is recorded as A^0 and A^1 . When signal A^0 does not participate in the reaction, A^0 represents the logic “OFF” in the dual-rail logic, then the corresponding A^1 represents the logic “OFF”. In a dual-rail circuit, “and”, “or”, and “not” logic are produced by a combination of a pair of “AND” and “OR” logic gates. Dual-rail logic has been widely used in the construction of seesaw circuits. Referring to Fig. 3(a)–(d), a schematic diagram of the corresponding dual-rail logic conversion of the AND gate, OR gate, NAND gate and NOR gate is obtained. According to the basic dual-rail logic conversion rule described above, a dual-rail circuit diagram of the entire complex logic circuit can be constructed. Figure 3 is dual-rail circuit diagram of the output circuit Y_4 .

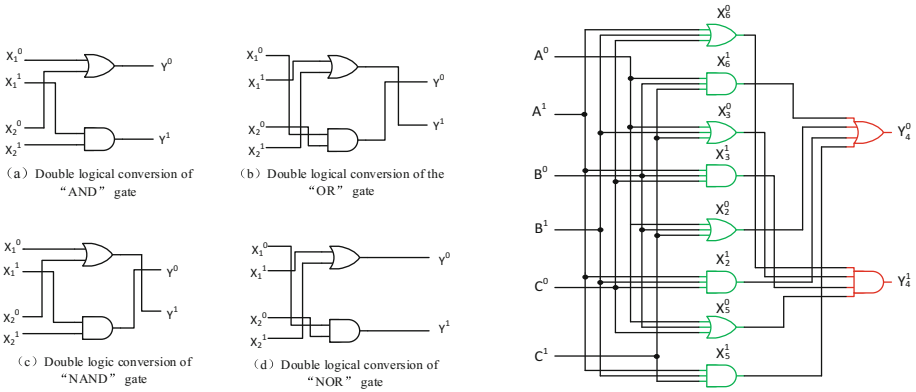


Fig. 3. Dual logic conversion of basic logic gates and dual-rail logic diagram of output signal “ Y_4 ”

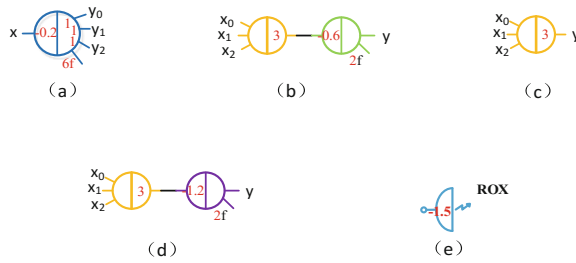


Fig. 4. Seesaw graphics of the basic logic gate

3.3 Design of the Seesaw Cascade Circuit

Some digital logic gates are often converted into biological gates in biometric research. The logic gates applied in the seesaw biochemical reaction mainly include amplification gates, integrated gates, threshold gates, and report expression gates, as shown in Fig. 4 (a)–(e). The amplification gates contain the threshold and fuel (the DNA strand added during the reaction to allow the reaction to continue), and the output signal is generated if and only if the total concentration of the input signal is greater than the initial concentration of the threshold. Otherwise, the output concentration logic value is 0.

The function of the amplification gate is used to obtain multiple output signals. These output signals have the following characteristics: When toehold is used as a demarcation point, the output signal contains the same left side recognition area and a different right side identification area. The reason is that the output signal is generated by the threshold selection of the same gate complex and needs to be applied to different low-level gate complexes. To facilitate adequate release of the output signal, the initial concentration of fuel is typically set to twice the total concentration of a given output signal.

The function of the integrated gates is opposite to that of the magnifying gates. It can receive multiple input signals and integrate them into one output signal after the reaction. These input signals have the following characteristics: When toehold is used as a demarcation point, the input signal contains the same right side recognition area and a different left side identification area. The reason is that the input signal is generated by the threshold of different gate complexes and needs to act on the same lower level gate complex. The function of the threshold gates is used to filter the input signal by concentration. If the total concentration is greater than the threshold concentration, an output signal is produced. Otherwise, there is no output.

The main function of the report expression gates is to facilitate observation of the experimental results. When an output signal is generated, the expression gates will generate a fluorescent signal for observation and analysis. In the design of the biochemical circuit, the threshold of the OR gate is taken as 0.6 based on the empirical value, and the threshold of the AND gate is taken as 1.2. The basic seesaw logic gate in the molecular circuit is shown in Fig. 5.

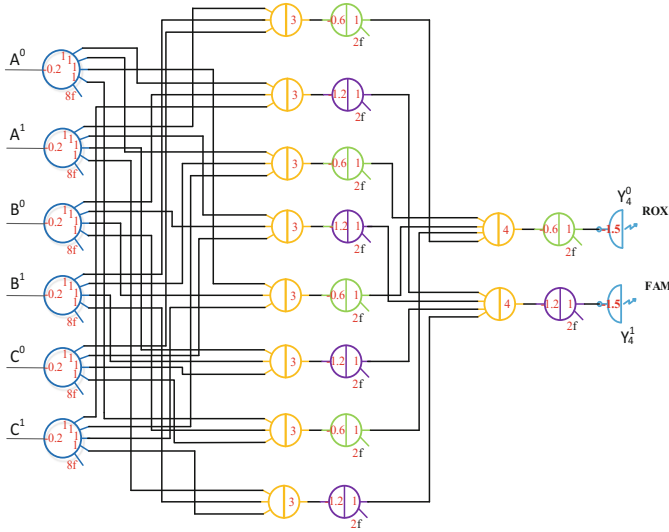


Fig. 5. The seesaw circuit diagram of Y_4

4 Seesaw Circuit Simulation with Visual DSD

Visual DSD is a software specially designed for DNA strand displacement, that is based on the programming language of DNA circuit, in which there is a series of basic factors contained, such as sequence field, toe and branch migration. The interface is composed of setting section, coding section and display section. In the setting section, it is setting for simulation condition, condition results and molecular model visual effect. In the coding section, it is usually for devising DNA molecular structure through compile program; setting parameter. For example: the reaction time, the information collecting times, the molecular combine, the separate time, and so on. Including molecular computing simulation of the whole process, the changing curve graph of DNA molecular, the initial state and terminate state of molecular model.

In the simulation diagrams of Visual DSD, the abscissa indicates the reaction time and the ordinate indicates the reaction concentration. The simulation time of the experiment was set to 40,000 s and the concentration was set to 100(nM). The output signal concentration OFF threshold is set to “0.1x” and the corresponding ON threshold is set to “0.9x”. Then the simulation is done by changing the input in the Visual programming language. Figure 6(a)–(h) are the eight output simulation results for input 000-111. The simulation output results are consistent with the expected output of the circuit design, so the design of this complex logic circuit is feasible.

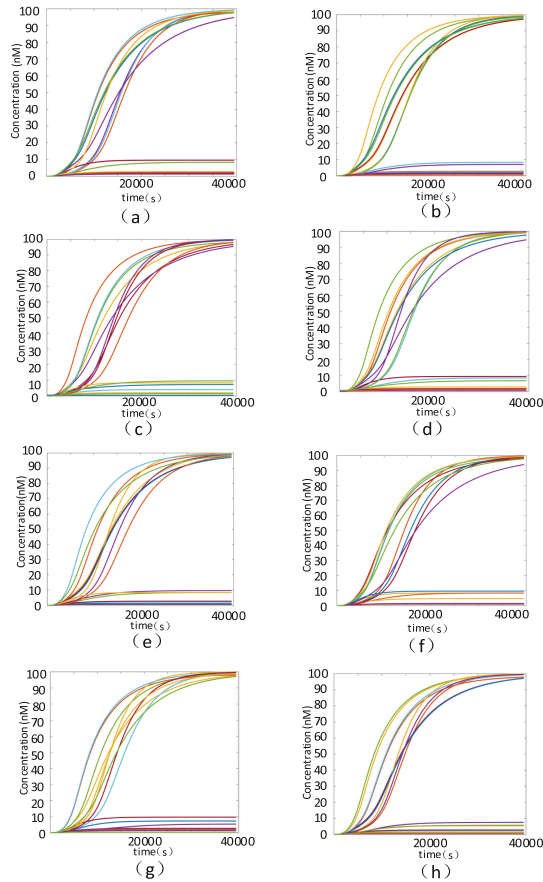


Fig. 6. Seesaw cascade circuit of three-input and nine-output

5 Conclusion

Based on the theory of DNA self-assembly calculation, this paper first constructs the corresponding DNA molecular structure model of the basic gate circuit in the logic operation. On this basis, combined with the principle of DNA strand displacement reaction, a complex logic circuit of the required design was constructed. The feasibility and accuracy of the designed logic circuit model are verified by Visual DSD software. It has been verified that logic circuits constructed by DNA strand displacement techniques can reliably generate all correct output combinations through different input signals. The above results indicate that DNA strand displacement reaction has broad application prospects in theoretical research and practical operation. With the continuous advancement of biological experimental technology, it will vigorously promote the development of DNA computing, and provide new ideas for cryptography, computer science and medical research. At the same time, it will make greater contributions to the development of research and application of nanoscale molecular logic circuits.

Acknowledgments. The work is supported by the State Key Program of National Natural Science of China (Grant No. 61632002), the National Key R&D Program of China for International S&T Cooperation Projects (No. 2017YFE010 3900), the National Natural Science of China (Grant Nos. 61603348, 61775198, 61603347, 61572446, 61472372), Science and Technology Innovation Talents Henan Province (Grant No. 174200510012), Research Program of Henan Province (Grant Nos. 172102210066, 17A120005, 182102210160), Youth Talent Lifting Project of Henan Province and the Science Foundation of for Doctorate Research of Zhengzhou University of Light Industry (Grant No. 2014BSJJ044).

References

1. Grzelczak, M., Vermant, J., Furst, E.M., Liz-Marzan, L.M.: Directed self-assembly of nanoparticles. *ACS Nano* **4**(7), 3591–3605 (2010)
2. Modi, S., Bhatia, D., Simmel, F.C., Krishnan, Y.: Structural DNA nanotechnology: from bases to bricks, from structure to function. *J. Phys. Chem. Lett.* **1**(13), 1994–2005 (2010)
3. Wing, J.M.: Computational thinking and thinking about computing. *Philos. Trans. R. Soc. Lond. A: Math. Phys. Eng. Sci.* **366**(1881), 3717–3725 (2008)
4. Kitano, H.: Computational systems biology. *Nature* **420**(6912), 206 (2002)
5. Ezziane, Z.: DNA computing: applications and challenges. *Nanotechnology* **17**(2), R27 (2005)
6. Bakar, R.B.A., Watada, J., Pedrycz, W.: DNA approach to solve clustering problem based on a mutual order. *Biosystems* **91**(1), 1–12 (2008)
7. Seeman, N.C.: DNA in a material world. *Nature* **421**(6921), 427 (2003)
8. Thiruvengadathan, R., Korampally, V., Ghosh, A., Chanda, N., Gangopadhyay, K., Gangopadhyay, S.: Nanomaterial processing using self-assembly-bottom-up chemical and biological approaches. *Rep. Progress Phys.* **76**(6), 066501 (2013)
9. Amodio, A., Zhao, B., Porchetta, A., Idili, A., Castronovo, M., Fan, C.: Rational design of pH-controlled DNA strand displacement. *J. Am. Chem. Soc.* **136**(47), 16469–16472 (2014)
10. Orbach, R., Guo, W., Wang, F., Iioubasherski, O., Willner, I.: Self-assembly of luminescent Ag nanocluster-functionalized nanowires. *Langmuir* **29**(42), 13066–13071 (2013)
11. Zhang, D.Y., Seelig, G.: Dynamic DNA nanotechnology using strand-displacement reactions. *Nat. Chem.* **3**(2), 103 (2011)
12. Qian, L., Winfree, E.: Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**(6034), 1196–1201 (2011)
13. Qian, L., Winfree, E., Bruck, J.: Neural network computation with DNA strand displacement cascades. *Nature* **475**(7356), 368 (2011)
14. Linko, V., Dietz, H.: The enabled state of DNA nanotechnology. *Curr. Opin. Biotechnol.* **24**(4), 555–561 (2013)
15. Lin, J., Dyer, C.: Data-intensive text processing with mapreduce. *Synth. Lect. Hum. Lang. Technol.* **3**(1), 1–177 (2010)
16. Ito, H., Konishi, R., Nakada, H., Tsuboi, H., Okuyama, Y., Nagoya, A.: Dynamically reconfigurable logic LSI: PCA-2. *IEICE Trans. Inf. Syst.* **87**(8), 2011–2020 (2004)
17. Maojo, V., Martin-Sanchez, F., Kulikowski, C., Rodriguez-Paton, A., Fritts, M.: Nanoinformatics and DNA-based computing: catalyzing nanomedicine. *Pediatr. Res.* **67**(5), 481 (2010)
18. Zang, L., Che, Y., Moore, J.S.: One-dimensional self-assembly of planar π -conjugated molecules: adaptable building blocks for organic nanodevices. *Acc. Chem. Res.* **41**(12), 1596–1608 (2008)

19. Ke, Y., Ong, L.L., Shih, W.M., Yin, P.: Three-dimensional structures self-assembled from DNA bricks. *Science* **338**(6111), 1177–1183 (2012)
20. Rothemund, P.W.K.: Folding DNA to create nanoscale shapes and patterns. *Nature* **440** (7082), 297 (2006)
21. Zhang, D.Y., Winfree, E.: Control of DNA strand displacement kinetics using toehold exchange. *J. Am. Chem. Soc.* **131**(47), 17303–17314 (2009)
22. Zhang, D.Y., Hariadi, R.F., Choi, H.M.T., Winfree, E.: Integrating DNA strand-displacement circuitry with DNA tile self-assembly. *Nat. Commun.* **4**, 1965 (2013)
23. Fujibayashi, K., Hariadi, R., Park, S.H., Winfree, E., Murata, S.: Toward reliable algorithmic self-assembly of DNA tiles: a fixed-width cellular automaton pattern. *Nano Lett.* **8**(7), 1791–1797 (2007)
24. Li, W., Yang, Y., Yan, H., Liu, Y.: Three-input majority logic gate and multiple input logic circuit based on DNA strand displacement. *Nano Lett.* **13**(6), 2980–2988 (2013)
25. Genot, A.J., Bath, J., Turberfield, A.J.: Reversible logic circuits made of DNA. *J. Am. Chem. Soc.* **133**(50), 20080–20083 (2011)
26. Frezza, B.M., Cockroft, S.L., Ghadiri, M.R.: Modular multi-level circuits from immobilized DNA-based logic gates. *J. Am. Chem. Soc.* **129**(48), 14875–14879 (2007)
27. Wang, Y., Tian, G., Hou, H., Ye, M., Cui, G.: Simple logic computation based on the DNA strand displacement. *J. Comput. Theor. Nanosci.* **11**(9), 1975–1982 (2014)
28. Bath, J., Turberfield, A.J.: DNA nanomachines. *Nature Nanotechnol.* **2**(5), 275 (2007)



Modified Mixed-Dimension Chaotic Particle Swarm Optimization for Liner Route Planning with Empty Container Repositioning

Mingzhu Yu¹, Zhichuan Chen¹, Li Chen^{2(✉)}, Rong Qu³,
and Ben Niu^{2(✉)}

¹ Department of Transportation Engineering, College of Civil Engineering,
Shenzhen University, Shenzhen 518060, China

² College of Management, Shenzhen University, Shenzhen 518060, China
C1.jx2005@163.com

³ School of Computer Science, University of Nottingham,
Nottingham NG8 1BB, UK

Abstract. Empty container repositioning has become one of the important issues in ocean shipping industry. Researchers often solve these problems using linear programming or simulation. For large-scale problems, heuristic algorithms showed to be preferable due to their flexibility and scalability. In this paper we consider large-scale the liner routing planning problem with empty container repositioning (LRPECR) model where allocation strategies and liner routes need to be designed to allocate empty containers from the supply ports to the demand ports. According to the characteristics of the LRPECR model, we combine the path of the ship to the algorithm encoding, set up the fitness function that minimizes the total cost, and use a modified Particle Swarm Optimization (PSO) algorithm to search for optimal shipping routes in a feasible space iteratively. The modified PSO combines chaotic theory and Cat map to overcome the defect of traditional PSO. In addition, we perform chaotic search in different dimensions to enhance the search accuracy of the algorithm that means the increased diversity of search scope. In order to validate our algorithm, standard PSO and GA are chosen as the compared algorithms. Through numerical studies based on real applications, the experimental results demonstrate that the modified PSO is able to find preferable solutions efficiently for the empty container repositioning problem.

Keywords: Empty container repositioning · Chaotic search
Particle swarm optimization · Cat map · Integer linear programming

1 Introduction

The liner routing problem (LRP) aims to establish a reasonable liner service shipping network between several supply and demand ports. Considering the transportation cost, risk and shipping capacity, shipping routes should be planned between ports. LRP is a combinatorial optimization problem where the ocean carriers plan and deploy the liner routes rationally to satisfy customer demands and to maximize profit.

As an essential issue in contemporary LRP, the efficient and effective repositioning of empty containers can significantly reduce the operation costs of shipping companies. At the same time, it also has positive effect on environment protection and sustainability [1, 2]. Shintani and Imai [3] solved the empty containers repositioning using a Genetic Algorithm (GA) for the first time. Based on this, Sun [4] proposed a model for empty container repositioning and solved it using hybrid genetic algorithm (HGA).

Particle Swarm Optimization (PSO) [5, 6] is one of the most widely-used evolutionary algorithms. However, traditional PSO is prone to suffer from being trapped in local optima, leading to premature convergence [7]. To overcome this defect, many improvements have been proposed, including chaotic search, one of the powerful hybrid algorithms. Chaotic searching algorithm was proposed first by Changkyu et al. [8]. Liu et al. [9] developed an improved PSO combined with chaotic searching algorithm. Tan et al. [10] found that the single-dimension chaotic search can evidently improve the algorithm precision and the efficiency of the chaotic search. On the other hand, Logistic map is frequently used for generating chaos sequence in the majority of chaotic search algorithms. Based on the research in the field of image encryption [11], Wang et al. [12] replaced the Logistic map with Cat map as the chaos sequence generator. The superior properties of Cat map, which are excellent ergodicity and sensitive dependence on initial conditions, were taken by Wang in chaotic search algorithm. Cat map overcomes the disadvantages of Logistic map, including non-uniformity and frequent loop.

Based on above mentioned researches, this paper proposes to solve the liner route planning problem considering empty container repositioning (LRPECR) [4] using a new modified PSO, which assimilated the experiences of chaotic search with Cat map [12] and single-dimensional and multi-dimensional search [10]. We name it as Modified Mixed-dimension Chaotic Particle Swarm Optimization (MDCPSO). The optimization result of MDCPSO are compared with those of standard PSO, GA and HGA.

The rest of the article is organized as follows. Section 2 provides a brief description of MDCPSO. Section 3 presents the LRPECR model and problem descriptions. The results and analysis of the experiments are shown and discussed in Sect. 4. Finally, the concluding remarks are provided in Sect. 5.

2 Modified Mixed-Dimension Chaotic Particle Swarm Optimization

2.1 Particle Swarm Optimization

PSO is a collection of intelligence optimization techniques. The system is initialized with a set of random solutions called “particles”, which move through the search space towards the optimal location by iterations. The search of PSO aims to strike a balance between exploration and exploitation. For more detailed information, please refer to [6].

2.2 Modified Mixed-Dimension Chaotic Particle Swarm Optimization

The standard PSO and GA show to usually suffer from being trapped into local optima. Inspired by the research by Tan and Wang [11, 12], the MDCPSO algorithm has been developed to enhance the performance of the standard PSO. The operators of MDCPSO include the chaotic search with cat map, and multi-dimension and single-dimension chaotic search methods.

Two Key Mechanisms of MDCPSO. Before describing how MDCPSO solves the LRPECR model, a brief introduction of two key mechanisms MDCPSO, namely the chaotic search mechanism and multi-dimension and single-dimension chaotic search, is given as follows.

Chaotic Search Mechanism with Cat Map. In order to improve the capability of PSO to escape from the local optimum, the chaotic search is used for constructing the algorithm. Logistic map is used in chaotic search frequently. It can map a number to a set of 0 to 1. Through multiple iterations, the values mapped by Logistic equation will traverse the whole set to achieve chaotic effect. Equation (1) shows the equation of Logistic map. $cx^{(0)}$ is a chaotic variable generated by the global best particles in the PSO system. And n represents the current chaotic iterations.

$$cx^{(n+1)} = 4cx^{(n)}(1 - cx^{(n)}), 0 < cx^{(0)} < 1. \quad (1)$$

In this paper, the chaotic Cat map is used to replace the traditional chaotic Logistic map. Compared with the Logistic map, Cat map as a chaos sequence generator shows to enrich the chaotic search behavior, because it can traverse the whole set of 0 to 1 faster [12]. Equation (2) shows the computational formula of Cat map.

$$\begin{cases} cx^{(n+1)} = (cx^{(n)} + y^{(n)}) \bmod 1, 0 < cx^{(0)} < 1, 0 < y^{(0)} < 1. \\ y^{(n+1)} = (cx^{(n)} + 2y^{(n)}) \bmod 1 \end{cases} \quad (2)$$

Here, *mod* is the modulus operator, and the result will be return to $cx^{(n+1)} \cdot y^{(0)}$. will be created randomly.

Multi-dimension and Single-Dimension Chaotic Search Method. Multi-dimension chaotic search means to map the value of all dimensions by chaotic map function. The existing chaos search method mainly use multi-dimensional chaotic search. However, compared to multi-dimensional chaotic search, single-dimensional search can produce a better search accuracy, because it only changes the value of one single dimension, increasing diversity of search scope. Combining the two different mechanisms, we propose a new method as follows.

Step 1: Initialize parameters of the chaotic system, with a random iteration number $y^{(0)}$ ($y^{(0)} \in [0, 1]$), iteration counter $n = 1$, the maximum iteration number M of the chaotic system, and a specified input solution x_{spec} .

Step 2: Mapping the specified solution x_{spec} to a set between [0, 1] according Eq. (3). $x_{max,i}$, $x_{min,i}$ represent the maximum and minimum values for each dimension of x_{spec} , respectively. d is the maximum dimension of x_{spec} .

$$cx_i^{(1)} = \frac{x_{spec,i} - x_{min,i}}{x_{max,i} - x_{min,i}}, i = 1, 2, \dots, d \tag{3}$$

Step 3: Generate a chaotic sequence $cx^{(n+1)}$ by Cat map using Eq. (2).

Step 4: Generate a random number r , r is between [0, 1].

Step 5: If $r < 0.5$, a multi-dimension chaotic local search is performed on $cx^{(n)}$ according to Eq. (4). Here, c represents one of the solution sequences generated by the chaotic search; α is the step length of the chaotic search. In this work, α is set to a random integer number between [-2, 2].

$$c_i^{(n)} = \alpha \cdot cx_i^{(n+1)} + x_{spec,i}, i = 1, 2, \dots, d \tag{4}$$

Step 6: If $r \geq 0.5$, a single-dimension chaotic local search is performed on $cx^{(n)}$ according to Eq. (5). What is different from **Step 5** is that the single-dimension search does not need to perform chaotic search on each dimension. It only needs to randomly select one of the dimensions to perform the chaotic search.

$$c_i^{(n)} = \alpha \cdot cx_i^{(n+1)} + x_{spec,i}, i = randint(1, d) \tag{5}$$

Step 7: Set iteration counter as $n + 1$. Go to Step 3 if $n \leq M$.

Step 8: Calculate fitness values of all solutions in c and store the best one.

Step 9: Replace x_{spec} by the best solution in c if the fitness value of the best solution in c is better than that of x_{spec} , otherwise x_{spec} will be maintained as the best solution.

Step 10: Stop the chaotic search and output the best solution.

3 MDCPSO for Liner Route Planning

3.1 The LRPECR Model

In this paper the same LRPECR model proposed by Sun in [4] is considered. In the LRPECR model, assignments of repositioning empty containers are required to the known demand ports. The LRPECR problem aims to design liner routes considering the satisfaction of empty containers which need to be transferred from the supply ports to the demand ports. The objective of this model is to plan the liner route with the minimized total cost. The problem is a mixed integer linear programming, and was solved by Sun in [4] using a hybrid GA. The LRPECR model is presented in Eqs. (6–12) and the variable definitions are shown in Table 1.

Mathematically, the objective is to minimize the total cost. The first item of the objective function represents all the costs, including loading cost, unloading cost and

Table 1. Parameters and definitions of LRPECR model

Variables	Definitions
I	The set of all container ports
M	The set of all empty container demand ports
X_{Tj}	The set of all alternative shipping liners to the demand port $j, j \in M$
C_{Lij}	The unit cost of loading an empty container from port i to demand port $j, i \in I, j \in M$
C_{Tij}	The unit cost of repositioning an empty container from port i to demand port j through the shipping liner $t, i \in I, j \in M, t \in X_{Tj}$
C_{Uij}	The unit cost of unloading an empty container from port i to demand port $j, i \in I, j \in M$
C_{Rj}	The unit cost of renting and loading an empty container in port $j, j \in M$
C_{Sj}	The unit cost of repositioning an empty container from the container renter to port $j, j \in M$
D_{Ej}	The demand volume of empty container in demand port $j, j \in M$
S_{Ni}	The volume of available empty container in port $i, i \in I$
Decision variables	Definitions
x_{ij}^E	The volume of empty container repositioned from port i to demand port $j, i \in I, j \in M$
x_{ij}	$x_{ij} = 1$ means that shipping liner t is chosen from X_{Tj} , and port i serves as the starting port, $i \in I, j \in M, t \in X_{Tj}$
x_j^R	The volume of empty container rented from renter to port $j, i \in I, j \in M$

the cost of shipping a container from port i to the known demand port. The second item means the cost of renting containers in port i .

$$\min z = \sum_{j \in M} \sum_{i \in I} \sum_{t \in X_{Tj}} ((C_{Tij} + C_{Lij} + C_{Uij}) \times x_{ij}^E + (C_{Sj} + C_{Rj}) \times x_j^R)$$

Subject to:

$$x_{ij}^E + x_j^R = D_{Ej}, \forall i \in I, \forall j \in M \tag{6}$$

$$\sum_{j \in M} x_{ij}^E \leq S_{Ni}, \forall i \in I \tag{7}$$

$$x_{ij}^E, x_j^R \geq 0, \forall i \in I, \forall j \in M \tag{8}$$

$$\sum_{t \in X_{Tj}} x_{ij} \leq 1, \forall i \in I, \forall j \in M \tag{9}$$

$$\sum_{t \in X_{Tj}} x_{ij} \times S_{Ni} \geq x_{ij}^E, \forall i \in I, \forall j \in M \tag{10}$$

$$M \times x_{ij}^E \geq \sum_{t \in X_{Tj}} x_{ij}, \forall i \in I, \forall j \in M \tag{11}$$

$$x_{ij} = \{0, 1\}, \forall i \in I, \forall t \in X_{Tj}, \forall j \in M \tag{12}$$

Equation (6) means that the volume of empty containers repositioned and rented at a certain port should equals to the empty container demand. In Eq. (7) the repositioning volume should be no more than the volume that the port can supply. Equation (8) ensures that the volume of transportation should not be less than 0. Equation (9) defines that there is only one shipping route for one assignment. According to Eqs. (10–11), once the assignment between port i to port j is determined, the transfer volume of empty containers between them must be more than 0, and the first M represents a sufficiently large number. Equation (12) restricts the value of x_{ij} to be 0 or 1.

3.2 MDCPSO for LRPECR Model

Encoding. A new encoding strategy is developed considering the characteristic of the LRPECR model. According to real problem data, a vessel will not berth at more than four ports in one shipping mission, because of the high cost. It is better to adopt the strategy of renting containers at the demand port. Therefore, a sequence of four numbers is used to represent a liner route for one shipping mission. When there are m shipping missions, a particle with $4 \times m$ dimensions is used.

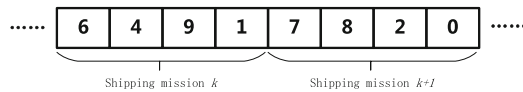


Fig. 1. An example for the encoding scheme for LRPECR

As Fig. 1 shows, in shipping mission k , four positions represent the ports on this route from the supply port to the demand port. The first position represents the demand port, and the second position is for the linked port of the demand port, and so on. The shipping route of mission k is thus port 1 → port 9 → port 4 → port 6. If the number of ports on the liner route is less than four (as shown in shipping mission $k + 1$), the rest of positions will be filled with 0. The shipping route of mission $k + 1$ is port 2 → port 8 → port 7 in Fig. 1.

Fitness. The location of every particle is different in the search space. The fitness is calculated according to Eq. (13). A greater fitness value means that the particle stays in a better location in the search space, so it will be preserved and to be adapted by the particles with a higher probability in the next iteration. Those particles with poor fitness values will be improved or even eliminated. In this paper, the fitness equation is the same as the objective function.

$$\text{fitness} = \sum_{j \in M} \sum_{i \in I} \sum_{t \in X_{Tj}} (x_{ijt} \times (C_{Titj} + C_{Lij} + C_{Uij}) \times x_{ij}^E + (C_{Sij} + C_{Rij}) \times x_{ij}^R) \quad (13)$$

Computational Steps of the MDCPSO Algorithm for LRPECR. Based on the mechanisms described above, an improved PSO algorithm is designed for solving LRPECR, and is implemented in MATLAB. The standard PSO and GA are chosen as the compared algorithms. In order to make fair comparisons, all algorithms compared use the same encoding, population sizes and iteration numbers. The length of encoding, population size and iteration number L are all set to 200, 50 and 1000. In both PSO and MDCPSO, learning factors $C1$ and $C2$ are set to 1.5. In GA, crossover rate Pc and mutation rate Pm are set to 0.8 and 0.09. The pseudo-code of MDCPSO for LRPECR is shown in Table 2.

Table 2. The pseudo-code of MDCPSO

Begin
Initialize parameters;
Initialize a population of particles with random locations and velocities in a domain of feasible solution, and evaluate fitness value f_i for each particle according to Eq. (13);
Initialize P_g with the best particle within the population;
Initialize P_i with a copy of each particle's location;
For ($l = 1 : L$):
Update velocities and locations for each particle;
Adjust the location for those particles beyond the boundary of the domain;
Evaluate the fitness f_i for all particles;
Update the P_g and P_i ;
Perform the mixed-dimension chaotic search on P_g (see Section 2.2 <i>Multi-dimension and Single-dimension Chaotic Search Method.</i>);
End
Output: the best solution for the LRPECR model

4 Experimental Results and Analysis

4.1 Experimental Data

The characteristics of the numerical examples are set the same as those in the LRPECR model in Sun [4]. Computational studies have been conducted on 5 instances. The total number of ports and missions in Instance 1 is 28 and 50, respectively.

Figure 2 shows the distribution of the ports and empty container status for each port in Instance 1. There are 28 ports distributed in this network. The number 0 means that the port needs to be supplemented by transportation form supply ports or renting. In Sun's case study, it was found that the cost of adopting a full leasing strategy was lower

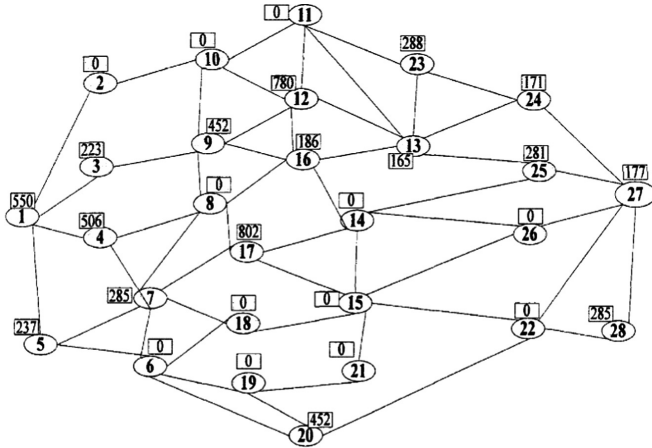


Fig. 2. Distribution and empty container status in Instance 1, Sun [4]

than the cost obtained by their optimization. The original data has thus been adjusted by increasing the cost of renting empty containers. There are 50 missions of assignment, and the target port and other details of parameters are shown in Appendix. For all instances 10 runs are conducted to obtain the average objective values and computation times.

4.2 Results and Analysis

Figure 3 and Table 3 show the results of Instance 1. MDCPSO shows to outperform GA and standard PSO. In the first 50 iterations, results from MDCPSO have exceeded the other two. In Fig. 3 we can observe that the convergence speed of MDCPSO is

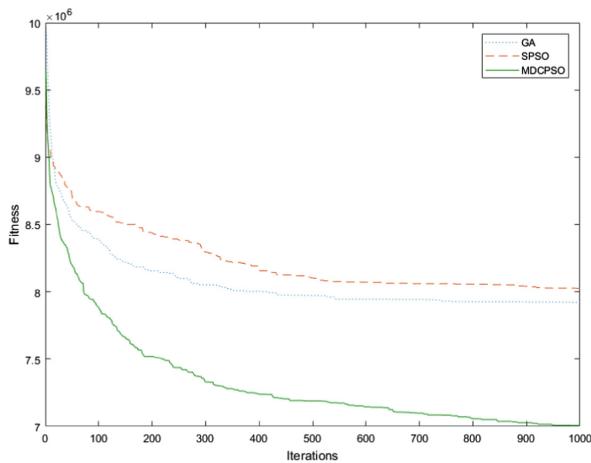


Fig. 3. Convergence of GA, PSO and MDCPSO for Instance 1

much faster than GA and PSO. That means MDCPSO can find a better solution in a short period of time. After 300 generations, both GA and PSO show to have converged, however MDCPSO still maintains an improving trend towards better solutions.

Table 3. The average results for Instance 1

	GA	SPSO	MDCPSO
Average fitness	7.9214e+06	8.0269e+06	7.0075e+06

Table 4. The optimized empty container allocation strategies based on MDCPSO of mission 1 to 16

Mission	Route	x_{ij}^E	x_j^R	Mission	Route	x_{ij}^E	x_j^R
1	1-1	260	0	9	5-5	97	13
2	2-2	0	500	10	6-6	0	100
3	2-2	0	340	11	6-6	0	110
4	3-3	130	0	12	6-6	0	310
5	1-3	290	100	13	6-6	0	150
6	3-3	93	37	14	16-8-7	150	0
7	4-8-9-3	450	0	15	7-7	285	35
8	5-5	140	0	16	9-9	300	0

Table 5. The optimized empty container allocation strategies based on MDCPSO of mission 17 to 50

Mission	Route	x_{ij}^E	x_j^R	Mission	Route	x_{ij}^E	x_j^R
17	9-9	152	238	34	20-20	170	0
18	9-9	0	260	35	20-20	60	0
19	12-12	230	0	36	17-15-21	380	0
20	12-12	180	0	37	17-14-15-21	110	0
21	13-13	90	0	38	17-15-21	172	8
22	12-13-24-13	230	0	39	20-22	170	0
23	4-8-16-14	56	44	40	28-22	285	215
24	13-25-14	36	254	41	23-23	288	190
25	12-16-14	140	50	42	23-23	0	190
26	13-25-14	75	195	43	23-23	0	170
27	15-15	0	200	44	24-24	90	0
28	15-15	0	120	45	24-24	81	129
29	15-15	0	80	46	25-25	281	149
30	15-15	0	330	47	25-25	0	150
31	17-17	140	0	48	26-26	0	230
32	17-17	0	100	49	27-27	177	193
33	18-18	0	110	50	20-22-27	52	158

MDCPSO has been run 10 times in MATLAB, leading to the best solution of \$6,845,666, while the best result from HGA by Sun is \$11,387,000 [4]. Note that the renting cost of container has been raised based on Sun’s model. MDCPSO shows to perform better than HGA in solving the RLPECR model. The detailed liner route plan and strategies of the empty container allocation are shown in Tables 4 and 5.

In order to confirm the effectiveness of MDCPSO in different scale of RLPECR, another four instances with different numbers of ports and assignments have been tested. They respectively are Instance 2 (35 assignments in 25 ports), Instance 3 (65 assignments in 31 ports), Instance 4 (80 assignments in 34 ports) and Instance 5 (95 assignments in 37 ports).

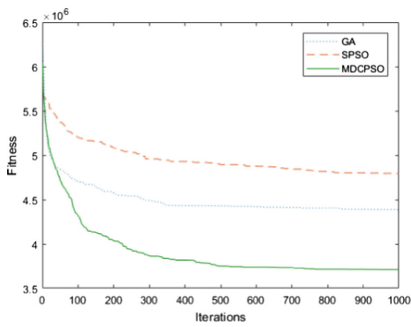


Fig. 4. Comparisons of algorithms for Instance 2

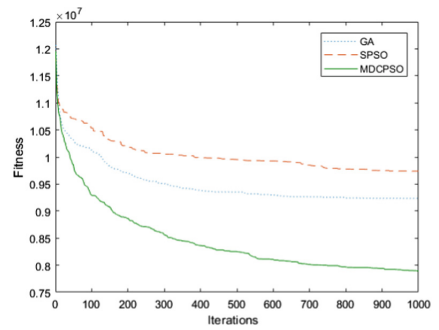


Fig. 5. Comparisons of algorithms for Instance 3

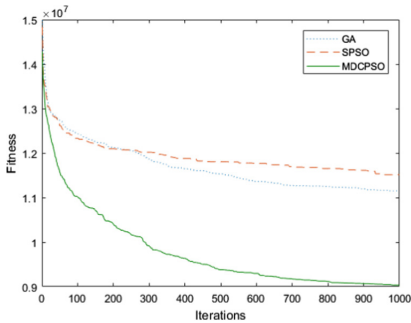


Fig. 6. Comparisons of algorithms for Instance 4

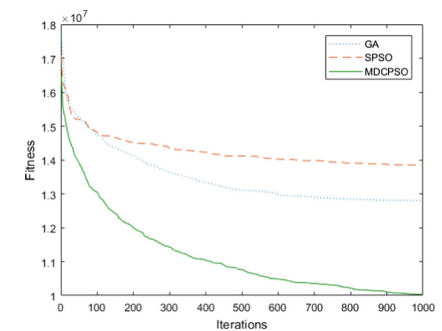


Fig. 7. Comparisons of algorithms for Instance 5

Based on above experiment results in Figs. 4, 5, 6 and 7 and Table 6, we can conclude that MDCPSO performs significantly better than PSO and GA in all instances. In addition, the convergence of standard PSO has shown to be always inferior to GA. When the mixed-dimension chaotic search with Cat map is integrated with PSO, the improvement is significant. It proves the effectiveness of those mechanisms on PSO.

Table 6. The average results from 10 runs of the three algorithms

Scale	GA	SPSO	MDCPSO
Instance 2	4.3839e+06	4.7942e+06	3.7108e+06
Instance 3	9.2304e+06	9.7392e+06	7.8939e+06
Instance 4	1.1147e+07	1.1514e+07	9.0342e+06
Instance 5	1.2797e+07	1.3845e+07	1.0034e+07

5 Conclusions

This paper proposed an improved new particle swarm optimization (PSO) algorithm, namely MDCPSO, for solving the liner routing planning problem with empty container repositioning (LRPECR) model based on chaotic PSO. MDCPSO employs the powerful search capability of the chaotic algorithm with Cat map and the superior searching precision of the mixed-dimension search. In order to evaluate the effectiveness of MDCPSO, two widely used algorithms, GA and PSO, are compared. The experimental results show that the performance of MDCPSO is outstanding in solving LRPECR. In the future, the key mechanisms of MDCPSO will be combined or integrated with other heuristic algorithms for addressing the LRPECR model with extended constraints.

Acknowledgment. This work is partially supported by The National Natural Science Foundation of China (Grants Nos. 71571120, 71271140, 61472257), Natural Science Foundation of Guangdong Province (2016A030310074).

References

1. Dong Ping, S., Jonathan, C.: Empty container repositioning in liner shipping. *Marit. Policy Manag.* **36**(4), 291–307 (2009)
2. Yoonjea, J., Subrata, S., Debajypti, C., Ilkyeong, M.: Direct shipping service routes with an empty container management strategy. *Transp. Res. Part E Logist. Transp. Rev.* **118**, 123–142 (2018)
3. Shintani, K., Imai, A.: The container shipping network design problem with empty container repositioning. *Transp. Res. Part E Logist. Transp. Rev.* **43**(1), 39–59 (2007)
4. Sun, J.: Optimizing empty container allocation based on hybrid genetic algorithm. Dalian Maritime University (2009). in Chinese
5. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: *IEEE International Symposium on MICRO Machine and Human Science*, pp. 39–43 (2002)
6. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: *1995 Proceedings of IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948 (1995)
7. Van den Bergh, F.: An analysis of particle swarm optimization. Ph.D. dissertation, Department of Computer Science, University of Pretoria, Pretoria (2006)
8. Changkyu, C., Ju-Jang, L.: Chaotic local search algorithm. *Artif. Life Robot.* **2**(1), 41–47 (1998)
9. Liu, B., Wang, L., Jin, Y.H., Tang, F., Huang, D.X.: Improved particle swarm optimization combined with chaos. *Chaos Solitons Fractals* **25**(5), 1261–1271 (2005)

10. Tan, Y., Tan, G.Z., Deng, S.G.: Hybrid particle swarm optimization with differential evolution and chaotic local search to solve reliability-redundancy allocation problems. *J. Cent. South Univ. (Engl. Ed.)* **20**(6), 1572–1581 (2013)
11. Chen, G., Mao, Y., Chui, C.K.: A symmetric image encryption scheme based on 3D chaotic cat maps. *Chaos Solitons Fractals* **21**(3), 749–761 (2004)
12. Wang, F., Dai, Y.S., Wang, S.S.: Modified chaos-genetic algorithm. *Comput. Eng. Appl.* **46**(6), 29–32 (2010)



A Wrapper Feature Selection Algorithm Based on Brain Storm Optimization

Xu-tao Zhang¹, Yong Zhang^{2(✉)}, Hai-rong Gao³, and Chun-lin He²

¹ Department of Electrical Engineering, Jiangsu College of Safety Technology, Xuzhou, China

² School of Information and Control Engineering, China University of Mining and Technology, Xuzhou, China
Yongzh401@126.com

³ Department of Transportation Engineering, Shandong Transport Vocational College, Tai'an, China

Abstract. Feature selection is an important preprocessing technique of data, which can be generally modeled as a binary optimization problem. Brain storm optimization (BSO) is a newly proposed algorithm that has not been systematically applied to feature selection problems yet. This paper studies an effective wrapper feature selection method based on BSO. Focused on this goal, firstly, a selective probability-based real encoding strategy of individual is introduced to transform the binary feature selection problem into a continuous optimization one. Based on this, then a continuous BSO-based feature selection algorithm (CBSOFS) is proposed. The proposed algorithm is tested on standard benchmark datasets and then compared to four representative algorithms. Experimental results show that CBSOFS achieves comparable results with compared algorithms.

Keywords: Feature selection · Brain storm optimization
Continuous encoding

1 Introduction

Many real-world applications involve more and more attributes (features) as their capabilities in acquiring and storing information increase. Among these features, many are irrelevant or/and redundant, because it is difficult for decision-makers to determine which one is useful in advance [1, 2]. The goal of Feature Selection (FS) is to select a feature subset from original feature set, and the selected subset should be necessary and sufficient to describe a target concept [3].

Existing FS methods can be generally classified into the three categories, i.e., the filter, the wrapper, and the hybrid approaches [4, 5]. The filter approach selects key features according to their ranks which are calculated by a series of criteria. This kind of approach is computation-efficient. The wrapper approach involves a learning algorithm determined in advance, which is evaluated by the selected feature subset [6]. Compared with the filter approach which is independent of any learning algorithm, the wrapper approach usually has better performances in most cases [5]. The hybrid approach mainly studies the combination of the filter and the wrapper approaches.

Focused on the wrapper approach, there have been many methods to seek an optimal feature subset [5]. Recently, nature-inspired algorithms have received much attention on seeking optimal feature subsets, because of their global search performance [7]. Part approaches include genetic algorithm (GA) [8, 9], differential evolution [10], ant colony optimization (ACO) [11, 12], bee colony optimization (BCO) [14], firefly algorithm [15, 30], particle swarm optimization (PSO) [16–19].

In recent years, Shi developed a new nature-inspired algorithm, namely brain storm optimization algorithm (BSO), based on the collective behavior of human being, [20, 21]. Later, BSO has been applied to solve wind speed forecasting in power dispatching problem [22], stock price forecasting [23], Loney’s Solenoid Problem [24], and so on. However, to the best of our knowledge, no one has yet applied this algorithm to feature selection problems. This motivates our attempts to investigate the efficiency of BSO on dealing with feature selection.

In this paper, a BSO-based wrapper feature selection algorithm is studied to find optimal feature subsets. Focused on this goal, a selective probability-based real encoding strategy of individual is introduced to transform a binary feature selection problem into a continuous optimization one. Based on this, a continuous BSO-based feature selection algorithm (CBSOFS) is proposed. Finally, the proposed feature selection algorithm will be tested and compared with other algorithms.

This paper is organized as follows. Section 2 gives related work. The proposed algorithm is given in Sect. 3. The efficiency of the proposed algorithm is tested in Sect. 4. Finally, the paper is concluded in Sect. 5.

2 Related Work

2.1 Feature Selection

Considering a data set S which contains K samples and D features, a FS problem can be described as follows: to select d features ($d \leq D$) from all the features, so that an appointed function $H(\cdot)$, such as the classification accuracy, are optimized. We adopt a binary string to encode a solution in FS problems:

$$Z = (z_1, z_2, \dots, z_D), \quad z_j \in \{0, 1\} \quad (1)$$

where $z_j = 1$ indicates the j -th feature is selected into the subset Z ; otherwise, $z_j = 0$. Furthermore, a FS problem is formulated as follows:

$$\begin{aligned} & \max/\min H(Z) \\ & \text{s.t. } Z = (z_1, z_2, \dots, z_D), \quad z_j \in \{0, 1\}, \quad j = 1, 2, \dots, D, \\ & \quad 1 \leq |Z| \leq D, \quad H(Z) \in [0, 1]. \end{aligned} \quad (2)$$

2.2 Brain Storm Optimization

The BSO algorithm is proposed based on the collective behavior of human being [19]. Generally, BSO includes mainly three phases: individual clustering/classification, new individual generation, and selection.

The Individual Clustering/Classification. In this phase, all individuals are divided into several clusters. Different clustering algorithms can be used. Generally, the basic k -means algorithm is used in the original BSO.

The New Individual Generation. In this phase, a new individual is generated based on cluster centers or individuals in one or two clusters. Here, generating a new individual from one cluster can refine a search region, and it enhances the exploitation capability of the population. On the contrast, generating a new individual from two clusters, which are far away each other, can enhance the exploration ability of the population. In the original BSO, a probability P_{gen} is utilized to determine that one or two clusters are selected. Two probability values, $P_{cluster}$ and $P_{t-cluster}$, are utilized to determine the cluster center or an individual will be chosen from one cluster, respectively. Moreover, an adaptive disturbance is added on the individual generated above to improve the diversity of new individuals, as follow:

$$\begin{aligned} X_i(t) &= X'_i(t) + \zeta(t) \times rand \\ \zeta(t) &= \log sig\left(\frac{0.5 \times T - t}{c}\right) \times rand \end{aligned} \quad (3)$$

Where, $rand$ is a random number within $[0, 1]$; t and T are the current iteration times and the maximal iteration times of BSO, $X'_i(t)$ is the individual generated based on cluster centers or individuals; $X_i(t)$ is the new individual after implement an adaptive disturbance.

The Selection. The selection phase is used to save elite solutions in all individuals. When a new individual is generated by using the phase of the new individual generation, if it is better than the existing individual with the same individual index, then save it and record it as the new individual.

3 The Proposed BSO-Based Feature Selection Algorithm

To solve a binary feature selection problem using BSO algorithm suitable for continuous problems, this section first introduces a selective probability-based real encoding strategy of individual, based on which the binary feature selection problem can be transformed into a continuous optimization one. After that, the continuous BSO-based feature selection algorithm, CBSOFS, is described based on the original BSO algorithm [19]. Of course, other continuous BSO algorithms such as those methods in [24, 25] also are used directly to solving the transformed feature selection problem. The purpose of this paper is to introduce a basic continuous BSO-based feature selection algorithm, so the original BSO algorithm is selected in our proposed CBSOFS.

3.1 The Selective Probability-Based Real Encoding Strategy

First we introduce a selective probability-based real encoding strategy of individual. In a nature-inspired optimization algorithm, an individual position represents a candidate solution of optimized problem. The proposed CBSOFS uses the selective probability-based real encoding strategy to encode an individual position. Note that the real

encoding has been adopted in a variety of literature [16, 17]. In this strategy, the probability of each feature chosen into the feature subset is taken as the encoded element, and multiple elements form a particle which represents a candidate solution of a feature selection problem. Taking a data set with D features as an example, an individual position is represented with a D -bit real string as follows:

$$X_i = (x_{i,1}, x_{i,2}, \dots, x_{i,D}), \quad i = 1, 2, \dots, |S| \quad (4)$$

where $|S|$ is the population size, and $x_{i,j} \in [0, 1]$ is the probability which the j -th feature is chosen into the feature subset. If $x_{i,j} > 0.5$, then the j -th feature is chosen into the feature subset; otherwise, it is not.

Algorithm 1: The proposed CBSOFS

Begin

1. Randomly generate N individuals with real encode in the search space $[0,1]$;
2. Calculate the fitness of each individual, i.e., the classification accuracy of each individual;
3. **While** termination not reached **do**
 % The first phase: the individual clustering/classification%
 4. Cluster N individuals into M clusters using k -means method;
5. Choose the best individual in each cluster as the cluster center;
 % The second phase: the new individual generation %
 6. **If** a random number $rand()$ is less than the probability P_{gen} , **then**
 7. Randomly select a cluster and determine its cluster center ;
 8. **If** a random number $rand()$ is less than the probability $P_{cluster}$,
 9. Select the cluster center;
 10. Add a disturbance to it to generate a new individual by using equation (3) ;
 Else
 11. Randomly select a normal individual from this cluster;
 12. Add a disturbance to it to generate a new individual by using equation (3);
 End if
 Else
 13. Randomly select two clusters to generate new individual
 14. **If** a random number $rand()$ is less than the probability $P_{t-cluster}$, then
 15. Combine the two cluster centers to generate a new individual;
 16. Add a disturbance on the new individual by using equation (3) ;
 Else
 17. Randomly select one normal individual from the two clusters respectively;
 18. Combine the two normal individuals to generate a new individual;
 19. Add a disturbance on the new individual by using equation (3);
 End if
 End if
 % The third phase: the selection strategy %
 20. The newly generated individual is compared with the existing individual with the same individual index; the better one is kept and recorded as the new individual;

End while

End begin

3.2 The Proposed CBSOFS

Based on the above encoding strategy, the original BSO algorithm [19] can be applied to optimize the transformed feature selection algorithm. Algorithm 1 shows steps of the proposed CBSOFS. In the Algorithm 1, the two normal individuals or cluster centers are combined by using an arithmetic crossover-like strategy to generate a new individual. Supposing two normal individuals or cluster centers from the two clusters are X_i and X_j , the existing arithmetic crossover-like strategy is as follows:

$$X_{new} = tem \times X_i + (1 - tem) \times X_j \quad (5)$$

Where, the parameter $tem \in [0, 1]$ is used to control the influence of two participants on the new individual.

4 Experiment and Analysis

This section tests the performance of CBSOFS on 8 benchmark datasets, including Vowel, Wine, Vehicle, Segmentation, WDBC, Ionosphere, Satellite and Sonar. These datasets cover small, medium and large dimensional datasets. Table 1 shows the data information, including the number of examples, features, and classes. For details of these datasets, please see the UCI Repository [26].

Table 1. Brief information of databases

Datasets	# of features	# of samples	# of classes
Vowel	10	990	11
Wine	13	178	3
Vehicle	18	846	4
Segmentation	19	2310	7
WDBC	30	569	2
Ionosphere	34	351	2
Satellite	36	6435	6
Sonar	60	208	2

4.1 Comparison Algorithms and Parameters Setting

Six feature selection methods, including the sequential forward selection method (SFS) [27] and the plus-1 take-away-r (PTA) method [28], the GA-based (GAFS) proposed in [29], the binary PSO algorithm (BPSOFS) in [16]. In our experiments, the same conditions are used to compare CBSOFS with the other nature-inspired algorithms, i.e., the size of the population/swarm = 20, and the 1-nearest neighbor method to evaluate the feature subset. Moreover, we set the same maximum iteration times, $T = 70$, for BPSOFS, BFFAFS and CBSOFS. Furthermore, Table 2 lists the parameter setting of the four nature-inspired algorithms in detail.

Table 2. Parameter settings for nature-inspired algorithms

Comparison algorithms	Values of the other parameters
SGA [29]	The mutation probability = 0.1, the crossover probability = 0.6
BPSO [16]	The acceleration coefficients $c1 = c2 = 2$, the inertia weight = 1
CBSOFS	$P_{gen} = 0.8$, $P_{t-cluster} = 0.5$, $P_{cluster} = 0.5$

4.2 Comparison Results Among Algorithms

The CBSOFS algorithm is compared with SFS, PTA, SGAFS, and BPSOFS on tackling the FS problems of 8 datasets in this subsection. Each algorithm is independently run 30 times for each dataset, and the average values are calculated based on all these runs. To investigate the performances of an algorithm, two measures, i.e. the classification accuracy (Acc) and the number of selected features (NF) are used.

Comparing the classification performance of CBSOFS and the other four algorithms, Table 3 shows the average classification accuracies on the test datasets by 30 independent runs, while Table 4 lists the number of selected features obtained by all the five algorithms. Herein the best classification accuracy for each dataset is shown in boldface. We can see from the Tables 3 and 4 that:

- (1) For seven out of all the eight datasets, CBSOFS all achieved the best classification accuracy values.
- (2) To be specific, for the dataset, Vowel, all the five algorithms obtained the same Acc values. It's not a challenge to CBSOFS, SFS, PTA, SGAFS and BPSOFS. For the datasets, Wine, Vehicle, Segmentation, WDBC, Ionosphere, Satellite, CBSOFS all achieved the best classification accuracy values. Its Acc values are 99.55% for Wine, 76.06% for Vehicle, 98.23% for Segmentation, 98.18% for WDBC, 95.92% for Ionosphere and 91.82% for Satellite, receptively. For the dataset Sonar, CBSOFS also had the second best Acc values. GAFS obtained the best Acc value for the dataset Sonar.
- (3) In terms of the measure NF , the proposed CBSOFS obtained the smallest value for Vowel and Satellite. For five of the rest six data sets, although other compared algorithms have the smallest NF value, their performances on the classification accuracy were degraded.

Table 3. The Acc values obtained by all the algorithms

Datasets	SFS	PTA	GAFS	BPSOFS	CBSOFS
Vowel	99.70	99.70	99.70	99.70	99.70
Wine	95.51	95.51	95.51	99.26	99.55
Vehicle	69.50	71.75	72.97	75.70	76.06
Segmentation	92.95	92.95	92.95	97.90	98.23
WDBC	94.02	94.02	93.95	97.65	98.18
Ionosphere	93.45	93.45	94.70	94.30	95.92
Satellite	90.45	91.10	91.36	91.33	91.82
Sonar	91.82	92.31	95.49	94.47	95.03

Table 4. The *NF* values obtained by all the algorithms

Datasets	SFS	PTA	GAFS	BPSOFS	CBSOFS
Vowel	8	8	8	8.00	8.00
Wine	8	8	5	7.20	8.16
Vehicle	11	11	7	10.10	8.20
Segmentation	8	8	8	10.50	11.50
WDBC	18	18	12	13.40	15.16
Ionosphere	7	7	7	10.50	15.07
Satellite	22	22	22	25.50	20.30
Sonar	48	48	24	31.20	32.16

5 Conclusion

Brain storm optimization is a relatively new nature-inspired algorithm. However, no one has yet applied this algorithm to feature selection problems. This paper first applied the original BSO algorithm to feature selection problems, and proposed an effective wrapper feature selection method, called CBSOFS. To evaluate the effectiveness of CBSOFS, we have conducted a series of experiments on 8 well-known datasets. The experiment results showed that the CBSOFS algorithm can achieve better classification accuracy than others nature-inspired algorithms such as PSO and GA. How to apply the BSO algorithm to more complex problems, such as multi-label or multi-objective feature selection problems [30], should be studied in our future work.

Acknowledgement. This work was jointly supported by National Natural Science Foundation of China (No. 61473299, 61473298, 61573361), and Jiangsu Six Talents Peaks Project of Province under Grant No. DZXX-053.

References

1. Jensen, R., Mac Parthalain, N.: Towards scalable fuzzy-rough feature selection. *Inf. Sci.* **323**, 1–15 (2015)
2. Park, C.H., Kim, S.B.: Sequential random k-nearest neighbor feature selection for high-dimensional data. *Expert Syst. Appl.* **42**(5), 2336–2342 (2015)
3. Wang, X., Yang, J., Teng, X., Xia, W., Jensen, R.: Feature selection based on rough sets and particle swarm optimization. *Pattern Recognit. Lett.* **28**(4), 459–471 (2007)
4. Liu, H., Yu, L.: Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans. Knowl. Data Eng.* **17**(4), 491–502 (2005)
5. Xue, B., Zhang, M.J., Browne, W.N., Yao, X.: A survey on evolutionary computation approaches to feature selection. *IEEE Trans. Evol. Comput.* **20**(4), 606–626 (2016)
6. Kohavi, R., John, G.: Wrappers for feature selection. *Artif. Intell.* **97**(1–2), 273–324 (1997)
7. Diao, R., Shen, Q.: Nature inspired feature selection meta-heuristics. *Artif. Intell. Rev.* **44**, 311–340 (2015)
8. Oreski, S., Oreski, G.: Genetic algorithm-based heuristic for feature selection in credit risk assessment. *Expert Syst. Appl.* **41**(4), 2052–2064 (2014)

9. Pedram, G., Jon Atli, B.: Feature selection based on hybridization of genetic algorithm and particle swarm optimization. *IEEE Geosci. Remote Sens. Lett.* **12**(2), 309–313 (2015)
10. Al-Ani, A., Alsukker, A., Khushaba, R.: Feature subset selection using differential evolution and a wheel based search strategy. *Swarm Evol. Comput.* **9**, 15–26 (2013)
11. Sina, T., Parham, M.: Relevance-redundancy feature selection based on ant colony optimization. *Pattern Recognit.* **48**(9), 2798–2811 (2015)
12. Wang, G., Chu, H.S., Zhang, Y.X.: Multiple parameter control for ant colony optimization applied to feature selection problem. *Neural Comput. Appl.* **26**(7), 1693–1708 (2015)
13. Zorarpaci, E., Ozel, S.A.: A hybrid approach of differential evolution and artificial bee colony for feature selection. *Expert Syst. Appl.* **62**, 91–103 (2016)
14. Hancer, E., Xue, B., Zhang, M.J.: Pareto front feature selection based on artificial bee colony optimization. *Inf. Sci.* **422**, 462–479 (2018)
15. Zhang, Y., Song, X.F., Gong, D.W.: A return-cost-based binary firefly algorithm for feature selection. *Inf. Sci.* **418–419**, 561–574 (2017)
16. Zhang, Y., Gong, D.W., Hu, Y.: Feature selection algorithm based on bare bones particle swarm optimization. *Neurocomputing* **148**, 150–157 (2013)
17. Xue, B., Zhang, M.J., Browne, W.N.: Particle swarm optimization for feature selection in classification: a multi-objective approach. *IEEE Trans. Cybern.* **43**(6), 1656–1671 (2013)
18. Zhang, Y., Gong, D.W., Cheng, J.: Multi-objective particle swarm optimization approach for cost-based feature selection in classification. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **22**(99), 64–75 (2017)
19. Zhang, Y., Gong, D.W., Zhang, W.Q.: Feature selection of unreliable data using an improved multi-objective PSO algorithm. *Neurocomputing* **171**, 1281–1290 (2016)
20. Shi, Y.: Brain storm optimization algorithm. In: Tan, Y., Shi, Y., Chai, Y., Wang, G. (eds.) *ICSI 2011. LNCS*, vol. 6728, pp. 303–309. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21515-5_36
21. Cheng, S., Qin, Q.D., Chen, J.F., Shi, Y.H.: Brain storm optimization algorithm: a review. *Artif. Intell. Rev.* **46**(4), 445–458 (2016)
22. Ma, X.J., Jin, Y., Dong, Q.L.: A generalized dynamic fuzzy neural network based on singular spectrum analysis optimized by brain storm optimization for short-term wind speed forecasting. *Appl. Soft Comput.* **54**, 296–312 (2017)
23. Wang, J.Z., Hou, R., Wang, C., Shen, L.: Improved v-support vector regression model based on variable selection and brain storm optimization for stock price forecasting. *Appl. Soft Comput.* **49**, 164–178 (2016)
24. Duan, H.B., Li, C.: Quantum-behaved brain storm optimization approach to solving loney’s solenoid problem. *IEEE Trans. Magn.* **51**(1), 1–7 (2015). ID: 7000307
25. Kennedy, J., Eberhart, R.C.: A discrete binary version of the particle swarm algorithm. In: *Proceedings of 1997 Conference Systems Man and Cybernetics*, pp. 4104–4108 (1997)
26. Murphy, P.M., Aha, D.W.: UCI repository of machine learning databases. Technical report, Department of Information and Computer Science, University of California, Irvine, California. <http://www.ics.uci.edu/~mllearn/MLRepository.html>
27. Pudil, P., Novovicova, J., Kittler, J.: Floating search methods in feature selection. *Pattern Recognit. Lett.* **15**(11), 1119–1125 (1994)
28. Kudo, M., Sklansky, J.: Comparison of algorithms that select features for pattern classifiers. *Pattern Recognit.* **33**(1), 25–41 (2000)
29. Oh, I.-S., Lee, J.S., Moon, B.R.: Hybrid genetic algorithms for feature selection. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(1), 1424–1437 (2004)
30. Zhang, Y., Gong, D.W., Sun, X.Y., Guo, Y.N.: A PSO-based multi-objective multilabel feature selection method in classification. *Sci. Rep.* **7**, 376 (2017)



A Hybrid Model Based on K-EPF and DPIO for UAVs Target Detection

Jinsong Chen¹, Lu Xiao², Jun Wang³, Huan Liu^{2(✉)},
and Qianying Liu²

¹ Department of Information Management,
National Yunlin University of Science and Technology, Douliu,
Yunlin 64002, Taiwan

² College of Management, Shenzhen University, Shenzhen 518060, China
592282827@qq.com

³ Network Engineering, South China Normal University,
Guangzhou 510630, China

Abstract. A hybrid model, combining the Division Pigeon-inspired Optimization (DPIO) with a novel target detection method which is based on K-means and Edge Potential Function (K-EPF), is proposed in this paper. In K-EPF, K-mean is used to segment the image into two parts, which is helpful to enhance the efficiency of shape-matching. Basic PIO algorithm is prone to falls into local optima. In DPIO algorithm, this problem is solved by the multi-population mechanism and the landmark operator based on elite list. It effectively improves the optimization performance and convergence speed of the algorithm. In order to prove the superiority of DPIO, a series of algorithm is utilized in our comparative experiments, including particle swarm optimization (PSO), and standard genetic algorithm (GA).

Keywords: Unmanned Aerial Vehicles · Pigeon-inspired optimization
Edge potential function

1 Introduction

Today, Unmanned Aerial Vehicles (UAVs) are extensively used in agriculture, industry and military because of its low prices and high maneuverability. It is more convenient and flexible to perform some dangerous tasks by Unmanned Aerial Vehicles than manned aircraft. For instance, Unmanned Aerial Vehicles (UAVs) technology can be used for Search and Rescue (SAR), and have great use in protecting people's lives and helping rescues search and rescue [1]. Moreover, UAVs are being exploited by numerous nations for defense-related missions. Therefore, UAVs technology exerts a profound influence on the development and security of nations.

To accomplish the various tasks, target detection system plays an important role in the application of UAVs. The study for UAVs in recent years, Meng et al. [8] used the template matching method to recognize and track the runway in the image sequences. Deng He Section [9] proposed a method for edge detection by improved artificial bee colony and visual attention. Niu et al. [10] used the target area in the DWT domain to

perform infrared and visible image fusion to achieve environmental recognition and detection tasks [2]. Shape representation and matching methods are the core to cope with the target recognition and detection problems.

“Edge Potential Function (EPF) is a novel approach which detects visual objects in digital images using edge maps. The method is based on the innovative concept of edge potential function (EPF) which is used to model the attraction generated by edge structures contained in an image over similar curves” [3]. EPF can extract the edge map from the digital image to calculate and compare a gravitational field to the magnetic field produced by the charged component. In the case of shape matching, the EPF can be used to display similar shapes in the image to compare user sketches: the higher the similarity of the two shapes, the higher the total amount produced by the fringe field.

Owing to the high efficiency and robustness of the EAs, there has been a growing interest in solving multicriteria optimization problems by evolutionary approaches such as Particle Swarm Optimization (PSO), Artificial Bee Colony Optimization (ABC), and Genetic Algorithm (GA). Moradi et al. [4] used a combined genetic algorithm (GA) with particle swarm optimization (PSO) in distribution systems to minimize network power losses, better voltage regulation and improve the voltage stability. Karaboga et al. [5] applied the ABC algorithm to the field of Cluster-based wireless sensor network routing.

Based on the special navigation behavior of the pigeons in the homing process, Duan et al. [6] proposed a bionic swarm intelligent algorithm—the pigeon-inspired optimization algorithm (PIO). In this algorithm, by simulating the mechanism in which the pigeons use different navigation tools at different stages to find the target, two different operator models are proposed: landmark operator, map and compass operator. However, the original PIO algorithm has a multi-peak state, but the peak gradient is very shallow except for the peak gradient near the optimal nearby position. In order to solve this problem, this paper proposes a Division PIO algorithm (DPIO) based on the PIO algorithm and improved EPF model K-means (K-EPF).

In this paper, we combined the K-EPF and DDPIO to deal with the target detection problems for UAVs. The role of EPF is to provide a matching sketch based on the selected contour. DDPIO then uses this sketch to find the optimal solution from the image. In the judging stage, the more accurate the sketch matches the image, the larger the value of EPF. The EPF will get the maximum if the sketch matches the image exactly by translating, reorienting and scaling itself.

The rest of this paper is organized as follows. Section 2 briefly describes the EPF model and the improved version of K-EPF. In Sect. 3, the novel DPIO algorithm is shown in detail. Section 4 gives the results of a series of experiments. Finally, the conclusion is presented in Sect. 5.

2 EPF Method

Edge potential functions [3] can be calculated from the edge map readily draw from an electronic image and represent a kind of attraction field compare with the field produced by a charged element. Under the situation of shape matching, the EPF can attract a user sketch in the position where a similar shape is showed in the image: in fact, the

higher the similarity of the two shapes, the edge field engenders the higher the total attraction. Equations (1)–(3) is from references [3].

In a potential field, the point where has the maximized unlike potential will attract opposite charge. Our model is analogous to the above behavior, the i th edge of the image at coordinates (x_i, y_i) is considered as the charge point $Q_{eq}(x_i, y_i)$ which forms the potential of whole image pixels:

$$EPF(x, y) = \frac{1}{4\pi\epsilon_{eq}} \sum_i \frac{Q_{eq}(x_i - y_i)}{\sqrt{(x - x_i)^2 + (y - y_i)^2}} \quad (1)$$

In the model, a test object is the generic sketch contour which is matched with the target content. Therefore, we want to see a group of equivalent charged points attract it which maximizes the edge potential.

The Windowed EPF (WEPF) defines a window beyond which the edge points are ignored can reduce the calculated complexity, and hence the robustness. So the WEPF can be shown as:

$$WEPF(x, y) = \frac{Q}{4\pi\epsilon_{eq}} \sum_{(x_i, y_i) \in w} \frac{1}{\sqrt{(x - x_i)^2 + (y - y_i)^2}} \quad (2)$$

Where w is the window chosen; Q is equal to the charge of each edge points.

When the EPF energy reaches the maximum, the optimal matching is got. For knowing whether the target image contains the object, which means our model successfully captured the sketch of a given position, rotation and scale factor, the EPF energy's matching function is defined as:

$$f(c_k) = \frac{1}{N^{(c_k)}} \sum_{n^{(c_k)}=1}^{N^{(c_k)}} \{EPF(x_n^{c_k}, y_n^{c_k})\} \quad (3)$$

Where $n^{(c_k)}$ is the n th pixel of the c th contour, and $N(c)$ is the sum of c th contour.

2.1 K-EPF Method

In order to improve the performance of target detection, a novel target detection method based on K-means and EPF (K-EPF) is presented in this paper. In this newly proposed method, edge extracted images are segmented into two parts by K-means, which narrows down search scope and reduces interference in images to make target sketch move on more accurate images. Therefore, it is beneficial to enhance the efficiency of shape-matching. Figure 1 shows the flowchart of K-EFP.

This method consists of the following parts:

- (1) **Edge extraction.** Edge is the most important feature of an image and it is the set of pixels around which the grayscale of the pixels has a step change or a roof

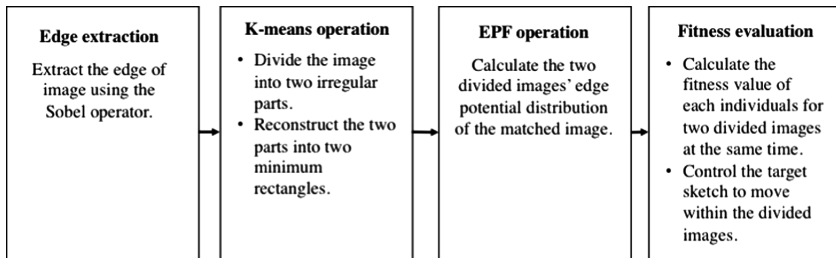


Fig. 1. The flowchart of K-EPF

change. Edge function is provided to detect and extract the edge of gray image using the Sobel operator.

- (2) **K-means operation.** As an effective and simple clustering approach, K-means is widely used in image segmentation. In this method, K-means is executed to divide the edge extracted image into two irregular parts firstly. Because the image value matrix is rectangular, if the irregular image is used directly, there will be a black field with a large range of value of 0 in the picture. As a result, the search direction will be lost and effective shape matching will not perform well. So these two parts of image need to be reconstructed into two minimum rectangles according to their own top, bottom, left and right edge points, which eliminates large black areas and makes the algorithm search normally.
- (3) **EPF operation.** After getting the two rectangle images, in order to increase the gradient of black parts in these images, they will be calculated edge potential distribution of the matched image by (2).
- (4) **Fitness evaluation.** Each individual representing a potential solution has four dimensions including pixels of moving left and right, pixels of moving up and down, rotation angle and multiples of enlargement or reduction. Target sketch will move on the divided images according to the parameters of four dimensions (move left, right, up, down; rotate; zoom in or out). After moving, use Eq. (3) to calculate the fitness value of pixels that target sketch covers on the divided images. Target detection is carried out for the two divided images respectively at the same time, and the best fitness value is selected. When the maximum iteration is reached, the individual with the best fitness value will be selected (In experiments, our purpose is to search the minimum value, so Eq. (3) is added a minus sign).
- (5) **Constraint handling.** The numerical values of the first two dimensions in each individual are pixel numbers that will be converted to integers in the process of calculation. In addition, when the target sketch moves, selects and scales according to the variables corresponding to the pigeon individual, the target sketch may cross the boundary range of the divided images (there's an invalid solution). At this point, a new pigeon individual will be created to replace the pigeon which crosses image boundaries. After calculating fitness values, integer pixel numbers and newly generated individual will be saved in relevant pigeon individuals.

3 PIO Algorithm and Its Improved Version DPIO

3.1 PIO Algorithm

Based on the special navigation behavior of the pigeons in the homing process, Duan et al. [6] proposed a bionic swarm intelligent algorithm—the pigeon-inspired optimization algorithm simulating the mechanism in which the pigeons use unlike navigation ways to search the destination, so two different operators named map and compass operator and landmark operator are proposed:

In the algorithm, the position and speed of each pigeon are initialized like $X_i = [x_{i1}, x_{i2}, \dots, x_{im}]$, $V_i = [v_{i1}, v_{i2}, \dots, v_{im}]$. Where i is the i th pigeon, and m is the dimension of the problem to be solved. In the multi-dimensional search space, position X_i and speed V_i are updated according to Eqs. (4) and (5).

$$V_i^{N_c} = V_i^{N_c-1} e^{-R \times N_c} + rand(X_{g_{best}} - X_i^{N_c-1}) \tag{4}$$

$$X_i^{N_c} = X_i^{N_c-1} + V_i^{N_c} \tag{5}$$

Where R is the map and compass factor, the value range is range from 0 to 1; $rand$ is a random number ranging from 0 to 1; N_c is the current iteration number; $X_{g_{best}}$, after $N_c - 1$ iterations, is the global optimal position obtained by comparing the positions of all the pigeons.

When the number of loops reaches the required number of iterations, the map and compass operators are stopped, and the landmark operator continues to work.

In the landmark operator, the half of pigeons will be eliminated each iteration. The position X_i of the pigeon is thus updated according to the following Eqs. (6)–(8), the fitness of each pigeon calculated by (9) and it greater than zero.

$$X_{center}^{N_c-1} = \frac{\sum_{i=1}^{N_c-1} X_i^{N_c-1} F(X_i^{N_c-1})}{N^{N_c-1} \sum_{i=1}^{N_c-1} F(X_i^{N_c-1})} \tag{6}$$

$$N^{N_c} = \frac{N^{N_c-1}}{2} \tag{7}$$

$$X_i = X_i^{N_c-1} + rand(X_{center}^{N_c-1} - X_i^{N_c-1}) \tag{8}$$

$$F(X_i^{N_c-1}) = \begin{cases} \frac{1}{fitness(X_i^{N_c-1}) + \epsilon}, & \text{for minimization problems} \\ fitness(X_i^{N_c-1}), & \text{for maximization problems} \end{cases} \tag{9}$$

Where X_{center} is the center position of the remaining pigeons and will be used as a landmark, a reference for flight.

After the above iteration loops to the maximum number we set, the landmark operator also stops working.

3.2 Improved PIO Algorithm Division-PIO

Similar to the combinatorial optimization problem of the UAV target detection, the original PIO algorithm has a multi-peak state, but the peak gradient is very shallow except for the peak gradient near the optimal nearby position. In order to solve this problem, this paper proposes a Division-PIO algorithm (DPIO) based on the PIO algorithm.

3.2.1 Division of the Multi-population Mechanism

This paper establishes a division of multi-population mechanism in the magnetic compass and solar operator and divides the pigeons into two populations: the leader birds and the follower birds. The follower birds, which make up the majority of the population, cruise in the given search direction. The group of historical optimal location (Elite list) will be updated after the follower birds explored. Based on the information of the elite list, the leader bird searches in the vicinity and uses the obtained optimal position information to guide the search from the next round of the follower birds (by participating in the comparison of the current optimal position).

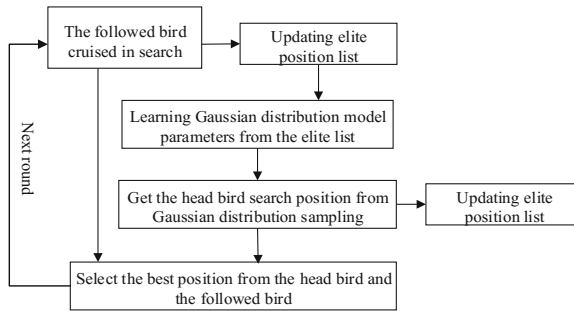


Fig. 2. The process of position update

As shown in Fig. 2, this paper proposes the formula of flight direction instead of the original formula in the process of cruise search of the follower birds, which is inspired by the DE algorithm. [7] (the current position of the pigeon group together with the flight direction will obtain the search position). Doing like this, it will focus more on the location near the optimal value when searching. In depth search, it is easier to find new peaks near the current optimal position. At the same time, the formula increases the diversity of the depth search path by double randomizing the difference between the optimal position and the current position. The double randomization includes the multiplication of the difference itself with the rand value and the size of the search step of the rand value. The formula is as follows:

$$\Phi_i = rand_1(D, 1) \cdot (rand_2(D, 1) \cdot (X_{best,G} - X_{r,G}) + rand_3(D, 1) \cdot (X_{best,G} - X_{i,G})) \quad (10)$$

Φ_i is the cruise direction of i (the number of the follower birds). rand_1 , rand_2 and rand_3 are three random vectors independently. D is the dimension of the solution. G is the times of the current iterations. $X_{best,G}$ is the optimal position in the iterations of G th in the pigeon population. $X_{r,G}$ is the position of a random pigeon in the current pigeon population. r is a random integer between $[1, \text{the number of the follower birds}]$. $X_{i,G}$ is the current location of the follower bird i .

The elite list used to save the optimal position to set parameter list size (i.e., the number of optimal position). After each new location producing, compared the values in the elite list with the current value. As shown below:

$$x_i = \text{norminv}(\text{rand}(D, 1), \mu_G, \sigma_G) \quad (11)$$

x_i is the location where the leader bird explores. $\text{norminv}()$ is the Gaussian distribution model. $\text{rand}(D, 1)$ is a vector generated randomly by the Gaussian distribution model. μ_G and σ_G are the mean vector and variance vector of the pigeon position in the current elite list (all are calculated by dimension).

As shown in the formula, the Gaussian distribution is randomly sampled according to the population of the leader bird to obtain the leader bird search position. The search position of leader bird is obtained by sampling randomly of the leader bird and the Gaussian distribution model. The position obtained in this way must be distributed near the position in the elite list. When the position of the elite list is in a large gradient, more birds can be assigned to explore the gradient through the guidance of the leader bird.

The process is to find the gradient by the follower birds, and confirm the gradient by the leader bird. After the gradient is confirmed (that is, the position of the leader bird becomes the current optimal position G_{best}), the next step will be started by the formula of the follower bird position update.

Through the above search process, the pigeons can sweep over the shallow peaks of the gradient quickly, so as to avoid missing the optimal position to a greater extent. The improved algorithm is more sensitive to the optimal and its surrounding position and is suitable for solving the combinatorial optimization problem of local optimal values except for the vicinity of the optimal value.

3.2.2 The Landmark Operator Based on the Elite List

The purpose of the landmark operator is to assume that the optimal position is near the current population when the search process is over. At this moment, the outward search is stopped and turn to explore the assumed range of locations. The key point of this step is to determine the center of the location and the number of populations that are searched. In the original PIO algorithm, the population is decremented exponentially per round, while the Division-PIO (DPIO) algorithm reduces the number of pigeons by a fixed number per round. This is to improve the utilization of landmark operators in the search process. Therefore, by reducing the number of population to reduce ineffective search.

In the magnetic compass and solar operator, the algorithm has performed a full search for the solution space. In our experiments, we found that the position in the elite list has formed a small range, and the division search of the algorithm is distributed

near the position of the elite list. (that is, the algorithm begins to converge). In the landmark operator, the mean of the position in the elite list is taken as the fixed center of the pigeons. The pigeons are distributed around the center for repeated search, so that the algorithm can get the optimal value finally. As shown below:

$$X_i = X_{i,G} + rand(D, 1) \cdot (\mu - X_{i,G}) \tag{12}$$

X_i is the search position of a pigeon i . $X_{i,G}$ is the position of a pigeon i . μ is the mean of the position in the elite list (the elite list is no longer updated in the landmark operator). As shown in Fig. 3, the process is as follows:

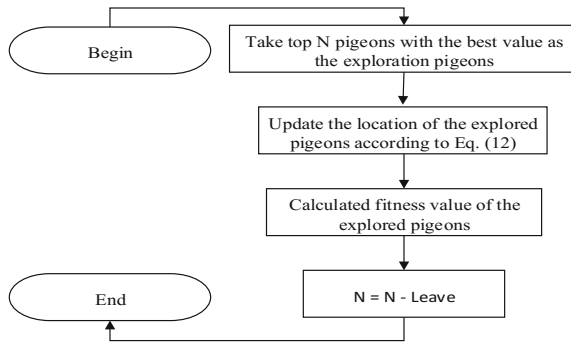


Fig. 3. The process of exploration

N is the total number of pigeons. Leave is the number of pigeons reduced per iteration. For each iteration, the pigeons in the top N are selected for position update and the fitness calculate.

4 Experiment

In order to study the feasibility and effectiveness of the hybrid model and DPIO algorithm in the application, we compare PIO, PSO, GA algorithm and select three images for the experiment.

4.1 The Target of Image Detection and Parameters Setting

In order to evaluate the performance of KEPF-DPIO in the target detection problem of the UAV, this paper selects three aerial views of different recognition targets for target detection. In the three images, the tests identify buildings, aircraft, and ships in different positions and sizes in the picture to demonstrate the applicability of the method. In order to evaluate the performance in the clutter background, Gaussian noise is added to each photo. At the same time, the combination of KEPF and EPF was used in the experiment to perform target detection, and the two recognition methods were compared horizontally.

The set of parameter for DPIO algorithm is shown in Table 1. The parameter for PIO is come from [3], $R = 0.2$. The parameter for PSO and GA are come from [2]. The detailed parameters of PSO: the learning factors of PSO is $C1 = 2, C2 = 2$. The weight of PSO is: $Wmax = 0.9$, and $Wmin = 0.4$. The detailed parameters of GA: the crossover probability of GA is 0.7 and the mutation probability of GA is 0.1. The pictures in this experiment are all from the Internet. The runtime of all experiments is 30.

Table 1. Set of the parameter for DPIO algorithm

Parameter	Description	Value
N	Number of pigeons	50
$Tmax$	Maximum times of iteration	100
D	The dimension of the solution	4
Ns	Number of cruises	5
$alma$	Number of elite sets	5
FB	Number of Follower bird	40
LB	Number of Leader bird	10
$Leave$	Number of outliers	2
$T2$	Number of iterations of the landmark operator.	12

4.2 Experimental Result and Analyses

Table 2 shows the results of DPIO, PIO, PSO, and GA in KEPF and EPF. The same case, the same algorithm, the optimal values are underlined in KEPF and EPF. As shown in Table 2, DPIO is clearly superior to PIO, and PSO and GA have the best value and the highest mean value in each test. At the same time, it can be noted that except for that it is similar to KEPF under PSO in case 1, and KEPF is better than EPF in other tests.

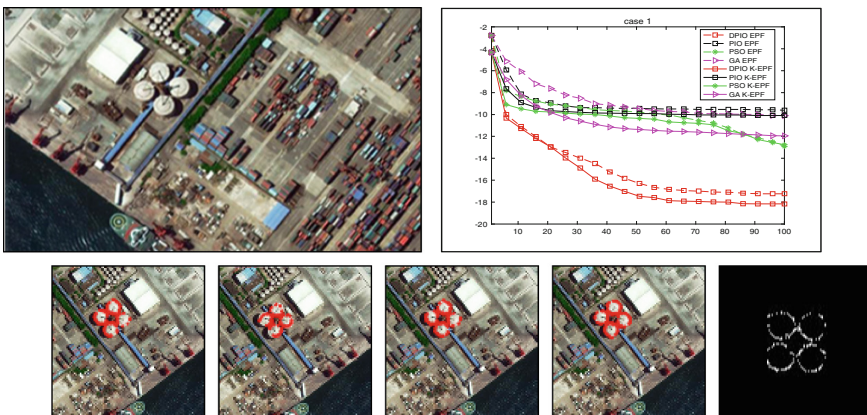


Fig. 4. Experimental result of case 1 (From left to right is DPIO, PIO, PSO, GA)

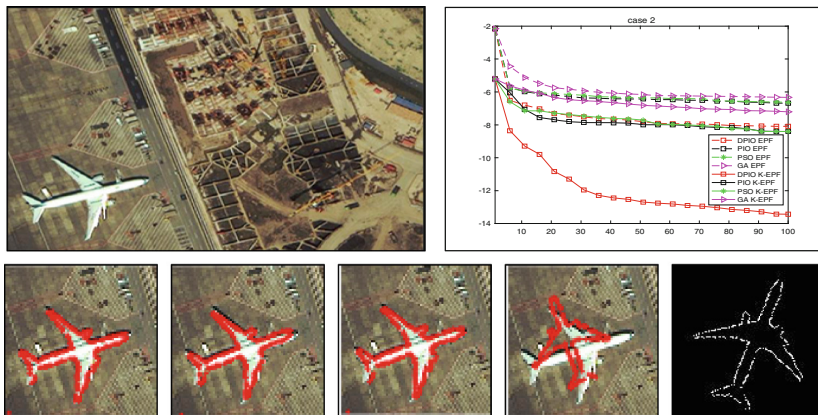


Fig. 5. Experimental result of case 2 (From left to right is DPIO, PIO, PSO, GA)

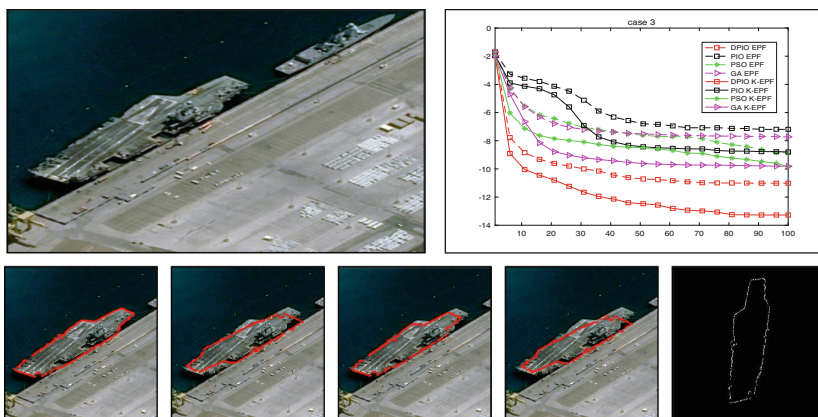


Fig. 6. Experimental result of case 3 (From left to right is DPIO, PIO, PSO, GA)

Figures 4, 5 and 6 shows the template maps, convergence curves, test originals and test results of KEPF and EPF in different test pictures. The resulting graph of the test is the divided graph in KEPF. The same algorithm is used in EPF and KEPF, we use the same color and different line types to show in the convergence graph (EPF with a dashed line, KEPF with solid line).

The starting point of the same initial particle is used in the test, but since KEPF uses the segmented graph, the fitness has two starting points in each graph.

In the division of multi-population mechanism, the algorithm can explore the position near to each current optimal position quickly, and jump out of the local optimal. It is difficult to detach when entering the vicinity of the optimal value with a deeper gradient so that the position of the peak can be further approached. As shown in Figs. 4, 5 and 6, DPIO has the fastest convergence speed and the optimal results are more stable than the other algorithm. From the target sketch, you can see that DPIO has

Table 2. The results of cases

Functions	Results	DPIO	PIO	PSO	GA
Case1 EPF	Mean	-1.724E+01	-9.613E+00	-1.292E+01	-1.010E+01
	Best fitness	-1.920E+01	-1.114E+01	-1.869E+01	-1.360E+01
	Best parameter	(45, 196, 310.36, 9.96)	(1, 201, 2.72, 8.37)	(44, 195, 310.31, 10.09)	(48, 196, 222.01, 9.69)
Case1 K-EPF	Mean	-1.816E+01	-1.009E+01	-1.276E+01	-1.195E+01
	Best fitness	-1.996E+01	-1.081E+01	-1.725E+01	-1.621E+01
	Best parameter	(45, 57, 310.27, 9.95)	(49, 58, 44.65, 8.82)	(46, 58, 311.45, 9.76)	(47, 57, 130.24, 10.03)
Case2 EPF	Mean	-8.093E+00	-6.684E+00	-6.639E+00	-6.328E+00
	Best fitness	-1.385E+01	-8.478E+00	-7.346E+00	-7.174E+00
	Best parameter	(164, 5, 309.68, 9.39)	(168, 11, 306.02, 8.36)	(20, 153, 237.01, 8.39)	(25, 109, 87.33, 9.23)
Case2 K-EPF	Mean	-1.343E+01	-8.399E+00	-8.398E+00	-7.204E+00
	Best fitness	-1.584E+01	-1.089E+01	-1.118E+01	-8.754E+00
	Best parameter	(42, 3, 309.97, 9.68)	(43, 8, 313.60, 9.34)	(41, 6, 312.35, 9.70)	(29, 23, 42.14, 8.26)
Case3 EPF	Mean	-1.101E+01	-7.191E+00	-8.878E+00	-7.721E+00
	Best fitness	-1.667E+01	-9.003E+00	-1.010E+01	-1.032E+01
	Best parameter	(98, 18, 319.94, 9.99)	(103, 34, 318.66, 8.62)	(117, 29, 139.67, 8.56)	(102, 22, 318.91, 9.86)
Case3 K-EPF	Mean	-1.327E+01	-8.795E+00	-9.843E+00	-9.804E+00
	Best fitness	-1.878E+01	-1.076E+01	-1.105E+01	-1.229E+01
	Best parameter	(98, 18, 319.99, 9.99)	(114, 24, 134.79, 9.24)	(108, 20, 320.66, 9.26)	(120, 28, 135.23, 8.74)

better results of target detection in each case. The above results show that our proposed hybrid algorithm of DPIO algorithm combined with the improved K-EPF model is effective and robust in solving the problem of UAV target detection.

5 Conclusions

In this paper, we propose a binding model based on the DDPIO and the K-EPF to accomplish the target detection task for UAVs. The comparison with GA, PSO, ABC, basic PIO and DDPIO shows that the DDPIO algorithm significantly improves the

performance of basic PIO, and has effectiveness and stability. Meanwhile, the statistics from our experiments also show that DDPIO with K-EPF has better performance than with EPF. Besides, compared with the basic PIO which only uses selected goals, the combination of the EPF and the DDPIO makes the application of the algorithm more flexible. Our algorithm is more suitable than other algorithms to complete more complex target detection tasks for UAVs.

Acknowledgment. This work is partially supported by The National Natural Science Foundation of China (Grants Nos. 71571120, 71271140, 61472257). Lu Xiao, and Jinsong Chen contributed equally to this work and shared the first authorship.

References

1. Lygouras, E., Gasteratos, A., Tarchanidis, K., Mitropoulos, A.: ROLFER: a fully autonomous aerial rescue support system. *Microprocess. Microsyst.* **61**, 32–42 (2018)
2. Li, C., Duan, H.: Target detection approach for UAVs via improved pigeon-inspired optimization and edge potential function. *Aerosp. Sci. Technol.* **39**, 352–360 (2014)
3. Dao, M.S., De, N., Massa, A.: Edge potential functions (EPF) and genetic algorithms (GA) for edge-based matching of visual objects. *IEEE Trans. Multimed.* **9**(1), 120–135 (2007)
4. Moradi, M.H., Abedini, M.: A combination of genetic algorithm and particle swarm optimization for optimal DG location and sizing in distribution systems. *Int. J. Electr. Power Energy Syst.* **34**(1), 66–74 (2012)
5. Karaboga, D., Okdem, S., Ozturk, C.: Cluster based wireless sensor network routing using artificial bee colony algorithm. *Wirel. Netw.* **18**(7), 847–860 (2012)
6. Duan, H.B., Qiao, P.X.: A new swarm intelligence optimizer for air robot path planning. *Int. J. Intell. Comput. Cybern.* **7**(1), 24–37 (2014)
7. Mallipeddi, R., Suganthan, P.N., Pan, Q.K., Tasgetiren, M.F.: Differential evolution algorithm with ensemble of parameters and mutation strategies. *Appl. Soft Comput.* **11**(2), 1679–1696 (2011)
8. Meng, D., Cao, Y., Guo, L.: A method to recognize and track runway in the image sequences based on template matching. In: *1st International Symposium on Systems and Control in Aerospace and Astronautics*, pp. 1218–1221. IEEE Press, New York (2006)
9. Deng, Y.M., Duan, H.B.: Biological edge detection for UCAV via improved artificial bee colony and visual attention. *Aircr. Eng. Aerosp. Technol.* **86**(2), 138–146 (2014)
10. Niu, Y., Xu, S., Wu, L., Hu, W.: Airborne infrared and visible image fusion for target perception based on target region segmentation and discrete wavelet transform. *Math. Prob. Eng.* (2012). <https://doi.org/10.1155/2012/275138>



A Hybrid Data Clustering Approach Based on Hydrologic Cycle Optimization and K-means

Ben Niu, Huan Liu, Lei Liu, and Hong Wang^(✉)

College of Management, Shenzhen University, Shenzhen 518060, China
ms.hongwang@gmail.com

Abstract. K-means is a popular and simple clustering method by grouping data into predefined K clusters efficiently. However, K-means performs poorly in the presence of poor centers and tends to converge prematurely. Hydrologic Cycle Optimization, as a novel algorithm inspired by the natural phenomena, has a good ability to search for the global optimal solutions. To overcome drawbacks associated with the K-means and find better initial centroids, in this study, a hybrid clustering algorithm based on Hydrologic Cycle Optimization and K-means (abbreviated as HCO+K-means) is proposed. The proposed algorithm includes two modules: HCO module and K-means module. It executes HCO module firstly to find the best individual with optimal fitness value. While the position of the best individual is then considered as initial set of centers for K-means module to search for a higher quality clustering solution. For comparison purpose, the K-means, PSO+K-means, WCA+K-means and HCO+K-means algorithms are chosen to evaluate on six different datasets. The experimental results indicate that the proposed HCO+K-means algorithm has a strong global search ability and obtains better clustering results in comparison to the other clustering methods.

Keywords: Hydrologic Cycle Optimization · K-means
Hybrid clustering method

1 Introduction

Clustering, an unsupervised classification technique, is widely used in many fields, like data mining [1, 2], image segmentation [3] and pattern recognition [4]. There are many clustering methods, out of which K-means is the most popular approach due to its computational efficiency and simplicity. However, K-means is sensitive to the initial selection of centroids and suffers from premature convergences [5].

To help K-means solve its drawbacks and find a better clustering solution, hybrid clustering approaches based on heuristic algorithms and K-means are proposed. Cao et al. [6] proposed a hybrid clustering approach called GAKREM, combining K-means and EM algorithms with genetic algorithms to solve the problems of the K-means and EM algorithms. Laszlo et al. [7] introduced a genetic algorithm to find good centroids for K-means by a new crossover operator. Li et al. [8] proposed an image segmentation algorithm by combining dynamic particle swarm optimization with K-means, in order

to enhance the K-means' global search ability. Niknam et al. [9] proposed a clustering method based on PSO, ACO and K-means to search better clustering solution. Kwedlo [10] presented a clustering algorithm based on DE and K-means and found that this algorithm can get lower SSE when K is large enough.

Recently, Hydrologic Cycle Optimization (HCO), inspired by the nature phenomenon of water cycle, has raised attention of scholars due to its high efficiency in optimization problems. HCO was proposed by Yan and Niu [11, 12], describing the processes including flow, infiltration, evaporation and precipitation in detail. In HCO, flow step helps an individual learn from another better individual, which is useful to improve the searching ability of individuals. Infiltration step makes individuals search in the neighborhood and is beneficial to increase the diversity of population. In addition, evaporation and precipitation steps will update individuals' position to avoid HCO falling into local optimal.

In this paper, we try to combine HCO with K-means and propose HCO+K-means to explore the performance of this hybrid method in finding clustering solutions. This hybrid clustering approach consists of HCO module and K-means module. HCO module provides the best individual's position to K-means module to execute clustering steps. A general fitness function "sum of squared error (SSE)" is used to evaluate the performance of clustering, which is not only to calculate the fitness values in HCO module but also in K-means module. The goal is to minimize the fitness value to find a better clustering solution through experiments.

The rest of the paper is organized as follows: Sects. 2 and 3 introduce K-means Algorithm and Hydrologic Cycle Optimization respectively. Section 4 presents the HCO+K-means in details. In Sect. 5, the experiment and results are discussed. Finally, conclusions of the work are presented in Sect. 6.

2 K-means Algorithm

K-means algorithm partitions a set of data vectors into K clusters. Euclidean formula is usually used to calculate the distance between data points and cluster centers [13].

K-means can be described as:

- (i) Choose initial K centroids randomly, $M = (M_1, M_2, \dots, M_j, \dots, M_K)$.
- (ii) Assign each data vector to the cluster $C_j (j = 1, \dots, K)$ with closest centroid vector. The distance of each data vector to each centroid is calculated by the equation:

$$d(X_p, M_j) = \sqrt{\sum_{n=1}^{N_d} (X_{pn} - M_{jn})^2} \quad (1)$$

where X_p denotes the p -th data vector; M_j denotes the j -th centroid; N_d is the input dimension and n subscripts the dimension.

- (iii) Recalculate the clusters centroids, according:

$$M_j = \frac{1}{n_j} \sum_{X_p \in C_j} X_p \quad (2)$$

where n_j is the number of data vectors belonging to cluster C_j .

- (iv) Repeat (ii) and (iii) until maximum number of iterations or any one of the stopping criterion is satisfied.

3 Hydrologic Cycle Optimization

Hydrologic Cycle Optimization (HCO) is a novel heuristic algorithm, which simulates the processes of water cycle including flow, infiltration, evaporation and precipitation.

The main principle of the HCO [11, 12] is summarized as follows:

- (i) **Initialization.** HCO creates initial individuals and then calculates the fitness value of all individuals according the fitness function. The best individual which has best fitness value will be regarded as the best individual firstly.
- (ii) **Flow.** In this operation, each individual will try to flow to another individual with better fitness value. If the new position is better than the original one, the new position will be saved and replace the original one, otherwise it will be given up and stay the old position. For the best individual, it will flow to an individual selected randomly. All individuals will execute flow operation until the new position becomes worse or the maximum flow times is satisfied. The potential position can be calculated by

$$X_{try} = X_i + (X_j - X_i) \cdot * \text{rand}(1, \text{Dimension}) \quad (3)$$

- (iii) **Infiltration.** In this part, each individual will begin its searching in neighborhood and update the position, using

$$X_{i,SD} = X_{i,SD} + (X_{i,SD} - X_{j,SD}) \cdot * 2 * (\text{rand}(1, sd) - 0.5) \quad (4)$$

- (iv) **Evaporation and precipitation.** When evaporation condition is satisfied ($\text{rand} < P_{eva}$), each individual will precipitate to another position randomly or to a neighbor position which is created by Gauss mutation. The probability of two ways happening is the same.

4 HCO+K-means Algorithm

In the HCO+K-means Algorithm proposed in this paper, each individual in HCO represents a set of centers for clustering. And the best individual with the best fitness value will be regarded as the best clustering solution.

This hybrid algorithm can be described as two modules, HCO module and K-means module. In the process of the hybrid algorithm, HCO runs a few times independently to find the best individual's position as the initial clustering centroids for K-means. Then, K-means will be executed to find the final optimal centroids.

4.1 The HCO Module

In HCO module, each individual can be constructed as below:

$$P_i = (M_{i1}, M_{i2}, \dots, M_{ij}, \dots, M_{iK})$$

where K denotes the number of clusters; M_{ij} denotes the j -th cluster centroid vector of the i -th individual in cluster C_{ij} .

The fitness function is used to calculate the fitness value of each individual to data vectors, which can be described as:

$$SSE = \sum_{j=1}^K \sum_{\forall X_p \in C_{ij}} d(X_p, M_{ij})^2 \tag{5}$$

where d is defined in Eq. (1); n_{ij} is the number of data vectors in cluster C_{ij} . The pseudo code of HCO module is given in Table 1.

Table 1. The pseudo code of HCO module

```

Read input data and Initialize parameters ( $i_j=50; i_s=50; pop\_size=50; maxFT=3; P_{eva} = 0.1$ )
For  $i=1: i_j$  :
    For  $j=1: pop\_size$  :
        Do flow step by (3) and calculate the potential individual's fitness value by (5);
        IF  $Fitness\_new < Fitness\_original \ \&\& \ flowtimes < maxFT$ 
            New position and its fitness value will be saved;
        End
    End
    For  $j=1: pop\_size$  :
        Do infiltration step by Eq.(4);           % All dimensions will be changed
        Update position and fitness value;
    End
    For  $j=1: pop\_size$  :
        IF  $rand < P_{eva}$ 
            Do evaporation and precipitation steps;
        End
    End
End
Output the best individual and its position

```

4.2 The K-means Module

After inheriting the initial clustering centroids from HCO module, K-means module will start its process to search for the final solution (Table 2 and Fig. 1).

Table 2. The pseudo code of K-means module

<p>Consider the best individual's position from HCO module as initial cluster centroid vectors</p> <p>For $i = i_j + 1; i_j + i_2$</p> <p style="padding-left: 20px;">Assign each data vector to the cluster with closest centroid vector;</p> <p style="padding-left: 20px;">Calculate the fitness value for cluster centroid vectors to data vectors by (5);</p> <p style="padding-left: 20px;">Recalculate the clusters centroids, using (2);</p> <p>End</p> <p>Output the final result</p>	
---	--

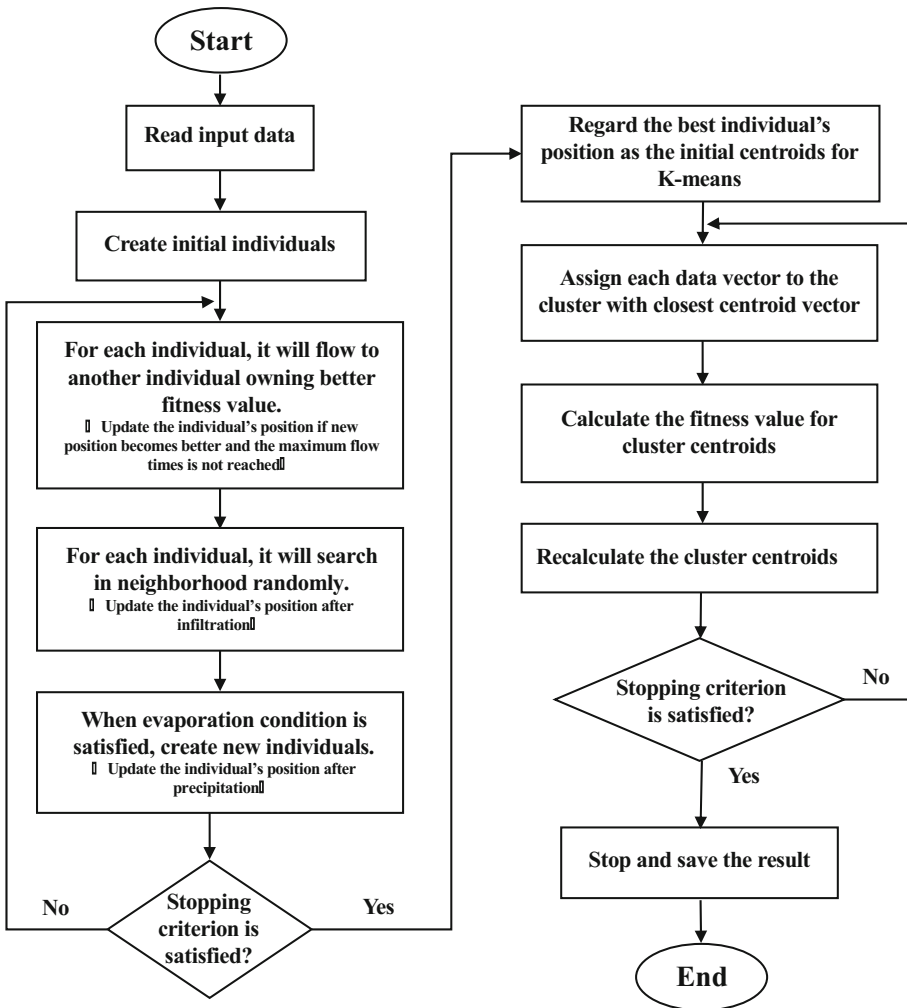


Fig. 1. Flowchart of HCO+K-means

5 Experiments and Results

5.1 Datasets and Experiment Settings

In order to test the performance of the proposed algorithm, six datasets from UCI are selected. Table 3 shows the descriptions of these six datasets. Before the experiments, in order to reduce the influence of abnormal points on the experimental results in real data, Seeds, Glass, Wine and Breast cancer datasets are preprocessed with minimum and maximum normalization. K-means, Particle Swarm Optimization (PSO) [14]+K-means, Water Cycle Algorithm (WCA) [15]+K-means and HCO+K-means algorithms are applied for clustering on these chosen datasets. SSE and accuracy are chosen as indexes to evaluate the performance of clustering in experiments.

In the experiments, the K-means can converge quickly and find a stable solution within 50 iterations while PSO, WCA and HCO need to repeat more iterations to find stable solutions. To compare them easily, the K-means runs 100 iterations. The PSO/WCA/HCO+K-means algorithm also runs 100 iterations, which executes 50 iterations of the PSO/WCA/HCO module and 50 iterations of the K-means module. For the hybrid clustering methods, the purpose of running PSO/WCA/HCO module is to find the best individual's position as clustering centroids for their own K-means module. In the PSO/WCA/HCO module, the individual with global best fitness value provides K-means with first sets of clustering centroids.

The parameter settings of the algorithms are as follows:

- *PSO+K-means algorithm*: PSO module uses 50 particles, $w = 0.7-0.9$, and $C1 = C2 = 2$, $V_{min} = 0.1 \times Lb$, $V_{max} = 0.1 \times Ub$.
- *WCA+K-means algorithm*: WCA module uses 50 individuals, $N_{sr} = 4$, $d_{max} = 1e-16$.
- *HCO+K-means algorithm*: HCO module uses 50 individuals, the maximum flow times $maxFT = 3$, evaporation probability $P_{eva} = 0.1$.

Table 3. The chosen six datasets

Name	Type	Number	Dimension	Class
Artificial 1	Artificial	400	2	2 (200, 200)
Artificial 2	Artificial	2000	2	5 (431, 67, 891, 280, 331)
Seeds	Real	210	7	3 (70, 70, 70)
Glass	Real	214	9	6 (70, 76, 17, 13, 9, 29)
Wine	Real	178	13	3 (59, 71, 48)
Breast cancer	Real	194	34	2 (148, 46)

5.2 Results and Analyses

To test the performance of the proposed algorithm, 30 times experiments were executed on each dataset. Table 4 shows the numerical results in terms of the mean value and standard deviation of SSE and accuracy (%), respectively. The best results are highlighted in bold. Figure 2 shows the convergence characteristics on six datasets.

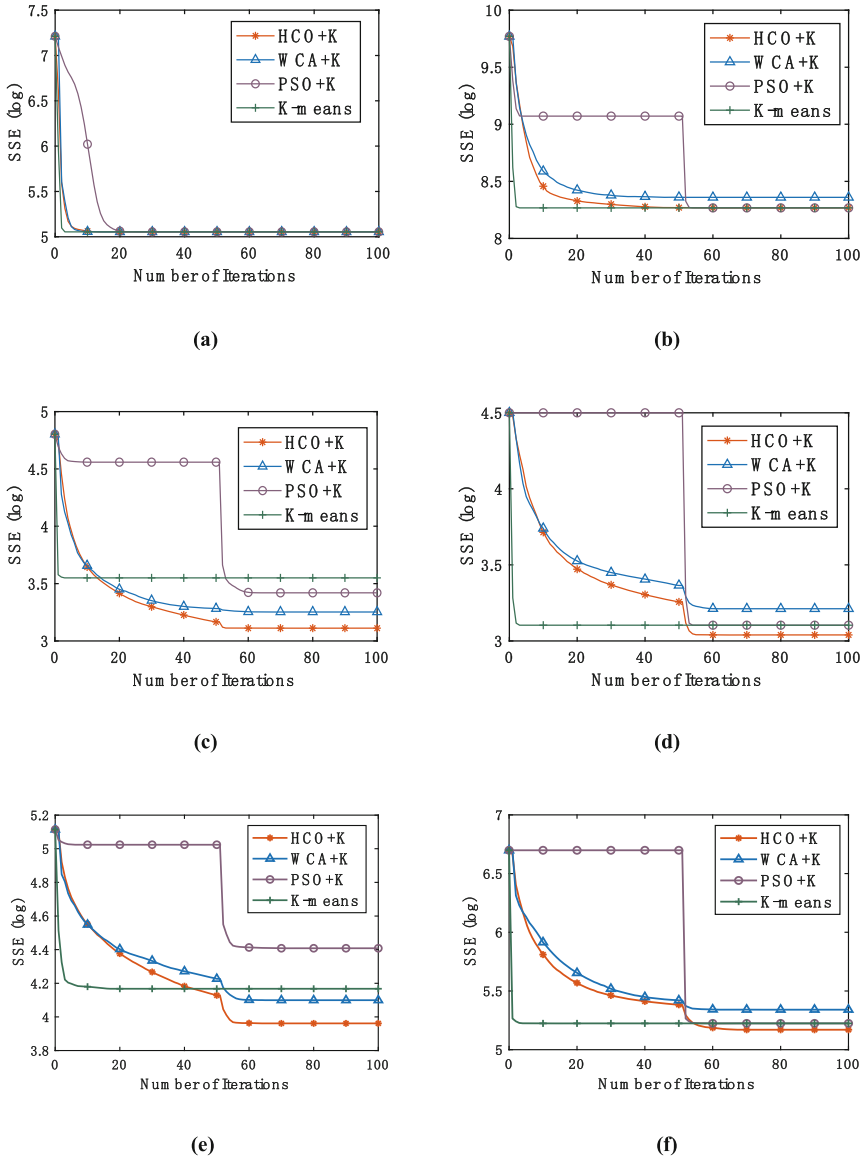


Fig. 2. Convergence characteristics on six datasets (a) Artificial 1, (b) Artificial 2, (c) Seeds, (d) Glass, (e) Wine, (f) Breast Cancer (HCO+K: HCO+K-means algorithm, WCA+K: WCA+K-means algorithm, PSO+K: PSO+K-means algorithm)

For Artificial 1 and Artificial 2 datasets, four algorithms all perform well in both SSE and accuracy. It is because these two datasets are simple with low dimension. For Seeds and Glass datasets, compared with the other three algorithms, HCO+K-means obtains the best value not only in the mean value of SSE but in the mean value of

accuracy, although it does not obtain the minimum standard deviation. In Glass dataset, all algorithms perform poorly in accuracy, probably because K-means is not good at dealing with this dataset. For Wine dataset, HCO+K-means performs significantly better than comparison algorithms in SSE and accuracy. As for Breast cancer dataset, even HCO+K-means doesn't get the best accuracy, it acquires the lowest SSE value.

As shown in Fig. 2, the K-means exhibits a faster, but premature convergence to larger fitness value, while the hybrid clustering algorithms converge slower but get smaller fitness value. Compared with the K-means, PSO+K-means and WCA+K-means algorithms, HCO+K-means obtains lower SSE value in the first 50 iterations, which means HCO can provide K-means with better centers on the same dataset. That's because HCO has a strong global search ability in solving SSE and it can escape from local optimal SSE value.

Table 4. Numerical results of six datasets

Algorithm	K-means	PSO+K-means	WCA+K-means	HCO+K-means
Artificial 1	156.5907	156.5907	156.5907	156.5907
	2.8908e-14	2.8908e-14	4.3522e-14	3.5404e-14
	99.7500	99.7500	99.7500	99.7500
	0	0	0	0
Artificial 2	3.8952e+03	3.8952e+03	4.2710e+03	3.8952e+03
	2.3126e-12	2.3126e-12	779.5148	2.3463e-12
	99.5000	99.5000	95.1150	99.5000
	0	0	9.8505	0
Seeds	34.8162	30.5529	25.8621	22.4518
	7.2269e-15	6.1322	5.9605	2.3347
	65.7143	73.3333	81.8571	88.0159
	2.8908e-14	10.9592	10.7505	4.2187
Glass	22.2800	22.2800	24.8222	20.8935
	3.6134e-15	3.6134e-15	2.5159	2.0080
	49.0654	49.0654	49.2368	49.9221
	0	0	1.4441	0.6951
Wine	64.5377	82.1399	60.3027	52.5335
	1.4454e-14	15.6548	11.9690	6.5706
	60.1124	48.6517	72.5094	87.3034
	4.3361e-14	10.1934	19.1449	15.0981
Breast cancer	185.6692	185.6692	208.6724	175.8685
	1.4454e-13	1.4454e-13	17.0877	3.0018
	53.0928	53.0928	72.3711	56.3058
	1.4454e-14	1.4454e-14	7.9690	0.7739

6 Conclusions and Further Work

Motivated by the observation that K-means method highly depends on the initial centers and has a tendency to converge prematurely, a hybrid HCO and K-means algorithm (HCO+K-means algorithm) is proposed in this paper. The performance of the proposed algorithm is demonstrated by making comparison with algorithms of K-means, PSO+K-means and WCA+K-means algorithms on six datasets. The results show that the proposed HCO+K-means algorithm can protect convergence from trapping in local optimal values and get better clustering results.

In our future study, the proposed algorithm can be improved to enhance the performance of clustering in high-dimensional data and solve some practical problems, like image segmentation, market segmentation, recommender system and so on.

Acknowledgment. This work is partially supported by The National Natural Science Foundation of China (Grants Nos. 71571120, 71271140, 71771154, 61603310, 71471158, 71001072, 61472257), and Research Foundation of Shenzhen University (85303/00000155).

References

1. Abraham, A., Das, S., Roy, S.: Swarm intelligence algorithms for data clustering. In: Maimon, O., Rokach, L. (eds.) *Soft Computing for Knowledge Discovery and Data Mining*, pp. 279–313. Springer, Boston (2008). https://doi.org/10.1007/978-0-387-69935-6_12
2. Ci, S., Guizani, M., Sharif, H.: Adaptive clustering in wireless sensor networks by mining sensor energy data. *Comput. Commun.* **30**(14), 2968–2975 (2007)
3. Portela, N.M., Cavalcanti, G.D.C., Ren, T.I.: Semi-supervised clustering for MR brain image segmentation. *Expert Syst. Appl.* **41**(4), 1492–1497 (2014)
4. Kuo, R.J., Wang, M.J., Huang, T.W.: An application of particle swarm optimization algorithm to clustering analysis. *Soft. Comput.* **15**(3), 533–542 (2011)
5. Pollard, D.: A central limit theorem for k-means clustering. *Ann. Probab.* **10**(4), 919–926 (1982)
6. Cao, D.N., Cios, K.J.: GAKREM: a novel hybrid clustering algorithm. *Inf. Sci.* **178**(22), 4205–4227 (2008)
7. Laszlo, M., Mukherjee, S.: A genetic algorithm that exchanges neighboring centers for k-means clustering. *Pattern Recogn. Lett.* **28**(16), 2359–2366 (2007)
8. Li, H., He, H., Wen, Y.: Dynamic particle swarm optimization and k-means clustering algorithm for image segmentation. *Opt.-Int. J. Light. Electron Opt.* **126**(24), 4817–4822 (2015)
9. Niknam, T., Amiri, B.: An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. *Appl. Soft Comput.* **10**(1), 183–197 (2010)
10. Kwedlo, W.: A clustering method combining differential evolution with the k-means algorithm. *Pattern Recogn. Lett.* **32**(12), 1613–1621 (2011)
11. Yan, X., Niu, B.: Hydrologic cycle optimization part I: background and theory. In: Tan, Y., Shi, Y., Tang, Q. (eds.) *ICSI 2018. LNCS*, vol. 10941, pp. 341–349. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93815-8_33
12. Niu, B., Liu, H., Yan, X.: Hydrologic cycle optimization part II: experiments and real-world application. In: Tan, Y., Shi, Y., Tang, Q. (eds.) *ICSI 2018. LNCS*, vol. 10941, pp. 350–358. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93815-8_34

13. Jain, Anil K.: Data clustering: 50 years beyond K-means. In: Daelemans, W., Goethals, B., Morik, K. (eds.) ECML PKDD 2008. LNCS (LNAI), vol. 5211, pp. 3–4. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-87479-9_3
14. Kennedy, J.: Particle swarm optimization. In: Sammut, C., Webb, G.I. (eds.) Encyclopedia of Machine Learning, pp. 760–766. Springer, Boston (2010)
15. Eskandar, H., Sadollah, A., Bahreininejad, A., Hamdi, M.: Water cycle algorithm—a novel metaheuristic optimization method for solving constrained engineering optimization problems. *Comput. Struct.* **110–111**(10), 151–166 (2012)



A Decomposition Based Multiobjective Evolutionary Algorithm for Dynamic Overlapping Community Detection

Xing Wan^{1,2}, Xingquan Zuo^{1,2(✉)}, and Feng Song^{1,2}

¹ School of Computer Science,
Beijing University of Posts and Telecommunications, Beijing, China
zuoxq@bupt.edu.cn

² Key Laboratory of Trustworthy Distributed Computing and Service,
Ministry of Education, Beijing, China

Abstract. Dynamic and overlapping are common features of community structures of many real world complex networks. There are few studies on detecting dynamic overlapping communities, but those studies consider only single optimization objective. In practice, it is necessary to evaluate the community detection by multiple metrics to reflect different aspects of a community structure. In this paper, we propose a multi-objective evolutionary algorithm based approach for the problem of dynamic overlapping community detection, with consideration of three optimization objectives: partition density (D), the extended modularity (EQ), and the community smoothing (NMI_{LFK}). The dynamic overlapping network is regarded as a set of network snapshots. The multi-objective evolutionary algorithm based on decomposition (MOEA/D) is used to detect the communities for each snapshot. Experiments show that our approach can find uniformly distributed Pareto solutions for the problem and outperforms those comparative approaches.

Keywords: Multi-objective evolutionary · Community detection
Overlapping · Dynamic

1 Introduction

In scientific research and industries, many systems can be modeled as complex networks, such as social networks, transportation networks and biological networks [1]. In a complex network, individuals are denoted by nodes and the interrelation among individuals is represented by edges. Generally speaking, a complex network has a specific community structure which reflects the network's potential laws and characteristics. Community structure detection is to discover a set of subnetworks (communities) in a complex network to make intra-community similarity higher than inter-community similarity [2]. Community structure detection is very significant and has many practical applications, such as personalized recommendation, search engine and gene identification [3–5].

Dynamical overlapping community detection is very significant since most of real-world complex networks have an overlapping community structure and meanwhile

their community structures dynamically change over time. Dynamical overlapping community detection typically involves multiple objectives, such as modularity, community score and partition density. Currently, studies [6, 7] on dynamical overlapping community detection only consider a single objective. It is well known that evolutionary algorithms (EAs) are very suitable for multi-objective optimization problems [8] since they can generate a set of Pareto solutions in one run. Although there are researches on using EAs to solve community detection problems [9], there exist no studies on using multi-objective EAs for dynamic overlapping community detection problem (DOCDP).

In this paper, we propose a MOEA/D based dynamic overlapping community detection approach (MDOA). The extended modularity EQ [10] and partition density D [11] are chosen as the accuracy objectives, and the improved mutual information NMI_{LFK} [12] as the objective to evaluate the temporal smoothness of two adjacent network snapshots. Main contributions of the paper include: (1) propose a multi-objective EA based approach for dynamic overlapping community detection; (2) MOEA/D is first introduced to solve the DOCDP; and (3) the dynamic optimization technique and a sampling update strategy are introduced into the approach to reduce the computational complexity.

2 Related Work

It is well known that communities in many complex networks are not independent of one another. Ahn *et al.* [11] proposed an overlapping community detection algorithm by partitioning links instead of nodes. If multiple edges containing the same node are assigned to multiple communities, the node is called overlapping one. Lancichinetti *et al.* [13] used local expansion and optimization to generate natural overlapping communities. In [14], Zhan *et al.* designed an encoding scheme with two segments and used an evolutionary algorithm to detection overlapping communities. All above algorithms consider the community discovery problem as a single-objective optimization problem.

For dynamic community detection problems (DCDP), their community structures evolve over time. Ma and Dong [15] proposed a framework based on evolutionary nonnegative matrix factorization to detect dynamic communities. Zhou *et al.* [16] proposed a multi-objective bio-geography based optimization algorithm with decomposition for DCDP. Gong *et al.* [9] proposed a multi-objective immune algorithm for DCDP. The algorithm optimized the modularity and normalized mutual information simultaneously to balance the quality of the community partitions and time cost.

Cazbet *et al.* [6] proposed an intrinsic Longitudinal Community Detection approach (iLCD), which used two parameters to control the increase of edges and the merging of similar communities. It could detect both static and dynamic communities. Chen *et al.* [17] proposed a centrality-based local-first and sequence smoothing mechanism approach for overlapping community detection in dynamic networks. Xu *et al.* [18] proposed a local fitness-based approach for dynamic overlapping community detection. It considered the impact of both adding and deleting nodes and edges. Aston *et al.* [7] developed a Speaker-listener approach based on the label propagation algorithm to

detect overlapping community in dynamic networks (SLPAD). DOCDP usually involve multiple objectives but there exist no studies on using a multi-objective optimization algorithm for such problems.

3 Problem Description

Typically, overlapping temporal networks can be defined as $G = \langle G_1, G_2, \dots, G_T \rangle$, where T represents the number of network snapshots in dynamic network G . $G_t = (V_t, E_t)$ represents the network snapshot at time t , where V_t and E_t represent the set of nodes and edges of the network at time t . $|V_t| = n_t$ indicates that the network G_t contains n_t nodes, and $|E_t| = m_t$ means that G_t has m_t edges connecting those nodes. The target of community detection is to partition nodes in G_t into communities (also called clusters) and generate a set $C_t = (c_{t1}, c_{t2}, \dots, c_{tk})$ with k clusters, where $\exists c_{ii}, c_{ij} \in C_t, c_{ii} \cap c_{ij} \neq \emptyset$ and at least one node belongs to more than one cluster.

4 Proposed Method

As a popular multi-objective EA, MOEA/D [19] has proved to be every effective for complex combinatorial optimization problems.

In this paper, we propose a MOEA/D based approach for the detection of overlapping communities in temporal networks. MOEA/D is used to find the Pareto community structures for each network snapshot $G_t, i = 1, 2, \dots, T$, to minimize the three objectives of EQ , D and NMI_{LFK} . To improve the computational efficiency, the dynamic optimization technique and a sampling update strategy are introduced into the approach. Objective functions, the encoding and genetic operators are described below, followed by the framework of the approach.

4.1 Objective Functions

In this paper, we choose three metrics of the extended modularity (EQ), partition density (D) and mutual information (NMI_{LFK}) as the optimization objectives.

Modularity is widely used to measure the community structure strength. In this paper we adopt extended modularity EQ [10] to evaluate the overlapping community.

$$EQ = \frac{1}{2L} \sum_{c=1}^C \sum_{v,w \in C_c} \frac{1}{O_v O_w} \left[A_{vw} - \frac{k_v k_w}{2L} \right] \quad (1)$$

where L represents the number of edges in the network. C is the number of the communities. O_v and O_w represent the number of communities that nodes v and w belong to, respectively. A_{vw} represents whether there is a link between nodes v and w . If there is a

link, $A_{vw} = 1$; otherwise $A_{vw} = 0$. k_v and k_w represent the number of edges (degrees) associated with nodes v and w , respectively.

Partition density D [11] evaluates the quality of the community partition by

$$D = \frac{2}{L} \sum_{c=1}^C L_c \frac{L_c - (n_c - 1)}{(n_c - 2)(n_c - 1)} \quad (2)$$

where L_c is the number of edges in the subnetwork cluster c . n_c is the number of nodes incident to links in cluster c .

For dynamic community detection, the network structures of adjacent time snapshots are expected to be similar. The improved NMI [12] is used to evaluate the degree of similarity between community structures of two adjacent network snapshots.

$$NMI_{LFK} = 1 - \frac{1}{2} \left(\frac{H(X/Y)}{H(X)} + \frac{H(Y/X)}{H(Y)} \right) \quad (3)$$

where X and Y represent the community partition results of two adjacent time snapshots. $H(X)$ and $H(Y)$ are the information entropy of random variables X and Y , respectively. $H(X/Y)$ and $H(Y/X)$ are the conditional entropy.

4.2 Genetic Encoding and Decoding

We use the Locus-based adjacency representation [14] as the coding of an individual, which has two parts: the primary partition encoding and the overlapping partition one.

The primary partition encoding is denoted by $g = (g_1, g_2, \dots, g_{n_t})$, and n_t is the number of nodes in the t th network snapshot $G_t = (V_t, E_t)$. The value of g_i represents a node in the set of all neighbor nodes of the i th node, namely $g_i \in \{j | e_{ij} \in E_t\}$. It means that nodes i and g_i belong to the same community.

The overlapping partition encoding is $O = (O_1, O_2, \dots, O_{n_t})$, where O_i is a variable-length vector. Its length equals the number of neighbor nodes of node i . Each element of O_i is an integer in $\{0, 1\}$ and its value represents whether node i and the neighbor node represented by the element belong to the same community.

4.3 Genetic Operators

Crossover and mutation operators are key issues for a genetic algorithm. The two operators are illustrated in Fig. 1. Crossover operator is to swap genes from two parent individuals in the literature [14]. Mutation operator is to randomly select a position and then assign a random value to it according to a given mutation probability.

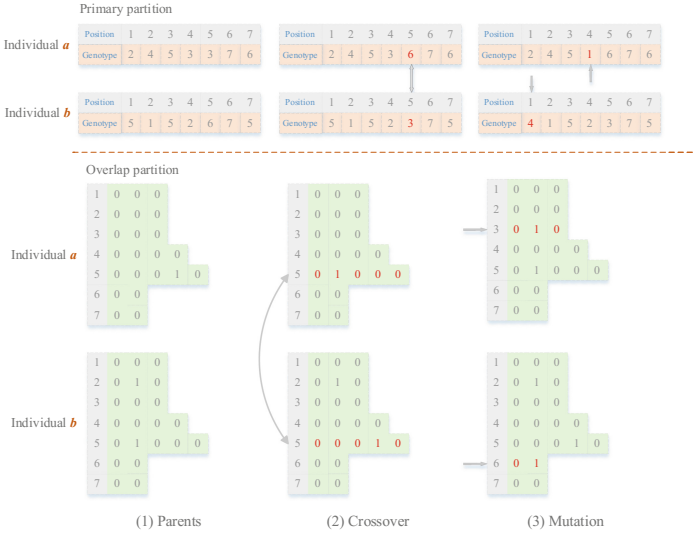


Fig. 1. The genetic operations.

4.4 MOEA/D Based Approach for Dynamic Overlapping Community Detection

Temporal networks can be regarded as a set of network snapshots. In this paper, we first devise a MOEA/D based approach for a network snapshot and then propose an approach for DOCDP.

MOEA/D decomposes DOCDP into a number of sub-problems. The three objectives are EQ , D and NMI_{LFK} . The decomposition strategy adopts Tchebycheff approach. The j th sub-problem can be expressed as follows [20]:

$$\begin{aligned} \text{Minimize } g(x|\lambda^j, z^*) &= \max_{1 \leq i \leq 3} \{ \lambda_i^j |f_i(x) - z_i^* | \} \\ \text{Subject to } x &\in \Omega \end{aligned} \tag{4}$$

where $\lambda^j = (\lambda_1^j, \lambda_2^j, \lambda_3^j)^T$, λ_i^j represents the weight coefficient of the i th ($i = 1, 2, 3$) objective function value for the j th sub-problem. $f_i(x)$ is the value of the i th objective function and Ω is the decision (variable) space. $z^* = (z_1^*, z_2^*, z_3^*)^T$ is the reference point, where $z_i^* = \max\{f_i(x)|x \in \Omega\}$ for each $i = 1, 2, 3$.

The MOEA/D based overlapping community detection approach for a network snapshot is described in Algorithm 1. For the first network snapshot, only objectives EQ and D are considered ($m = 2$) since it has no pre-snapshot. For other network snapshots, all the three objectives are considered ($m = 3$). To improve search efficiency, a sampling update strategy is proposed to reduce the number of evaluations, i.e., randomly selects a part of individuals from the population to perform genetic operations to generate new individuals. Individuals are randomly ordered first, and then

the top $N \times a$ individuals carry out genetic operations, where N and a are the number of sub-problems and the sampling ratio, respectively.

Algorithm 1: MOEA/D based overlapping community detection approach

Input: network snapshot $G_t = (V_t, E_t)$, the number of iterations n_{iter} , the number of sub-problems N , the number of objective functions m , the sampling ratio α , uniform spread of weight vectors $(\lambda^1, \lambda^2, \dots, \lambda^N)$, the number of neighbors of each weight vector P .

Output: A set of Pareto solutions, EP.

```

1. Initialization:
   Set EP =  $\emptyset$ .
   For  $i = 1$  to  $N$ :
     Find set  $B(i) = (i_1, i_2, \dots, i_p)$  to make  $\lambda^{i_1}, \lambda^{i_2}, \dots, \lambda^{i_p}$  are the closest  $P$  vectors to  $\lambda^i$ .
     Randomly produce individual  $x_i$ , and compute its objective function value by (4).
     Update the reference point  $z^i$  using  $x_i$ :
     For  $j = 1$  to  $m$ :
       If  $z^i < f_j(x_i)$ :
          $z^i = f_j(x_i)$ .
       End if
     End for
   End for
2. Update the population:
   For  $n = 1$  to  $n_{iter}$ :
     Sort the population  $X$  randomly.
     For  $i = 1$  to  $N \times a$ :
       Selected two indices  $k$  and  $l$  randomly from  $B(i)$ .
       Apply crossover and mutation operators to  $x_k$  and  $x_l$  to product a new solution  $y$ .
       Update reference point  $Z$  by the solution  $y$ :
       For  $j = 1$  to  $m$ :
         If  $z^i < f_j(y)$ :
            $z^i = f_j(y)$ .
         End if
       End for
       Update neighboring solutions:
       For  $p = 1$  to  $P$ :
         If  $g(y | \lambda^{i_p}, z^i) \leq g(x_{i_p} | \lambda^{i_p}, z^i)$ :
            $x_{i_p} = y$ .
         End if
       End for
       Remove all vectors from EP which dominated by  $y$ .
       If no individual in EP dominates  $y$ :
         add  $y$  to EP.
       End if
     End for
   End for
End for

```

The MOEA/D based approach for dynamic overlapping detection is described in Algorithm 2. The idea of dynamic evolutionary algorithm is to use the found community partition results of network snapshot G_{t-1} to initialize the population of G_t to make the algorithm converge more quickly. Generally speaking, adjacent network snapshots have similar community structures, such that they have similar community partitions. It means that a good solution (community partition) for G_{t-1} is probably a good solution for G_t . Thus, the population of G_t is composed of two parts: half of the population is randomly initialized and the other half is obtained by performing crossover and mutation operations on the individuals in EP generated for G_{t-1} .

Algorithm 2: MOEA/D based Approach for dynamic overlapping community detection

Input: dynamic network $G = \{G_1, G_2, \dots, G_T\}$.

Output: sets of non-dominant solutions for all network snapshots, $EPs = \{EP_1, EP_2, \dots, EP_T\}$.

```

1. If  $t == 1$ :
   Use Algorithm 2 (considering objectives  $EQ$  and  $D$ ,  $m=2$ ) to produce  $EP_1$ .
2. Else if  $t > 1$ :
   ① Initialization:
     50% of the initial individuals was randomly initialized and 50% came
     from performing crossover and mutation operations on  $EP_{t-1}$ .
   ② Update:
     Use Algorithm 2 (considering objectives  $EQ$ ,  $D$  and  $NMI_{LFK}$ ,  $m=3$ ) to produce
      $EP_t$ .
End if

```

5 Experimental Results

To validate the effectiveness of MDOA, it is compared with two dynamic overlapping community detection algorithms: SLPAD [7] and iLCD [6].

5.1 Data Sets

Synthetic Datasets. The three synthetic networks are generated by the Lancichinetti-Fortunato-Radicchi (LFR) benchmark network generator [21]. LFR provides many parameters to generate a desired network structure. Utilizing parameters listed in Table 1, LFR is used to generate 3 overlapping networks, which are considered as their first network snapshots. Let each overlapping network consist of 10 network snapshots. Each of the following snapshots is created by randomly adding 5 nodes and 10 edges to its former network snapshot and meanwhile removing 5 nodes and 10 edges from that snapshot.

Table 1. Parameters for synthetic networks.

Networks	n	k	$maxk$	mu	$minc$	$maxc$	on
SynNet_1	100	20	40	0.2	20	50	5
SynNet_2	200	20	40	0.2	20	50	10
SynNet_3	300	20	40	0.2	20	50	15

Real-World Datasets. The football dataset is a complex community network of U.S. Football League matches [9]. The network includes 121 vertices and 12 communities. We select the years 2001–2010 to constitute a real dynamic network containing 10 network snapshots. The VAST dataset is from IEEE VAST 2008 Challenge. It contains the telephone calls in 10 days and its community structure changes in those days [22]. The entire network contains 9,534 call records and 400 persons within 10 days in June 2006.

5.2 Experimental Results and Analysis

The three algorithms are applied to all the data sets mentioned above. Algorithm parameters are given in Table 2. Since iLCD is a deterministic algorithm, it runs once for each data set. SLPAD and MDOA are stochastic algorithms, such that 10 runs are done for each data set to obtain statistic results.

Table 2. Algorithm parameters.

Datasets	MDOA				SLPAD	iLCD	
	Sampling	Population	Crossover	Mutation	Threshold	Ratio	Threshold
SynNet_1	30%	100	0.6	0.05	0.3	0.25	0.2
SynNet_2	30%	200	0.6	0.02	0.3	0.25	0.15
SynNet_3	30%	300	0.6	0.01	0.3	0.25	0.2
Football	30%	150	0.5	0.02	0.3	0.15	0.3
VAST	30%	300	0.5	0.01	0.3	0	0.2

Experimental results on real-world datasets and Synthetic datasets are shown in Tables 3 and 4, respectively, where EQ_{avg} , D_{avg} and $NMI_{LFK_{avg}}$ represent the average value of EQ , D and NMI_{LFK} , respectively. Since MDOA gets a set of Pareto

Table 3. Comparison of network community detection results for real-world dataset.

Tested networks		MDOA			SLPAD			iLCD		
		EQ_{avg}	D_{avg}	$NMI_{LFK_{avg}}$	EQ_{avg}	D_{avg}	$NMI_{LFK_{avg}}$	EQ	D	NMI_{LFK}
Football	1	0.590	0.543	–	0.585	0.467	–	0.581	0.493	–
	2	0.553	0.487	0.507	0.551	0.426	0.262	0.535	0.457	0.293
	3	0.564	0.487	0.677	0.553	0.450	0.285	0.551	0.480	0.297
	4	0.616	0.514	0.677	0.608	0.478	0.308	0.581	0.511	0.221
	5	0.616	0.524	0.676	0.619	0.494	0.325	0.577	0.504	0.164
	6	0.588	0.514	0.673	0.582	0.495	0.343	0.558	0.502	0.281
	7	0.582	0.510	0.681	0.576	0.480	0.353	0.558	0.494	0.304
	8	0.589	0.533	0.685	0.583	0.479	0.355	0.571	0.516	0.274
	9	0.593	0.540	0.690	0.588	0.479	0.320	0.571	0.515	0.266
	10	0.593	0.542	0.699	0.589	0.486	0.295	0.568	0.519	0.281
VAST	1	0.654	0.341	–	0.653	0.331	–	0.193	0.016	–
	2	0.64	0.337	0.541	0.641	0.328	0.25	0.197	0.006	0.116
	3	0.63	0.335	0.69	0.631	0.325	0.25	0.252	0.009	0.394
	4	0.616	0.326	0.689	0.618	0.319	0.25	0.206	0.011	0.42
	5	0.604	0.316	0.695	0.605	0.311	0.25	0.199	0.01	0.379
	6	0.589	0.302	0.686	0.591	0.297	0.249	0.208	0.009	0.024
	7	0.579	0.299	0.686	0.581	0.295	0.25	0.218	0.006	0.246
	8	0.565	0.292	0.688	0.567	0.286	0.249	0.205	0.004	0.002
	9	0.55	0.29	0.69	0.553	0.284	0.25	0.193	0.012	0.283
	10	0.541	0.292	0.692	0.543	0.279	0.247	0.219	0.008	0.198

solutions in each run, the average value means the average of the maximum values in the 10 sets. For each synthetic dataset, the community structure becomes increasingly fuzzy along with the number of network snapshots increasing, and SLPAD is more sensitive to the clarity of the community structure. MDOA performs well in three objective functions regardless of whether the network structure is clear. For the real-world datasets, MDOA achieves better results than the other comparative approaches. Although SLPAD performs better than MDOA with respect of EQ for the VAST data set, MDOA has significant advantages in metrics of D and NMI_{LFK} .

Table 4. Comparison of network community detection results for Synthetic datasets.

Tested networks	MDOA			SLPAD			iLCD			
	EQ_{avg}	D_{avg}	$NMI_{LFK_{avg}}$	EQ_{avg}	D_{avg}	$NMI_{LFK_{avg}}$	EQ	D	NMI_{LFK}	
SynNet_1	1	0.403	0.409	–	0.400	0.340	–	0.350	0.332	–
	2	0.379	0.373	0.696	0.324	0.299	0.640	0.316	0.317	0.268
	3	0.370	0.338	0.697	0.270	0.259	0.653	0.310	0.305	0.397
	4	0.349	0.296	0.698	0.240	0.230	0.646	0.290	0.274	0.617
	5	0.327	0.270	0.698	0.201	0.211	0.515	0.261	0.254	0.523
	6	0.327	0.259	0.696	0.187	0.196	0.385	0.249	0.244	0.323
	7	0.319	0.241	0.697	0.011	0.143	0.265	0.249	0.229	0.275
	8	0.290	0.191	0.699	0.011	0.132	0.182	0.217	0.189	0.619
	9	0.295	0.187	0.695	0.011	0.127	0.182	0.216	0.177	0.367
	10	0.299	0.179	0.695	0.011	0.121	0.182	0.228	0.171	0.209
SynNet_2	1	0.548	0.27	–	0.546	0.266	–	0.527	0.265	–
	2	0.534	0.268	0.651	0.531	0.263	0.512	0.523	0.26	0.470
	3	0.526	0.261	0.695	0.527	0.259	0.517	0.502	0.258	0.441
	4	0.51	0.258	0.695	0.505	0.252	0.545	0.461	0.255	0.455
	5	0.496	0.251	0.69	0.491	0.241	0.547	0.447	0.244	0.374
	6	0.483	0.245	0.694	0.434	0.213	0.538	0.437	0.24	0.429
	7	0.45	0.224	0.693	0.423	0.204	0.536	0.419	0.225	0.503
	8	0.455	0.223	0.691	0.44	0.21	0.541	0.39	0.213	0.503
	9	0.431	0.21	0.693	0.324	0.174	0.537	0.375	0.21	0.481
	10	0.417	0.211	0.694	0.20	0.148	0.467	0.344	0.20	0.491
SynNet_3	1	0.653	0.339	–	0.65	0.333	–	0.638	0.332	–
	2	0.639	0.334	0.545	0.639	0.327	0.296	0.62	0.333	0.307
	3	0.626	0.326	0.693	0.628	0.32	0.298	0.61	0.332	0.342
	4	0.617	0.318	0.693	0.619	0.309	0.302	0.593	0.323	0.384
	5	0.605	0.313	0.693	0.604	0.299	0.304	0.58	0.312	0.369
	6	0.592	0.303	0.692	0.594	0.292	0.307	0.57	0.301	0.321
	7	0.581	0.298	0.694	0.582	0.29	0.312	0.556	0.299	0.294
	8	0.573	0.295	0.695	0.57	0.289	0.316	0.545	0.292	0.35
	9	0.556	0.286	0.694	0.557	0.275	0.32	0.531	0.290	0.379
	10	0.546	0.284	0.694	0.547	0.27	0.317	0.519	0.285	0.366

To intuitively show Pareto solutions found by MDOA, SynNet_1 data set is taken as examples to show their solutions obtained by the 3 approaches. In Fig. 2, “×” represents the solutions generated by MDOA in 10 runs. “▲” and “●” represent

solutions generated by SLPAD in 10 runs and iLCD in one run. Due to space limitation, only 4 network snapshots are chosen for SynNet_1 data set to show their solutions. MDOA generates a set of solutions, most of which are superior to those of comparative approaches in both terms of modularity and partition density.

5.3 The Effect of Dynamic Optimization and Sampling Strategy

To investigate the effect of dynamic optimization technique in reducing the computational complexity, 10 runs of MDOA are done for SynNet_1. Parameters of MDOA are the same as those in Table 2. EQ and D values obtained by MDOA for each snapshot are shown in Fig. 3. EQ (D) refers to the average of the best EQ (D) values found in each run of MDOA.

In Fig. 3, Random 100 and 30 represent the average results obtained by running MDOA with a random initial population for 100 and 30 generations, respectively. Dynamic 100 and 30 represent the average results of running MDOA with the dynamic optimization strategy for 100 and 30 generations, respectively. EQ and D values of Dynamic 30 are slightly lower than those of Dynamic 100. Both of Dynamic 30 and 100 get better results than Random 30 and 100. Therefore, using the community partition results from last network snapshot makes the algorithm converge more quickly.

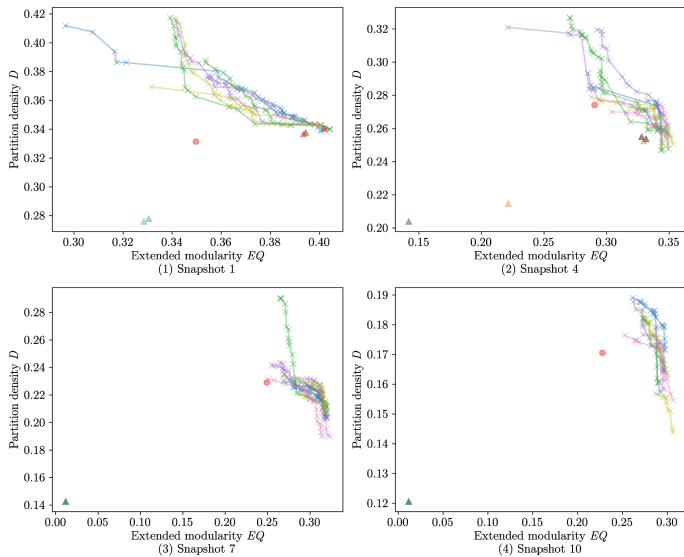


Fig. 2. Pareto solutions generated by our approach and comparative approaches on SynNet_1 dataset.

The sampling ratio a of MDOA is given as 5%, 10%, 30%, 50%, 80% and 100%. Other parameters are the same as those in Table 2. For each sampling ratio, 10 runs of

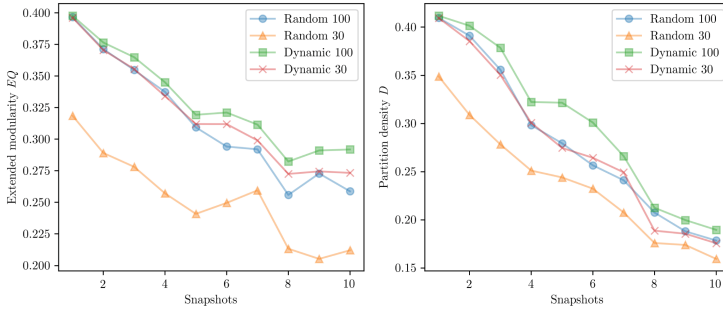


Fig. 3. The effect of dynamic optimization technique.

MDOA are done for each snapshot of SynNet_1. For each snapshot, the average EQ and D values obtained by MDOA with different a values are shown in Fig. 4. We can see that the values of EQ and D for each snapshot are similar when the sampling ratio is greater than 30%. Since the computational effort is in proportion to the sampling ratio, about 70% of computational time is saved when we set a to be 30%.

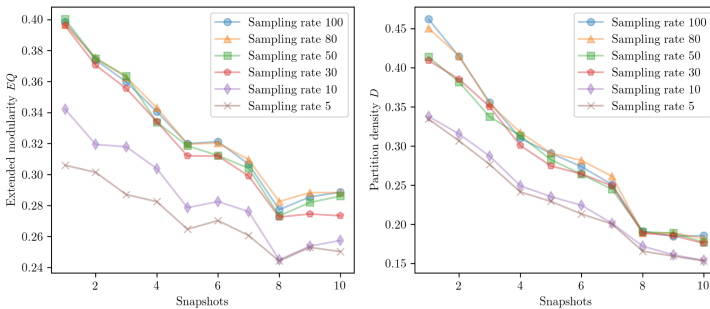


Fig. 4. Effect of the sampling update strategy.

6 Conclusion

In this paper, we propose a MOEA/D based approach for dynamic overlapping community detection problem, with consideration of three objectives of extended modularity (EQ), partition density (D) and improved mutual information (NMI_{LFK}). The problem is to detect the community structures for a set of network snapshots. For each snapshot, MOEA/D is used to detect its communities by decomposing the multi-objective community detection problem into a number of single objective sub-problems.

Experimental results show that the proposed approach can effectively detect the overlapping communities in temporal networks and find high quality Pareto solutions.

Acknowledgments. This work was supported by National Natural Science Foundation of China under Grant 61873040 and 61374204.

References

1. Atay, Y., Koc, I., Babaoglu, I., Kodaz, H.: Community detection from biological and social networks. *Appl. Soft Comput.* **50**, 194–211 (2017)
2. Chintalapudi, S.R., Prasad, M.H.M.K.: A survey on community detection algorithms in large scale real world networks. In: 2nd International Conference on Computing for Sustainable Global Development, pp. 1323–1327. IEEE Press, New York (2015)
3. Feng, H., Tian, J., Wang, H.J., Li, M.: Personalized recommendations based on time-weighted overlapping community detection. *Inf. Manag.* **52**(7), 789–800 (2015)
4. Muslim, N.: A combination approach to community detection in social networks by utilizing structural and attribute data. *Soc. Netw.* **5**(1), 11–15 (2016)
5. Sul, W.J.: Microbial community analysis assessed by pyrosequencing of rRNA gene: community comparisons, organism identification, and its enhancement. Dissertations and Theses - Gradworks. The Michigan State University, East Lansing (2009)
6. Cazabet, R., Amblard, F., Hanachi, C.: Detection of overlapping communities in dynamical social networks. In: IEEE Second International Conference on Social Computing, pp. 309–314. IEEE Press, New York (2010)
7. Aston, N., Hertzler, J., Hu, W.: Overlapping community detection in dynamic networks. *J. Softw. Eng. Appl.* **7**(10), 872–882 (2014)
8. Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. *IEEE Trans. Evol. Comput.* **18**(4), 577–601 (2014)
9. Gong, M., Zhang, L., Ma, J., Jiao, L.: Community detection in dynamic social networks based on multiobjective immune algorithm. *J. Comput. Sci. Technol.* **27**(3), 455–467 (2012)
10. Shen, H., Cheng, X., Cai, K., Hu, M.B.: Detect overlapping and hierarchical community structure in networks. *Phys. A* **388**(8), 1706–1712 (2009)
11. Ahn, Y.Y., Bagrow, J.P., Lehmann, S.: Link communities reveal multiscale complexity in networks. *Nature* **466**(7307), 761 (2010)
12. Lancichinetti, A., Fortunato, S., Kertész, J.: Detecting the overlapping and hierarchical community structure of complex networks. *New J. Phys.* **11**(3), 19–44 (2008)
13. Lancichinetti, A., Radicchi, F., Ramasco, J.J., Fortunato, S.: Finding statistically significant communities in networks. *PLoS ONE* **6**(4), e18961 (2010)
14. Zhan, W., Guan, J., Chen, H., Niu, J., Jin, G.: Identifying overlapping communities in networks using evolutionary method. *Phys. A* **442**, 182–192 (2013)
15. Ma, X., Dong, D.: Evolutionary nonnegative matrix factorization algorithms for community detection in dynamic networks. *IEEE Trans. Knowl. Data Eng.* **29**(5), 1045–1058 (2017)
16. Zhou, X., Liu, Y., Li, B., Sun, G.: Multiobjective biogeography based optimization algorithm with decomposition for community detection in dynamic networks. *Phys. A* **436**, 430–442 (2015)
17. Chen, X., Sun, H., Du, H., Huang, J., Liu, K.: A centrality-based local-first approach for analyzing overlapping communities in dynamic networks. In: Kim, J., Shim, K., Cao, L., Lee, J.-G., Lin, X., Moon, Y.-S. (eds.) PAKDD 2017. LNCS (LNAI), vol. 10235, pp. 508–520. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57529-2_40

18. Xu, B., Deng, L., Jia, Y., Zhou, B., Han, Y.: Overlapping community detection on dynamic social network. In: Sixth International Symposium on Computational Intelligence and Design, pp. 321–326. IEEE Press, New York (2013)
19. Zhang, Q., Li, H.: MOEA/D: a multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.* **11**(6), 712–731 (2007)
20. Ma, J., Liu, J., Ma, W., Gong, M., Jiao, L.: Decomposition-based multiobjective evolutionary algorithm for community detection in dynamic social networks. *Sci. World J.* **2014**, 22 (2014)
21. Basu, S., Banerjee, A., Dey, A., Mukherjee, S., Pan, I.: Clustering by feature optimization for static community detection. In: 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, pp. 1936–1939. IEEE Press, New York (2017)
22. Montanari, A., Sen, S.: Semidefinite programs on sparse random graphs and their application to community detection. In: ACM SIGACT Symposium on Theory of Computing, pp. 814–827. ACM, New York (2016)



Research on Public Opinion Communication Mechanism Based on Individual Behavior Model

Weidong Huang and Yang Cui^(✉)

Nanjing University of Posts and Telecommunications, Nanjing, China
cuiyang9515@163.com

Abstract. With the development of the Internet era, the changes and spread of online public opinion have gradually become the mainstream of the change and spread of public opinion in the world today. And the development trend of online public opinion directly or indirectly affects daily life, so the research on the communication mechanism of network public opinion can provide reference for the work of the public opinion monitoring department. At the same time, the research has more important practical significance for further exploration of network public opinion. However, a large part of the current research does not focus on the individuals of netizens. The different behaviors caused by differences lead to changes in the communication mechanism. In order to solve this problem, the simulation method is used to deduct the evolution mechanism of network public opinion, and the overall model and impact model based on individual behavior are mainly used to study the influence of communication mode and individual multi-dimensional characteristics of netizens on hotspot public opinion transmission. The analysis and more accurately reflect the actual law of the network public opinion transmission process under the influence of individual behavior. The application value and significance of studying these contents is to make the obtained public opinion communication mechanism model and the law of public opinion evolution provide a certain reference for the regulatory authorities to adopt corresponding regulatory measures. At the same time, it can also benefit the network society and provide better help for maintaining network security and network communication order.

Keywords: Individual behavior · Public opinion
Public opinion communication · Simulation modeling

1 Introduction

In today's information society, it's no longer just social affair that concerns the entire society, The development of online public opinion [1] is closely related to the development of social situation trends. Along with the continuous innovation of high and new technology and the continuous advancement of scientific and technological means, data and information have exploded into people's daily life. The data and information have affected people's daily life and the behavior of people's public life. Sina Weibo is the most common social tool for online social platforms. It has a sensational

communication and fast message speed, and users have various ways of behavior on this platform. Therefore, the analysis of the public opinion communication mechanism of the Weibo platform is more representative.

Further analysis of the literature found that the focus and direction of foreign research on online public opinion is different from that in China because the Western political and social model is relatively stable, and the freedom of public opinion has a long history. Compared with our free environment, it is less free, so the threat of public opinion is lower. So there is relatively little research on this aspect. Domestic research on public opinion mainly studies the infectious disease model [2] as a basic model for studying information dissemination mechanisms. Among them, the SIR model is similar to the information dissemination process, so it is very suitable to be applied to the study of the public opinion transmission mechanism. There are many different perspectives on the study of public opinion communication, but the novelty of the public opinion transmission mechanism based on individual behavior patterns is based on the fact that individual cognition is derived from actual individual behavior, and then a basic system of comparative systems is introduced. The behavioral model, and then based on this model, is more accurate. The simulation of the time-dependent propagation of a public opinion event is carried out, and finally the law of public opinion transmission based on individual behavior is obtained.

2 Individual Behavior Patterns and Simulation Ideas

2.1 Individual Behavior Model – Subject Model

As far as the netizen is concerned, it is a complex system. He has his own thoughts, emotions, values, outlook on life, etc. These are the identifying attributes that he already possesses. What the society gives him is also an attribute that belongs to him only, such as authority, degree of certification, and social status. These individual attributes will have a certain impact on language interaction and communication exchange on the social platform, and will have a certain degree of effect on their own published content and other people's published content. A commentary and a paragraph on the Internet can reflect the emotional polarity and strong emotion of the publisher. When we discriminate the emotional polarity, we can initially reflect the emotional tendency of the publisher. When these emotions are further quantified, the emotional intensity can be more accurately reflected. According to the characteristics of the subject itself, its behavior can also be changed according to the characteristics of its subject. Therefore, when studying the public opinion transmission in the individual behavior pattern [3], we can study the transmission mechanism of a public opinion event by dividing different dimensions. For example, the characteristics of the subject: gender, age, interest, influence, etc. In this way, the influence of individual behavior based on the subject model on the public opinion communication will be studied in more detail from different aspects of individual behavior, so that the final rule is more comprehensive.

2.2 Individual Behavior Pattern—Impact Model

The impact model best reflects the interactions of people on social networking sites. On the Internet platform, it is embodied in the continuous dissemination of information, that is, the forwarding function on social platforms such as Weibo. It can describe how individuals interact with each other throughout the process of information dissemination and even the spread of public opinionism (see Fig. 1).

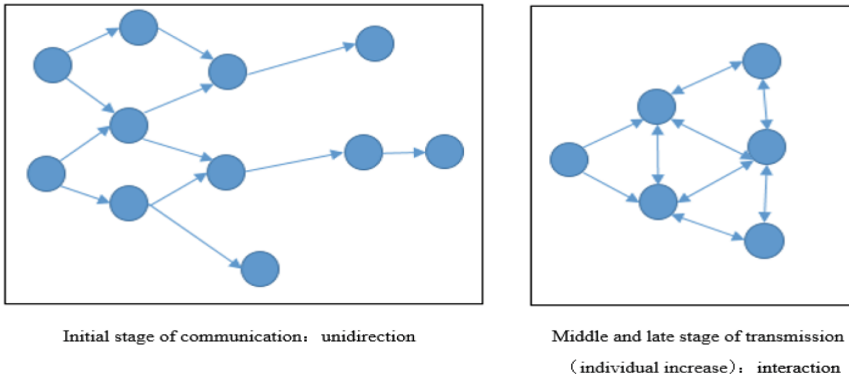


Fig. 1. Impact model

2.3 Analysis of Individual Behavioral Factors

As a part of society, people bear great responsibility and far-reaching influence. Only by quantitatively analyzing people’s behavior can we study and solve problems more accurately and persuasively. Man is a complex system in itself and the behavior he produces is caused by different behaviors. Coupled with many uncertain external factors, the whole process is a very complicated process. First we have to know what is the process of individual behavior. This will adjust the research perspective to the order from the result to the cause. First, we have to recognize the fact that individual behavior is a transition from individual cognition. Then, we can divide individual cognition into four parts: feeling, analysis, decision, and action. Then study the causes of the impact of individual cognition. The influencing factors are mainly from macro and micro, that is, subjective and objective. This paper divides these factors into personal factors, environmental factors, social factors, and technical factors [4] (see Fig. 2).

According to the influencing factors, the specific composition of the available individual behavior patterns can be further studied. In this picture, we can clearly see the individual inside, the human being, which does not exist as a single individual, but shows a framework of complex in the complex and versatile subjective and objective factors [5], thus paving the way for subsequent quantitative research (see Fig. 3).

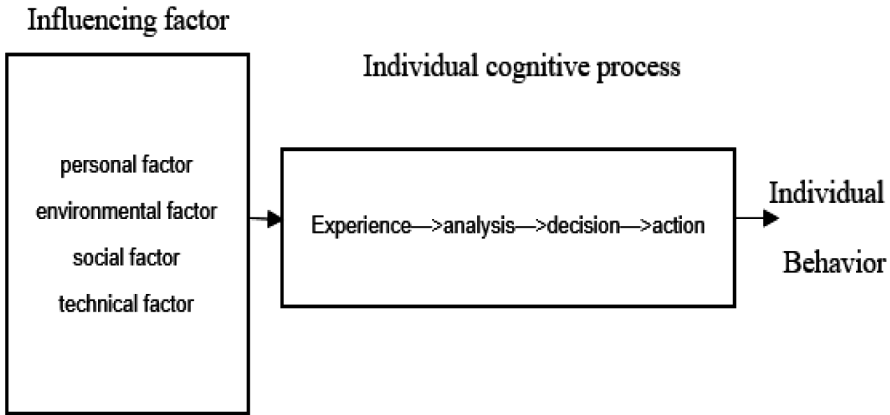


Fig. 2. The process of individual behavior

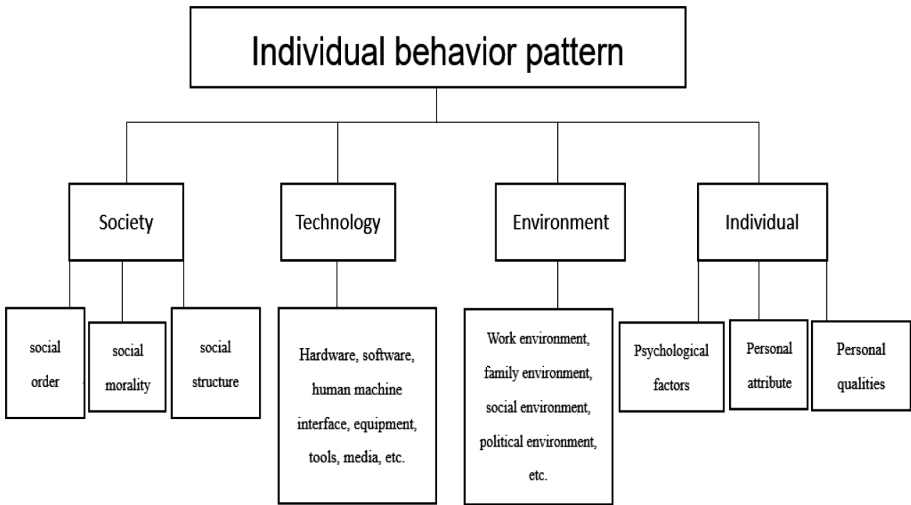


Fig. 3. The composition of individual behavior pattern

2.4 Individual Behavior Dimension Classification

Combined with the characteristics of the selected public opinion events, plus the differences between individuals, each individual has its own unique feature identification and habit difference. Even the same individual, he will be affected by information sources and environmental factors to produce different individual behaviors [6]. In the selected events, because cervical cancer vaccines are required for both age and gender, age is a dimension from the perspective of the subject; secondly, the gender of the vaccinated subjects is also important, because not only women are The main host of cervical cancer, the male itself also has the incentive to produce cervical cancer, so

gender is also a dimension from the perspective of the subject. From the perspective of the overall impact, it is also possible to analyze the dimensions through the influence of the selected subject, the relationship between the subject and the subject, and the interaction between them.

2.5 Individual Influence and Interaction

In the big environment of the Internet, all kinds of netizens exist, and they all have a large or small influence, which in turn causes other netizens to produce different degrees of behavior. Therefore, as a more important source of individual behavior, analyze the individual behavior characteristics related to individual influence, such as the individual's professional credibility and individual's popularity. In order to better quantify the analysis rather than qualitative analysis alone, the collected data are based on the information data of the well-known authoritative bloggers of Weibo, which is both easy to quantify and a summary of the appeal characteristics.

From the perspective of research, it is mainly to study the performance of individuals in information dissemination. In short, the manifestation of individual influence is mainly reflected by the number of Weibo content forwarding. The ever-increasing number of forwards indicates that this individual has a great influence. In addition, many studies have shown that the number of fans owned by an online registered user also indirectly or directly affects the influence of their bloggers. Because the more fans, the more people see bloggers spreading content, the more likely they are to be forwarded. Therefore, the number of fans and the number of shares can be used as the evaluation criteria and measurement of individual influence. From the perspective of communication mode, Weibo has a one-to-one and one-to-one communication form. Netizens can send information to one friend in one-to-one form, and can also spread multi-point information to netizens in their circle of friends. The kind of netizen forwarding behavior structure constitutes the link network structure of Weibo public opinion information on the basis of social network. Different netizens have different roles in the social network chain, and their forwarding behavior has different effects on public opinion communication. Some are at the core of information dissemination and become key points; most netizens are at the edge of information dissemination and become non-critical points of information dissemination [7]. Therefore, the relationship and role between the subject and the subject can be reflected by forwarding.

3 Description and Conditional Hypothesis of Public Opinion Communication Model Based on Individual Behavior Model

The article refers to the method of SIR infectious disease model to the study of public opinion transmission mechanism. Based on the theory of infectious disease model and the public opinion communication mechanism based on individual behavior patterns, the model is established to obtain the law. In the process of public opinion communication, the person who does not know the public opinion information is the unknown person, that is, S; the person who already knows the public opinion information and may forward the comment is recorded as the proliferator, which is I; the person who is

exposed to the public opinion information again but does not forward the diffusion is the immunized person. R ; and need to make assumptions about the state, attributes, etc. of each subject and object in the model: (1) as the original data in the model, the number of subjects and the initial state in the studied public opinion events are known; (2) Research The type factor of each subject is not considered, that is, the degree of perception is the same for the influence of external factors; (3) The public opinion events that occur in the study are assumed to be from the source of great influence, and the network media The scope of communication is the largest, followed by traditional media, and the average spread of ordinary netizens is the smallest. And assume that traditional media will affect unknowns and immunizations in the system.

4 Model Simulation and Demonstration

4.1 The Introduction of Netlogo

Whenever a problem appears in front of us, we tend to abstract and simplify the problem, and the model plays a big role at this time. And by building the model, we can visually see which solutions are feasible to get the final solution. Simultaneous modeling is also a low-risk, low-cost solution. The multi-agent model environment built by the Netlogo platform interacts with individuals, making individual changes the basis of the entire system change. It mainly simulates the behavior of microscopic individuals and the appearance of macroscopic models and the connection between the two [8]. The application scope of Netlogo is mainly a complex system that can be changed significantly over time.

4.2 The Simulation Experiment of Netlogo

According to the Baidu Index, the age distribution and gender distribution of cervical cancer were searched nationwide from April 1, 2018 to April 30, 2018 (see Fig. 4).

Of the 1000 comments collected, there were 856 female messages and 144 male messages. The proportion is about 4:1. The 30–39 age group and the 40–49 age group are mostly consistent with the Baidu index cervical cancer vaccine search and comment ratio. The figure below shows the spread of the initial outbreak and the end of the outbreak (see Fig. 5).

4.3 Experimental Results and Comparison

As for the cervical cancer vaccine, it should not directly reflect the public opinion, but as the event continues to ferment, it can reflect the attention of the netizens to the incident and the concern according to their age. The depth is also different, and it also reflects the depth of the public opinion event (see Fig. 6).

According to the experimental results, whether it is the age dimension or the gender dimension, the direction of public opinion communication is ultimately explosive, and the scope and speed are also expanding, eventually reaching a threshold. Therefore, it can be concluded that public opinion communication is based on individual behavior

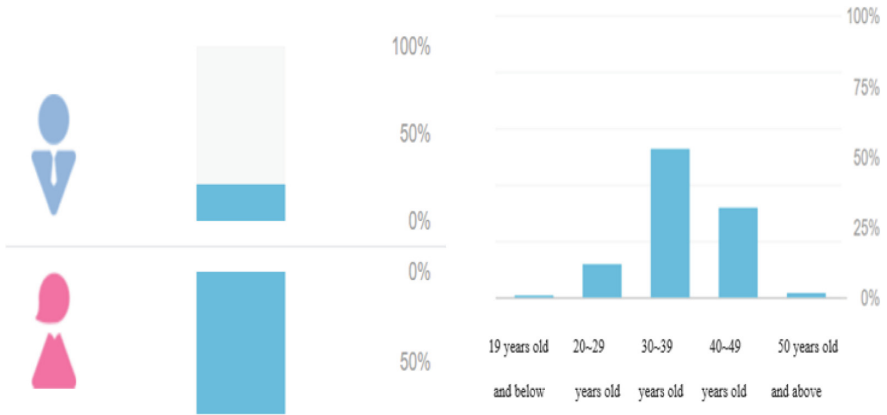


Fig. 4. The data from Baidu Index

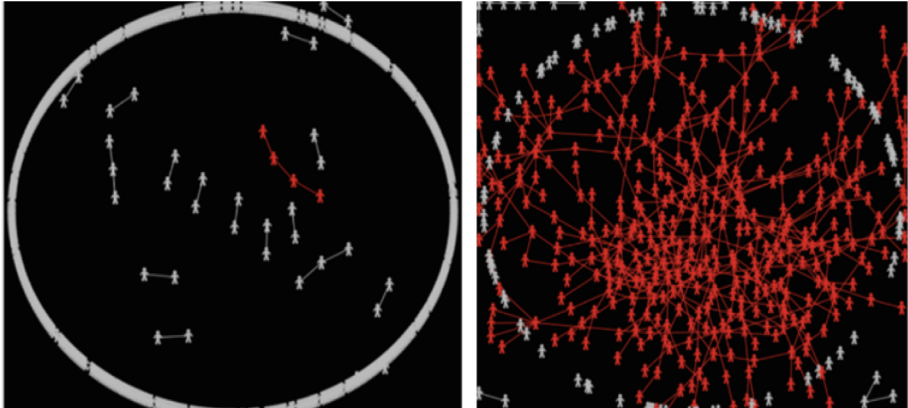


Fig. 5. The experimental result

and is directly affected by the gradual fermentation of public opinion events, forming a reunion interaction, and finally reaching the peak of communication, that is, because of the particularity of the event, Part S (susceptible, here refers to the 20–29 age group and female netizens) joined the initial dominant I (infected, here refers to the 40–49 age group and male netizens), and gradually replaced them to become mainstream. This is the target of the public opinion event itself, which ultimately leads to the ultimate public opinion orientation. Therefore, the individual behavior from the role of individual characteristics to the interaction affects the public opinion event and finally develops towards the ultimate result that matches the nature of its own event.

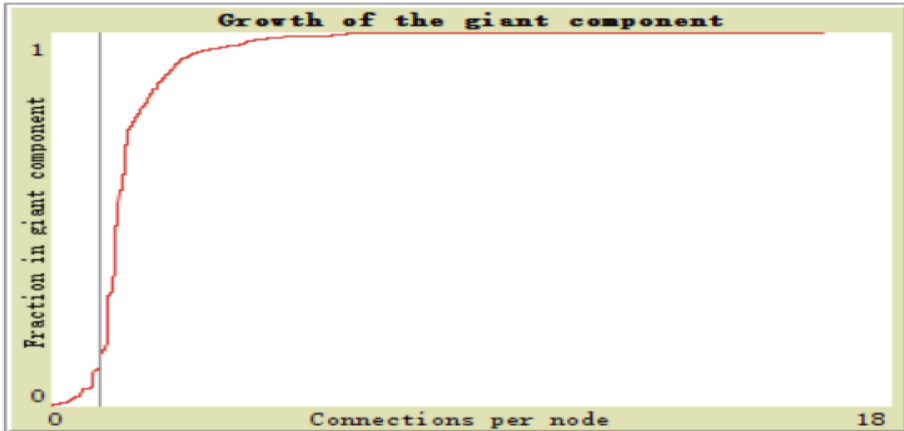


Fig. 6. The trend chart

5 Conclusion

Through the research of public opinion communication mechanism based on individual behavior patterns, the individual behaviors of multiple dimensions of collected data sets are used to analyze and design a certain public opinion event propagation process, and the approximate and accurate public opinion communication mechanism based on individual behavior patterns is obtained. law. The Netlogo simulation platform is used to visualize the entire communication mechanism and compare it with the existing model to obtain a more convincing summary of the propagation mechanism. This paper mainly produces the following results and conclusions:

- (1) Through the existing research theories: individual behavior factor diagnosis model and online social network collaborative recommendation framework based on user behavior cognition, individual behavior patterns, etc. Dimensional classification of selected public opinion events, so that the color of individual behaviors occupies a basic position. Become a means and basis for studying public opinion communication. It is divided into individual attributes, that is, the factors that influence the influence of public opinion transmission are constructed according to the subject model; then they are divided into the overall attributes, that is, the overall model interacts between individuals and individuals to produce the basic operation process of public opinion communication. Continue to ferment and change.
- (2) By analyzing the types of individual participants generated by individual behaviors during Weibo communication, the individual is divided into five categories: communicator, audience, content, media and feedback effects, by refining and quantifying different types of individuals. Features, and use data crawling function and data analysis classification method to achieve the analysis of individual behavior. Studying the problem of information dissemination from the perspective of individual behavior has deepened the research depth of problems in

this field. At the same time, the classification and dimension division of individual behaviors have provided the possibility and laid the foundation for the reconstruction of the subsequent information dissemination model.

- (3) Based on the theoretical basis of the infectious disease model-SIS model and simulation platform, the effects of the individuals and individuals selected for the event “cervical cancer vaccine should not be beaten” on the transmission of public opinion events are presented in the form of graphs. The scope of the propagation is expressed by the range, that is, the frequency of the event itself, and the internal and horizontal comparisons are made with the red and white graphics, and finally the effect corresponding to the model is obtained, that is, in the infectious disease model-SIS model, S The higher the proportion of individuals (20–29 years old and female-transmitted individuals in the experiment), the wider the spread of information and the faster the spread. The type I individuals (40–49 age group and male-transmitted individuals in the experiment) also experienced a decrease in the change due to the event, because the medical event was continuously introduced with the price of the vaccine, and the change of public opinion, 40- The 49-year-old and male netizens are gradually withdrawing from the spread because their own vaccination restrictions have led them to believe that the vaccination has no practical significance. Therefore, individual behaviors are interrelated and affect each other in the process of the transmission of public opinion events. Individual behaviors from the role of individual characteristics to interactions affect the public opinion events and eventually develop towards the ultimate result that matches their own event attributes.

References

1. Erdos, P., Reny, A.: On the evolution of random graphs. *Publication of the Mathematical Institute of the Hungarian Academy Offences*, vol. 38, no. 1, pp. 17–61 (1961)
2. Zhou, H., Zhang, W., Lao, P.J., et al.: Group convergence effect and its influence mechanism in network interaction. *Science & Technology Progress and Policy*, vol. 13, pp. 68–72 (2014)
3. Zhou, L.: *Research on the evolution mechanism of group behavior in mass emergencies*. University of Science and Technology of China, Hefei (2015)
4. Zhang, T., Zhang, W.: Study on education public opinions: exploring effective path from Tsinghua University Education Research, vol. 5, pp. 102–107 (2011)
5. Milgram, S.: The small world problem. *Psychology Today* **32**(2), 185–195 (1967)
6. Chen, Y.H., Li, G.: Study on the public opinion communication process of public emergencies in public emergencies-from the perspective of network organization. *J. Inf.* **34**(2), 43–46 (2015)
7. Sznajd Weron, W., Sznajd, J.: Opinion evolution in closed community. *Int. J. Mod. Phys. C* **11**(06), 1157–1165 (2000)
8. Ping, L., Zong, L.Y.: Study on Weibo information communication based on social network centrality analysis-taking Sina Weibo as an example document. *Inf. Knowl.* **13**(6), 92–97 (2010)



A Comprehensive Evaluation: Water Cycle Algorithm and Its Applications

Rana Muhammad Sohail Jafar¹, Shuang Geng^{1(✉)}, Wasim Ahmad¹,
Safdar Hussain^{1,2}, and Hong Wang¹

¹ College of Management, Shenzhen University, Shenzhen 518060, China
gracegeng0303@163.com

² Pir Mehr Ali Shah, Arid Agriculture University, Rawalpindi 46000, Pakistan

Abstract. Recently nature-inspired optimization algorithms have become a popular choice for solving complex optimization problems. Water Cycle Algorithm (WCA) is a nature-inspired new optimization technique, which has successfully applied to solve the constrained optimization and engineering design problems. As a result, the WCA studies have extended significantly in the last 5 years. This review paper provides the comprehensive assessment of WCA in the area of modifications, hybridizations, and applications. Moreover, it will provide the awareness to the researchers how the current algorithm can be modified according to the nature of the problems. The narrative of how WCA was used in the tactics for solving these kinds of problems. Future research directions are also discussed based on the comprehensive conclusion as well as discussion. To the best of our knowledge, this is the first review article which has enclosed extensive information about the WCA and its applications.

Keywords: Water cycle algorithm (WCA) optimization algorithms
Artificial intelligence

1 Introduction

Among optimizations ways and means, metaheuristic approaches have proven their abilities in providing best solutions to real life problems, when the other methods sometimes are unable to provide the best solution of problems within a reasonable time. Especially, when several local minima hemmed in global minimum. The notions of such optimizer are usually motivated by observing natural phenomena. For example, Genetic algorithms (GAs) [1], simulated annealing (SA) [2], particle swarm optimization (PSO) [3], ant colony optimization (ACO) [4], are all inspired from nature [5].

The water cycle is a new optimization technique which has solved the many real-world problems; nowadays this algorithm has been widely using in different ways to address the complex problems. WCA was proposed by Eskandar et al. [6], which is employed in various constrained optimization and engineering contrive problems. The key concepts and celebrations which lie within the projected method is inspired by nature and based on the observance of water cycle process and precisely how rivers and streams flow towards the sea in the real world. However, there are abundant

engineering optimization problems prevail in real-world, multifaceted and challenging to solve.

‘Akin’ to PSO algorithm the WCA is also a population-oriented and nature-inspired algorithm which introduces a unique metaheuristic method for optimizing coerced functions and engineering complications. The performance of WCA was examined on common constrained optimization problems, and the obtained results were matched with other optimizer’s results regarding function number and evaluations. Furthermore, evaporation and raining process worked as mutation operator to prevent WCA from getting trapped in local optima [6]. Many researchers (Haddad and Moravej, et al. [7]; Lenin, Reddy, and Kalavathi [8]; Jabbar and Zainudin [9]; Guney and Basbug [10]; Sadollah et al. [11, 12]; Zhu et al. [13]) have used the water cycle in different ways to solve the complicated engineering optimization problem by utilizing WCA. The downsides regarding effectiveness and precision of prevailing numerical methods have stimulated scholars to depend on metaheuristics based on nature-inspired techniques to solve engineering optimization hitches. Therefore, this algorithm usually used by combining different rules of natural phenomena [14]. The evolutionary algorithms (EAs) are commonly known as common purpose algorithms within the optimization methods are acknowledged as to be most proficient in finding the near-optimum solution to the numerical real-valued test problems. EAs have been magnificently answered the constrained optimization problems. PSO performs multi-dimensional search and used the velocity vector to update the current position of each particle in the swarm [3].

The primary objectives of this article are to furnish the WCA broad applicability in various areas, as well as bringing it future challenges and opportunities. Up till the writing of this review article, there is no study which has provided the comprehensive review of WCA. Therefore, this study will provide the researchers a comprehensive discussion of the applicability and usability of WCA. Note that this article categorized the studies on WCA based on its modifications such as multi-objective-based, Gradient-based, and hybridizations as well as on its applications. So, this categorization aims at simplifying the understanding of the improving trends in the WCA.

2 Fundamental Structure of WCA

Nature speaks lots of secret languages, those who try to understand the nature’s secret language make an invention in this world. Some nature-inspired optimization procedures have been established in the last two decades [15, 16]. Similarly, the idea of WCA was established on natural phenomenon, based on the observation of water how it completes its cycle naturally. To recognize this phenomenon, we should understand that how streams and rivers flow downhill towards the sea in the real world. The streams or rivers are formed when snow or glaciers melt at the top of the mountains. Then this stream or river flows downhill from one place to another, and their journey is eventually ending up in a sea. In this process of downhill, a lot of events occur such as rain and water’s evaporation. During the evaporation, process water vanished away from lakes and rivers, while plants emit water in photosynthesis. In the atmosphere, this evaporated water converted into clouds and when these clouds smash with cold winds

then they produce rain, and water again reached back to earth. This whole natural setup is known as the hydrologic cycle (water cycle) as presented in Fig. 1 [17]. To explore a new region of the search space evaporation and raining condition perform excellently and streams movement to exploiting the neighboring solutions. As a result, alteration of steps can affect the overall performance of WCA [18]. In order to employ WCA, required steps are explained as follows:

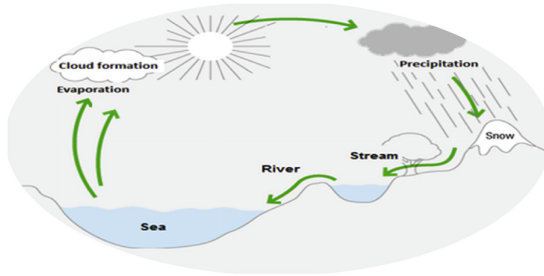


Fig. 1. Water cycle process in the real world by Deihimi et al. [19].

2.1 The Steps and Principles of WCA

Similar to many other metaheuristic algorithms, the WCA also starts with an initial population called raindrops. Suppose that we have rain, the best drop of rain will represent a sea, the sound raindrops are serving here like a river, and the rest of the raindrops are streams that flow into the rivers and seas. Depending on the size of the river represented in the subsequent each river take up water from the streams. Although, the amount of water in a stream that flows into rivers and the sea deviates from other streams. Also, rivers flow into the sea, which is the most downhill point on earth [6]. Below given Fig. 3 display the flow chart and Table 1 describes the fundamental steps of WCA.

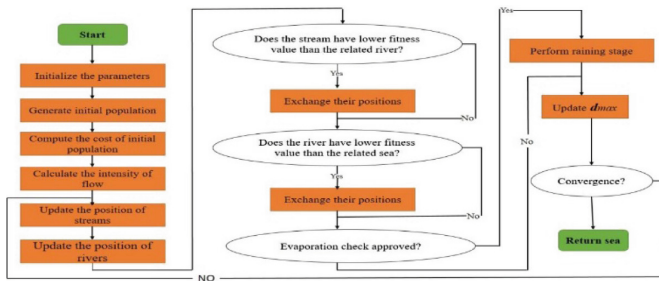
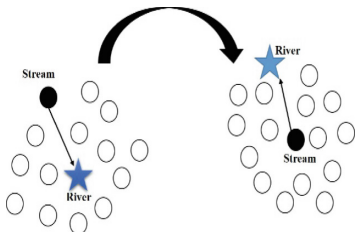


Fig. 3. Flowchart of the proposed WCA.

Table 1. Steps of WCA

<p>Step 1: Selection of initial parameters</p> <p>Select the initial parameters of the WCA: N_{sr}, d_{max}, N_{pop}, $max_iteration$, etc.</p> <p>Step 2: after the selection of initial parameters create initial population and streams by using Eqs. (2), (3), and (4). In these given equations, stream, river and the sea demonstrate $1 \times N_{var}$ dimensional arrays, the solutions are defined as $stream = [x_1, x_2, x_3, \dots, x_N]$ N_{sr} is the total number of rivers and sea, N_{stream} denotes a total number of streams.</p> <p>Step 3: Estimate the value (cost) for each raindrop The population size is represented here as N_{pop}. Some of the best solutions (N_{sr}) are deliberated as rivers, and the best river is taken as the sea in Eq. (5)</p> <p>Step 4: Determine the intensity of flow for rivers and sea As the following Eq. (6) is given: N_{Sr} is the number of streams, which flow to the specific rivers or sea.</p> <p>Step 5: Movement of streams and rivers The movement of streams towards the rivers can be analyzed as given by Eq. (7). Where $rand$ denoted as a uniformly distributed random number, and its value between 0 and 1. The exchange of positions will happen only when the solution provided by a <i>stream</i> is better than its joining <i>river</i>, then positions of <i>river</i> and <i>stream</i> are exchanged. Fig. 2 represents the exchanging of positions between the streams and rivers.</p> <p>Step 6: Similarly, to step 5</p> <p>The rivers move towards the sea which is the most downward place using Eq. (8).</p> <p>Step 7: The exchange of existing rivers with new picked up streams that offer the best possible value as shown in Fig. 2.</p> <p>Step 8: this step is quite similar with Fig. 2, and the swapping of the position will be exactly same as step 7, if the solution given by a river is better than the sea then their positions will also be exchanged.</p> <p>Step 9: Evaporation condition Evaporation condition will help out to stay away from getting hem in local optimum solutions. The movement of the river towards the sea can be determined by this following Pseudocode. In this pseudo code d_{max} is a small number which is close to zero. When the distance between river and sea is reached close to d_{max}, then it will be considered as that river/stream has joined the sea, and evaporation condition will be applied. A large value for d_{max} decreases the search while a small value encourages the search intensity near the sea. For that reason, d_{max} controls the search intensity near the sea. The value of d_{max} adaptively reduces as Eq. (9).</p> <p>Step 10: The raining process will take place using Eqs (10), and (11) when the evaporation condition is contented as: In Eq. (10), LB and UB are lower and upper bounds which are defined by a given problem. The μ represents the range of searching region, its value is 0.1, furthermore, in Eq. (11), $\sqrt{\mu}$ represents standard deviation, and 1 denoted the variance. By going through these steps, the created individuals with variance 1 are the best optimal solutions for the problem[6].</p> <p>Step 11: By using Eq. (10) reduce the value of d_{max}. (10).</p> <p>Step 12: In the last step, look at the convergence criteria. If the ending condition has fulfilled, then the algorithm will be stopped, if not return to Step 5.</p>	<p>$Raindrop = [X_1, X_2, X_3, \dots, X_N]$ (1)</p>  <p>$X = \text{Population of RaindropNpops}$ $\begin{bmatrix} \text{RaindropNpop1} \\ \text{RaindropNpop2} \\ \text{RaindropNpop3} \\ \dots \\ \dots \\ \text{RaindropNpop} \end{bmatrix}$ (2)</p> <p>$N_{Sr} = \text{Number of Rivers} + 1$ Here 1 is a sea (3)</p> <p>$N_{Raindrops} = N_{pop} - N_{Sr}$ (4)</p> <p>$C_i = \text{Cost}_i = f(x_1^i, x_2^i, \dots, x_{N_{var}}^i)$ $i = 1, 2, 3, \dots, N_{pop}$ (5)</p> <p>$N_{Sr} = \text{round} \left\{ \left\lfloor \frac{\text{Cost}_n}{\sum_{i=1}^{N_{Sr}} \text{Cost}_i} \right\rfloor \times N_{Raindrops} \right\}$ $n = 1, 2, 3, \dots, N_{Sr}$ (6)</p> <p>$X_{Stream}^{i+1} = X_{Stream}^i + rand \times C$ $\times (X_{River}^i - X_{Stream}^i)$ (7)</p> <p>Fig. 2. Exchange of the positions among rivers and streams.</p> <p>$X_{River}^{i+1} = X_{River}^i + rand \times C \times (-X_{Sea}^i - X_{River}^i)$ (8)</p> <p>Pseudocode For Evaporation Condition if $X_{Sea}^i - X_{River}^i < d_{max}^i = 1, 2, 3, \dots, N_{Sr} - 1 \text{end}$</p> <p>$d_{max}^{i+1} = d_{max}^i - \frac{d_{max}^i}{\text{max iteration}}$ (9)</p> <p>Raining Process</p> <p>$X_{Stream}^{new} = LB + rand \times (UB - LB)$ (10)</p> <p>And, $X_{Stream}^{new} = X_{sea} + \sqrt{\mu} \times randn(1, N_{var})$ (11)</p>
--	--

2.2 Constraint Handling and Convergence Criteria

Many studies emphasis on constraint handling strategies of metaheuristic algorithms [20]. In WCA, an improved feasible-based mechanism is introduced to control the problem specific constraints based on the following four rules [21].

Rule 1: Preference will be given to feasible solution instead of an infeasible solution.

Rule 2: A slight variation of infeasible solutions such as: (from 0.01 in the first iteration to 0.001 in the last iteration) is considered as feasible solutions.

Rule 3: Among two feasible solutions, the one holding better objective function value will be preferred.

Rule 4: Among two infeasible solutions, the one having the smaller sum of constraint violation will be preferred [22].

In metaheuristic algorithms it is also called as termination criteria, the best result is calculated when the termination condition can be considered as the maximum number of iterations, CPU time, or ϵ which is a small non-negative value and is demarcated as an acceptable tolerance between the last two results. Similarly, the WCA run until the maximum number of iterations as a convergence criterion is fulfilled [22].

3 Categories of Modified WCA

The study has divided the WCA into different categories based on the hybridization, modification, and structural and parametrically improvement of the algorithm. The following Fig. 4 describes that up to now how much percentage of studies have been conducted into these categories. For the performance enhancement of WCA, regarding its different aspects, many researchers have proposed different strategies and applied WCA into constrained problems. In this section detail of all these variants provided:

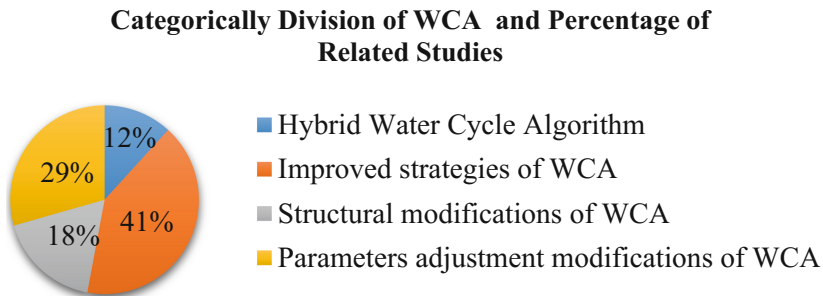


Fig. 4. Categorically division of WCA and percentage of related studies
Source: Data was collected from different online databases

‘It was a big challenge to find studies related to WCA’, owing to this, we have searched various databases such as Science-Direct, IEEE, Springer and Google-Scholar since 2012 to 2018 to overcome the thirst of WCA. As we know that the WCA was proposed in 2012 by Eskandar et al. [6], therefore in later years the importance of WCA research has been increased. Many researchers have focused on this metaheuristic with different perspectives of its modification and applications, the Fig. 5 demonstrate that how many studies have been conducted yearly and another category in the figure describes those other studies which have a close relationship to this review article.

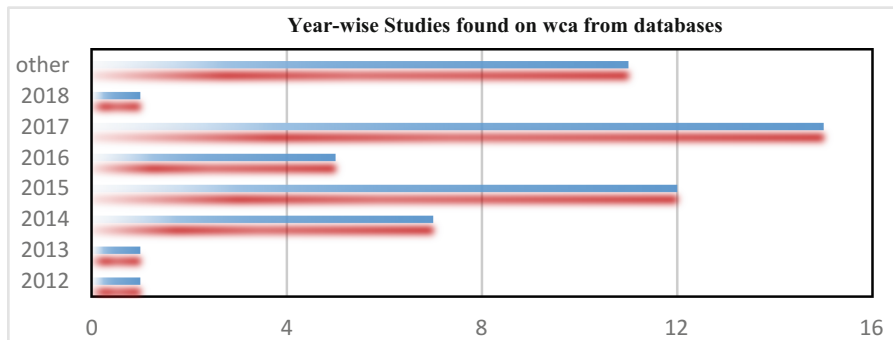


Fig. 5. Year-wise studies found on WCA from online databases

3.1 Hybrid WCA

Hybridization of algorithms have been carried out into many ways since a long time and this technique has been helped out the researchers to solve many constrained and complex engineering problems in the fields of information system analysis, artificial intelligence, decision-making systems, and data mining, etc., [23]. To improve the performance of algorithms hybridization technique applied by joining two methods. Similarly, WCA hybridization with other algorithm is a new hot trend in research, and some studies have improved its efficiency and feasibility such as:

Improved water cycle algorithm (IWCA) was proposed by Al-Saedi, [24] for attribute reduction in rough set theory(RST). The RST is considered as the primary source of attribute reduction and it is the most proficient tool for extracting useful knowledge and data. The study presented an improvement of WCA (IWCA) for rough set attribute reduction, by hybrid WCA with hill climbing algorithm to improve the exploitation process of the WCA. The results of the experiments revealed that the IWCA performs better than other methods of attribute selection.

Another hybrid technique was used by Khalilpourazari and Khalilpourazary [25] to increase the randomization and algorithm exploitation ability. WCA was hybridized with Moth-Flame Optimization (WCMFO) method, and this hybridized algorithm efficiently solved the engineering optimization problems. Based on the reviewed studies we can say that WCA has proven itself the best option for all those researchers/scientists due to its exploration and exploitation process.

3.2 Improved Strategies of WCA

Strategies are not algorithms, in the era of the common core, the evidence of change is all around us. Researchers improve the algorithm according to the nature of problems, they inject their idea and try to implement the new strategy in the existing algorithm. Furthermore, researchers have been improving different algorithms to reap the full benefits of their usability and applicability into different domains of life. Similarly, many scholars have also developed the WCA for resolving any problems related to different fields, such as:

Pahnehkolaei, et al. [26], proposed a Gradient-based WCA (GWCA) with evaporation rate. Its performance was testified by twelve well-known benchmark functions, which showed that GWCA has the ability to solve constrained problems efficiently. Dual-system WCA (DS-WCA) consists of two cycles, which are known as inner cycle and outer cycle. Therefore, it has the ability to perform exploration and exploitation process, which make it able to enhance population diversity and accelerate the convergence speed. Its performance was also compared with other famous metaheuristic techniques and it showed outstanding results regarding, speed, stability, and quality of solutions [27].

For power dispatch problems, Heidari [18] proposed a Gaussian bare-bones WCA (NGBWCA). It helps to alleviate the premature convergence and stagnation in local optima. Power dispatch has primary objectives to minimize the Resistive losses and voltage deviations. Owing to this, NGBWCA efficiency has compared on IEEE 30, 57 and 118 bus power systems. Remanufacturing rescheduling problems (RRP) can be solved by the discreet strategy provided by Gao, et al. [28] which called discrete WCA (DWCA). Authors have solved six real-life remanufacturing cases with different scales by DWCA. The obtained results of this metaheuristic specify the significance of the proposed DWCA strategy among other bi-objective algorithms.

Similarly, Qiao et al. [29], proposed an improved WCA with percolation behavior and the self-adaptive process of rainfall. It has the strong global searching ability and local optimization ability, which can effectively avoid all those deficiencies that conventional algorithms faced. Therefore, it is convenient and useful to solve multifarious optimization problems. Evaporation rate based WCA (ER-WCA), provides the better searchability than standard WCA. Because it keeps a fine balance between exploration and exploitation phases. Sadollah, et al. [5] Confirmed its results after comparing with standard WCA. Gao, et al. [30] strategically improved the WCA algorithm to solve the traffic scheduling problems, with another metaheuristic algorithm such as harmony search (HS) and Jaya algorithms.

3.2.1 Structural Modifications of WCA

Various amendments of the well-known algorithm have been proposed recently, which improve the empirical performance of the original algorithm by structural changes. Owing to this, some studies also have been conducted on the structural modification of the WCA such as:

Heidari [31] in an article contains chaotic patterns in stochastic processes of WCA to improve the performance of the conventional algorithm and to alleviate its premature convergence problem. Several chaotic signal functions, accompanied by various

chaotically improved WCA strategies, are implemented, and the best signal is preferred as the most suitable chaotic technique for modifying WCA. The statistically exposed results that chaotic WCA with a sinusoidal map and chaotic-enhanced operators can not only utilize efficient high-quality solutions but can also surpass WCA optimizer and other investigated algorithms. Guney and Basbug (2015) [10] proposed a quantized WCA (QWCA) and used it for the antenna array pattern synthesis with low side-lobe levels (SLLs) and nulls at desired directions by using four-bit digital phase shifters. Moreover, QWCA has an internal quantization mechanism and a pre-calculated array factor method. The mechanism of the internal QWCA of quantification is used to achieve digital values that correspond to the different values of the phase slider rather than simply rounding up or down after optimization. The results of the quantized QWCA revealed to achieve good SLLs and null depth (NDLs) in the composite pattern, results are achieved in an exceptionally short optimization time. Sadollah, et al. [32] possesses unique structure, sandwich panels have special features that are most important with a high strength to weight ratio. Sandwich panels with different prism cores were researching, and the compared were performed for the quest for the best design. It was implemented WCA, the figures and results obtained to infer that the diamond prism topology was most effective in weight among other things existing designs under a certain load direction.

3.3 Parameters Adjustment Modifications of WCA

Parameters adjustment approach has been vitally acknowledged technique. The linear deviations in parameters are the most common, however, some other approaches also using nonlinear or stochastic functions. Owing to this, for the performance improvement, Méndez, et al. [33] introduced fuzzy based dynamic parameters adaptation, Oscar, et al. 2017 [34], used the intuitionistic fuzzy logic for the enhancement of WCA. its performance has been compared with other well-known states of the art functions. Furthermore, to solve the loss of power supply, maintenance and operations in power management system Sarvi, and Avanaki [35] used Water cycle algorithm and fuzzy logic controller. Rezk and Fathy, [36] developed an innovative approach for taking out the optimal parameters of PV module with high-efficiency InGaP/InGaAs/Ge triple-junction solar cells (TJSCc) which is based on WCA. Single diode model represents the TJSC model and then a constrained objective function has been derived to be used in the optimization process of optimal parameter estimation.

3.4 Logically Analysis of WCA Modifications

Dynamic performance of structures plays an important role in engineering; however, there are always some circumstances in which dynamic structural performance does not meet the design requirements or actual situations in practice. Therefore, it is common that some existing structures need to be modified to acquire desired dynamic performance [37]. Many researchers have put forward many methods for the algorithm's modification. Depending on the nature of the problem and understanding/ideas of the researchers/scientists' algorithm can be modified. As stated by reviewed studies which have been conducted on WCA it is noticed that WCA cannot be merely implemented

for a specific engineering problem, although it can easily be modified according to the vibrant nature of problems. Above mentioned reviewed studies are the evidence for its applicability to real-life problems. WCA has performed successfully and efficiently in its all modified forms such as hybridization, structurally and strategically modification or parametrically modification. Furthermore, the WCA comparison has also been undertaken with all well-known metaheuristic algorithms. The obtained optimization results indicated the superiority of WCA against the existing metaheuristics. Following table-2 provides the information about the modified versions of WCA and related studies.

Table 2. Modified versions of WCA

WCA	Studies
Fuzzy based WCA	Méndez et al. [33], Sarvi and Avanaki [35]
Discrete WCA algorithm	Gao et al. [28], Guney and Basbug [10]
Dual-system WCA	Jahan [27]
Chaotic WCA	Heidari [31]
Gradient-based WCA	Pahnehkolaei et al. [26]
Evaporation rate WCA (ER-WCA)	Sadollah et al. [5]
Self-adaptive percolation behavior WCA (SPWCA)	Qiao et al. [29]
Hybrid WCAs	Al-Saedi [24], Khalilpourazari [25]

4 Applications of WCA

Owing to the WCA efficacies and applicability in real life, more and more researchers are working with it. Several studies have testified WCA application and appreciated its performance. In 2012, Eskandar, et al. [6] was proposed the water cycle algorithm Eskandar, et al. [6] proposed the original WCA in 2012 in which its performance evaluation was compared with other metaheuristics.

4.1 Application to Economics and Management Problems

Portfolio selection is considered one of the most important financial problem in the studies. Moradi et al., [38] solved the portfolio optimization problem with multi-objective water cycle algorithm, as well, for the optimal operation management Dehimi, et al. [39], presented multi-objective uniform WCA. Both proposed techniques were testified by comparing the results with multi-objective particle swarm optimization, normal constraint algorithm, and non-dominated GA-II. To deal with supply chain management (SCM) problems e.g. to minimize the supply chain cost Khalilpourazari, [40] used WCA and presented a mathematical model of such problems. This is not only used for cost minimization, but it also can guide the flow of material and number of vehicles. Moreover, various test functions were used to evaluate the efficiency of the

WCA. In the SCM, supplier selection is one of the most challenging task purchasing management. WCA, artificial bee colony algorithm, and hybrid water cycle-artificial bee colony algorithm have potential to solve the SCM related problems efficiently [41]. In this regards, literature revealed that water cycle algorithm is the best choice for the engineers and managers among the other metaheuristics.

4.2 Applications to Engineering Problems

The potential advances in the use of evolutionary algorithms and metaheuristics in engineering applications bring an opportunity and also a challenge for researchers to improve and advance in design and optimization of products, systems, and services for societal benefits [42]. Keeping in the view some researchers have been successfully implemented the WCA into the engineering fields for solving NP-hard problems.

Multiprocessor scheduling problems are difficult task to perform because this process required a couple of excellent processors for operation. Nayak, et al. [43] used WCA to deal with the multiprocessor scheduling work. His study results revealed that WCA performs better than Genetic Algorithm (GA), Bacteria Foraging Optimization (BFO) and Genetic-based Bacteria Foraging (GBF). Large-scale urban traffic light scheduling problem (LUTLSP), was efficiently handled by three metaheuristics techniques such as Jaya algorithm, harmony search and WCA [30]. Sadollah, et al. [32] stated that to find the best design and weight ratio, diamond prismatic topology is the best solution for truss and sandwich panels structure. Truss structures optimization problems considered as a most complex problem in the engineering field. A study conducted by Sadollah, Eskandar, et al. [44] revealed that nature-inspired algorithm could solve the truss structure problems proficiently than other traditional methods. They used the mine blast algorithm (MBA), improved mine blast algorithm (IMBA) and WCA to solve such problems. The proficiency of IMBA, WCA, and MBA was studied using four truss structures. Optimization results revealed that nature-inspired algorithms are more efficient in handling the engineering design and weight measurements problems.

Moreover, combinatorial nonlinear optimization problems such as Water distribution system (WDS) design optimization problems also can solve with the help of WCA. Water cycle combined with the hydraulic simulator and EPANET has applied for finding the optimal cost design of WDS. The study results have been verified by other states of the art algorithms [45].

4.3 Applications to Power and Energy System Problems

Over a decade, electricity production will be a major concern in the world, especially for Lanners in the field of electro-technology, where the increase in electricity demand has led to a more and more electricity market liberal and very Competitive. The main objective is to generate a total electrical production of all producing units at the lowest possible cost, taking into account the satisfaction of the total tax requirement and the fulfillment of the Production capacity of all units of production in energy [19]. This requires the proper design, operation, and control of the electricity produced by each

production unit. Therefore, this task can be defined as a problem with the as Economic Dispatch (ED) and listed as a constraint optimization problem.

Deihimi et al. [19], Ashouri and Hosseini [46], and Naveed et al. [47] has applied the WCA algorithm, or resolving Economic Load Dispatch (ELD) problem. The WCA optimized results are the evidence for dealing with complex ED problems. Moreover, Optimal reactive power dispatch (ORPD) and Optimal power flow (OPF) considered as most crucial tools in power system operations. Heidari, [18] recommended a Gaussian bare-bones WCA (NGBWCA), and Barzegar, et al. [48] utilized standard WCA for dealing with ORPD and OPF problems respectively.

Elhameed, and El-Fergany, [49] offered a useful methodology based on WCA for the resolution of unique and multiple objectives of the economic load dispatch (ELD) to produce the best current energy value generated for each unit. The proposed methodology based on WCA has shown in three cases of tests with different complications and under a series of objective scenarios. The numerical results and subsequently compared with other provocation optimizers indicate the viability and confirm the potency of the proposed WCA method.

Furthermore, El-Hameed, and El-Fergany, [50] solved the problem of the multi-area interconnected power system by utilizing the WCA methodology with load frequency controller. WCA efficiently generate the optimal alteration for ‘proportional–integral–derivative’ and Ziegler–Nichols PID tuning methods which have confirmed the impact of the proposed strategy. For the optimization multi-reservoir systems Yanjun, et al. [51] suggested an enhanced version of water cycle algorithm, and its results compared with standard WCA and WCA with evaporation rate. Simulation-based results demonstrated that proposed strategy could be efficiently utilized in this field. To find the optimal parameters of Power system stabilizers (PSS), it is also noted that WCA can find out the optimal parameters for PSS proficiently concerning computational time to increase the power system stability [52].

Kler, et al. [53] researched to investigate precise performance and control of photovoltaic (PV) systems. Therefore, they have applied ‘Evaporation Rate based WCA’ for useful parameters estimation of PV cell/module under varying temperature and irradiation conditions.

Hydro-thermal scheduling is an essential step in the work planning of the electrical system. It is coordinated between the performance of Hyde and the heat machine so that the cost of producing electricity is minimal under the satisfaction constraints. Due to that reason, Haroon and Malik [54] carry out research on Evaporation Rate-Based WCA for Short-Term Hydrothermal Scheduling. Their study results demonstrated the advantage of ER-WCA over the other metaheuristic algorithm.

4.4 Analysis Based on the Applications of WCA

WCA has potential to solve the constrained optimization and engineering design problems efficiently. Although this is recently proposed an algorithm, it still has successfully implemented into different domains, for example, Economics and Management, Engineering, power, and energy system. Studies carried out on the applications of WCA indicated that WCA has been solved the complex nonlinear problems such as water distribution, portfolio optimization, economic load dispatch problems, electric

power system, truss structure design and supply chain management problems very efficiently and effectively. Based on the reviewed studies results have been compared with other metaheuristics which have authenticated and testified the WCA. Following are the examples of WCA success and efficiency:

1. To find the best pipe diameter sizes for minimizing the construction cost. Comparison with other reported algorithms has been carried out regarding the optimized statistical value and computational efforts (i.e., number of function evaluations). Based on the obtained optimization results, the WCA offered cheaper design (i.e., configuration of pipe diameters) compared with other optimizers (\$295,000 saving money) having faster convergence rate [45].
2. Dihem, et al. [19], Ashouri and Hosseini [46] studies are related to ED problem considering practical constraints of generating units. The effectiveness of the ‘WCA’ has been reconnoitered on three test systems with different convolution and scales, compared to the objective function, the cost of fuel has both types of curves smooth and non-smooth respectively. Substantiated the results of simulations for complex ED problems that present the best properties of solutions in comparison with another set of rules.
3. The study related to job-shop scheduling problem analysis using the WCA technique in the universal optimization of flexible job-shop problem exhibited that the proposed algorithm in cost function was able to get hold of the best solution in the problem than the other different approaches in the literature [27].
4. Gao, et al. [30] solved a large-scale urban traffic light scheduling problem (LUTLSP). The evaluations and dialogs verify that the WCA methods can successfully explain the LUTLSP significantly superior the existing techniques.

So far the majority of the studies have been conducted in the field of power and energy system, which are the evidence of WCA superiority. Although many studies also held in the area of Engineering, Economics, and Management there is still a dire need to do more. Following given table-3 describes those studies which have been carried out on specific application areas of WCA.

Table 3. Application areas of WCA

Area	Studies
Economics and management	Moradi et al. [38], Deihimi et al. [39], Khalilpourazari [40], Praepanichawat [41]
Engineering problems	Nayak et al. [43], Gao et al. [30], Sadollah et al. [32], Sadollah [45], Sadollah and Eskandar et al. [44]
Power and energy system problems	Dihem et al. [19], Barzegar et al. [48], Elhameed and El-Fergany, [49], El-Hameed, and El-Fergany [50], Yanjun and Yadong et al. [51]

5 Conclusions and Discussion

About the reviewed articles, it will be appreciated that WCA has been mostly applied in solving massive optimization problems. After reviewing the studies which have been resolved the economics and management problems it is noted that WCA has not only performed better than the traditional approaches, it has also provided the best and efficient results.

In General, a fundamental feature of meta-heuristics is exploitation (intensification) and exploration (diversification). Exploitation to keep track of information from the best current solutions, looking for the end of the current solution and choose the best candidate. While the exploration characteristic ensures that you further explore the search space, as it is can often diversify with several random strategies that are essential for an algorithm to jump out of some local Optima. WCA is an easy to implement and easy to modify according to the nature of the problem. WCA has a comparative advantage over traditional metaheuristic algorithms, as another metaheuristic algorithm quickly falls into their local optimum solution such as PSO. WCA avoids getting trapped in the local optimum solutions due to its evaporation condition and raining process. WCA convergence speed has been more improved by a new self-adaptive WCA with percolation behavior approach. Simultaneously, a self-adaptive rainfall process can generate the new stream, more and more new position can be explored, consequently, increasing the diversity of the population. The improved WCA can efficiently utilize to various multifarious optimization tasks in diverse areas of sciences and technologies. The MOWCA was used to solve some distinguished benchmark and engineering Mops. The efficiency and performance of the MOWCA were demonstrated using three standard criteria (i.e., generational distance, metric of spacing, and spread metric). The MOWCA offers competitive solutions compared with other population-based algorithms.

The article summarizes the review of ‘WCA’ studies, as mentioned above in Tables 2 and 3, where ‘WCA’ publications on various areas of application, modification and hybridization to various formulations of combined optimization problems have been listed. Based on these tables, we can see that the growth of this algorithm develops rapidly, despite the fact that its proposition is about five years. So far, most researchers have focused on solving energy and energy problems with ‘WCA’, and it provides very superior results among others metaheuristic algorithms. According to the articles examined, it can be analyzed that the literature still has a thirst for studies focusing on WCA applications and modifications in several disciplines. Many improved WCA tackles the constrained problems very efficiently such as:

1. The ER-WCA shows its potential for tackling constrained problems by providing better statistical optimization results in a fewer number of function evaluations compared with the considered optimizers. It can the global minimum of multimodal functions with minimum possibility of getting trapped in local minima.
2. NGBWCA can be applied to other global optimization problems, especially engineering optimization problems.
3. The Multi-objective version of WCA (MOWCA) for solving unconstrained and constrained multi-objective optimization problems.

4. Hybrid water cycle-artificial bee colony algorithm (HWAA) introduced to find the optimal solutions for optimal order allocation problem.
5. The self-adaptive process of rainfall can generate the new stream, more and more new position can be explored, consequently, increasing the variety of population.

In future, some studies should be conducted on

1. Hybridizing of PSO and WCA, GA or DE and WCA, and ACO and WCA. Such studies can provide much stronger evidence for solving linear or non-linear problems. It is expected that new technique obtains high-quality solutions in tackling constrained and unconstrained problems and some applications, for example, training of neural networks (NNs).
2. Hence, researchers should still focus on this algorithm for the implementation in other fields to solve the complex problems.
3. The proper utilization of 'WCA' can be more helpful in all fields of engineering and optimization. The researchers should try to implement it for various complex optimization problems.
4. Further investigation of this algorithm by its adaptation to other domain, which adjustment of control parameters and theoretical studies should be explored in the next future. Overall literature shows that not much work was done on the theoretical aspects of the WCA, it would be interesting to perform a theoretical study of runtime and convergence nature of this algorithm.

As a final point, this study conducted an efficient, far-reaching review to get hold of the relevant literature on the applications, modifications, and hybridizations of the WCA when applied to solve problems of high dimensionality in the different domain. This comprehensive review will be advantageous for the researchers, literature, and community. In addition to for those are working or want to research this domain. In conclusion, there are still many exciting and innovative research directions where WCA deployment can be helpful for optimization problems.

Acknowledgments. The work described in this paper was supported by grants from The Natural Science Foundation of China (Grant No. 71571120, 71271140); Project of Guangdong Province Universities and Colleges Pearl River Scholar Funded Scheme 2016; China Postdoctoral Science Foundation (Grant No. 2016M602528).

Conflict of Interests. The authors declare that there is no conflict of interests regarding the publishing of this paper.

References

1. Holland, J.H.: Genetic algorithms. *Sci. Am.* **267**(1), 66–73 (1992)
2. Ingber, L.: Simulated annealing: practice versus theory. *Math. Comput. Model.* **18**(11), 29–57 (1993)
3. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39–43. IEEE Press, New York (1995)

4. Dorigo, M., Di Caro, G.: Ant colony optimization: a new meta-heuristic in evolutionary computation. In: Proceedings of the 1999 Congress on Evolutionary Computation, pp. 1470–1477. IEEE Press, New York (2002)
5. Sadollah, A., et al.: WCA with evaporation rate for solving constrained and unconstrained optimization problems. *Appl. Soft Comput.* **30**, 58–71 (2015)
6. Eskandar, H., et al.: WCA—A novel metaheuristic optimization method for solving constrained engineering optimization problems. *Comput. Struct.* **110**, 151–166 (2012)
7. Haddad, O.B., Moravej, M., Loáiciga, H.A.: Application of the WCA to the optimal operation of reservoir systems. *J. Irrig. Drain. Eng.* **141**(5), 0401–4064 (2014)
8. Lenin, K., Reddy, B.R., Kalavathi, M.S.: WCA for solving optimal reactive power dispatch problem. *J. Eng. Technol. Res.* **2**(2), 1–11 (2014)
9. Jabbar, A., Zainudin, S.: WCA for attribute reduction problems in rough set theory. *J. Theor. Appl. Inf. Technol.* **61**(1), 107–117 (2014)
10. Guney, K., Basbug, S.: A quantized water cycle optimization algorithm for antenna array synthesis by using digital phase shifters. *Int. J. RF Microw. Comput.-Aided Eng.* **25**(1), 21–29 (2015)
11. Sadollah, A., et al.: WCA for solving multi-objective optimization problems. *Soft. Comput.* **19**(9), 2587–2603 (2015)
12. Sadollah, A., Eskandar, H., Kim, J.H.: WCA for solving constrained multi-objective optimization problems. *Appl. Soft Comput.* **27**, 279–298 (2015)
13. Zhu, H., et al.: Particle swarm optimization (PSO) for the constrained portfolio optimization problem. *Expert Syst. Appl.* **38**(8), 10161–10169 (2011)
14. Lee, K.S., Geem, Z.W.: A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. *Comput. Methods Appl. Mech. Eng.* **194** (36–38), 3902–3933 (2005)
15. Nesmachnow, S.: An overview of metaheuristics: accurate and efficient methods for optimization. *Int. J. Metaheuristics* **3**(4), 320–347 (2014)
16. Jordehi, A.R.: A chaotic artificial immune system optimization algorithm for solving global continuous optimization problems. *Neural Comput. Appl.* **26**(4), 827–833 (2015)
17. David, S.: *The Water Cycle*, Illustrations by John Yates. Thomson Learning, New York (1993)
18. Heidari, A.A., Abbaspour, R.A., Jordehi, A.R.: Gaussian bare-bones WCA for optimal reactive power dis-patch in electrical power systems. *Appl. Soft Comput.* **57**, 657–671 (2017)
19. Deihimi, A., et al.: Solving smooth and non-smooth economic dispatch using WCA. In: 2017 The 5th International Conference on Electrical Engineering - (ICEE-B), Bahria University Islamabad Campus, Boumerdes, Algeria, pp. 29–31. IEEE (2017)
20. Hu, Z., Wang, X., Taylor, G.: Stochastic optimal reactive power dispatch: formulation and solution method. *Int. J. Electr. Power Energy Syst.* **32**(6), 615–621 (2010)
21. Mezura-Montes, E., Coello, C.A.C.: An empirical study about the usefulness of evolution strategies to solve constrained optimization problems. *Int. J. Gen Syst* **37**(4), 443–473 (2008)
22. Kaveh, A., Talatahari, S.: A particle swarm ant colony optimization for truss structures with discrete variables. *J. Constr. Steel Res.* **65**(8), 1558–1568 (2009)
23. Wang, C., Ou, F.: An attribute reduction algorithm in rough set theory based on information entropy. In: 2008 International Symposium on Computational Intelligence and Design. ISCID 2008, pp. 3–6. IEEE (2008)
24. Al-Saedi, A.S.J.: Hybrid water cycle algorithm for attribute reduction problems. In: Proceedings of the World Congress on Engineering and Computer Science (WCECS), San Francisco, USA, vol. I (2015)

25. Khalilpourazari, S., Khalilpourazary, S.: An efficient hybrid algorithm based on water cycle and moth-flame optimization algorithms for solving numerical and constrained engineering optimization problems. *Soft Comput.* **21**(20), 1–24 (2017)
26. Pahnkolaei, S.M.A., et al.: Gradient-based WCA with evaporation rate applied to chaos suppression. *Appl. Soft Comput.* **53**, 420–440 (2017)
27. Jahan, M.V., Dashtaki, M., Dashtaki, M.: WCA improvement for solving job shop scheduling problem. In: 2015 International Congress on Technology, Communication and Knowledge, pp. 576–581. IEEE Press, New York (2015)
28. Gao, K., Duan, P., Su, R., Li, J.: Bi-objective water cycle algorithm for solving remanufacturing rescheduling problem. In: Shi, Y., et al. (eds.) SEAL 2017. LNCS, vol. 10593, pp. 671–683. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68759-9_54
29. Qiao, S., Zhou, Y., Wang, R., Zhou, Y.: Self-adaptive percolation behavior water cycle algorithm. In: Huang, D.-S., Bevilacqua, V., Prashan, P. (eds.) ICIC 2015. LNCS, vol. 9225, pp. 85–96. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22180-9_9
30. Gao, K., et al.: Jaya, harmony search and WCAs for solving large-scale real-life urban traffic light scheduling problem. *Swarm Evol. Comput.* **37**, 58–72 (2017)
31. Heidari, A.A., Abbaspour, R.A., Jordehi, A.R.: An efficient chaotic WCA for optimization tasks. *Neural Comput. Appl.* **28**(1), 57–85 (2017)
32. Sadollah, A., et al.: Sizing optimization of sandwich panels having prismatic core using WCA. In: 2013 Fourth Global Congress on Intelligent Systems, pp. 325–328. IEEE (2013)
33. Méndez, E., Castillo, O., Soria, J., Melin, P., Sadollah, A.: Water cycle algorithm with fuzzy logic for dynamic adaptation of parameters. In: Sidorov, G., Herrera-Alcántara, O. (eds.) MICAI 2016. LNCS (LNAI), vol. 10061, pp. 250–260. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-62434-1_21
34. Oscar, C., Eduardo, R., Olympia, R.: WCA augmentation with fuzzy and intuitionistic fuzzy dynamic adaptation of parameters. *Notes Intuitionistic Fuzzy Sets* **23**(1), 79–94 (2017)
35. Sarvi, M., Avanaki, I.N.: An optimized fuzzy logic controller by WCA for power management of stand-alone hybrid green power generation. *Energy Convers. Manag.* **106**, 118–126 (2015)
36. Rezk, H., Fathy, A.: A novel optimal parameters identification of triple-junction solar cell based on a recently meta-heuristic WCA. *Sol. Energy* **157**, 778–791 (2017)
37. Liu, Z., Li, W., Ouyang, H.: Structural modifications for torsional vibration control of shafting systems based on torsional receptances. *Shock. Vib.* **9** (2016). <https://doi.org/10.1155/2016/2403426>
38. Moradi, M., et al.: The application of WCA to portfolio selection. *Econ. Res.-Ekon. Istraživanja* **30**(1), 1277–1299 (2017)
39. Deihimi, A., Zahed, B.K., Iravani, R.: An interactive operation management of a micro-grid with multiple distributed generations using multi-objective uniform WCA. *Energy* **106**, 482–509 (2016)
40. Khalilpourazari, S., Mohammadi, M.: Optimization of closed-loop supply chain network design: a WCA approach. In: 2016 12th International Conference on Industrial Engineering (ICIE), pp. 41–45. IEEE (2016)
41. Praepanichawat, C., Khompatraporn, C., Jaturanonda, C., Chotyakul, C.: Water cycle and artificial bee colony based algorithms for optimal order allocation problem with mixed quantity discount scheme. In: Gen, M., Kim, Kuinam J., Huang, X., Hiroshi, Y. (eds.) Industrial Engineering, Management Science and Applications 2015. LNEE, vol. 349, pp. 229–239. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-47200-2_26
42. Magalhães-Mendes, J., Greiner, D.: Evolutionary algorithms and metaheuristics in civil engineering and construction management. Springer, Heidelberg (2015). <https://doi.org/10.1007/978-3-319-20406-2>

43. Nayak, S.K., Panda, C.S., Padhy, S.K.: Efficient multiprocessor scheduling using water cycle algorithm. In: Ray, K., Pant, M., Bandyopadhyay, A. (eds.) *Soft Computing Applications*. SCI, vol. 761, pp. 131–147. Springer, Singapore (2018). https://doi.org/10.1007/978-981-10-8049-4_7
44. Sadollah, A., et al.: Water cycle, mine blast and improved mine blast algorithms for discrete sizing optimization of truss structures. *Comput. Struct.* **149**, 1–16 (2015)
45. Sadollah, A., et al.: Application of WCA for optimal cost design of water distribution systems. In: *11th International Conference on Hydroinformatics*. CUNY Academic Works (2014)
46. Ashouri, M., Hosseini, S.M.: Application of krill herd and WCAs on dynamic economic load dispatch problem. *Int. J. Inf. Eng. Electron. Bus.* **6**(4), 12 (2014)
47. Naveed, S., Haroon, S.S., Khan, N.A.: Solving non-convex economic dispatch using WCA. *NED Univ. J. Res.* **13**(2), 31 (2016)
48. Barzegar, A., et al.: Optimal power flow solution using WCA. In: *14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 1–4. IEEE (2016)
49. Elhameed, M., El-Fergany, A.: WCA-based economic dispatcher for sequential and simultaneous objectives including practical constraints. *Appl. Soft Comput.* **58**, 145–154 (2017)
50. El-Hameed, M.A., El-Fergany, A.A.: WCA-based load frequency controller for interconnected power systems comprising non-linearity IET generation. *Transm. Distrib.* **10**(15), 3950–3961 (2016)
51. Yanjun, K., et al.: An enhanced WCA for optimization of multi-reservoir systems. In: *16th International Conference on Computer and Information Science*, pp. 379–386. IEEE Press, New York (2017)
52. Ghaffarzadeh, N.: WCA based power system stabilizer robust design for power systems. *J. Electr. Eng.* **66**(2), 91–96 (2015)
53. Kler, D., et al.: PV cell and module efficient parameters estimation using evaporation rate based WCA. *Swarm Evol. Comput.* **35**, 93–110 (2017)
54. Haroon, S.S., Malik, T.N.: Evaporation rate-based WCA for short-term hydrothermal scheduling. *Arab. J. Sci. Eng.* **42**(7), 2615–2630 (2017)



A Bias Neural Network Based on Knowledge Distillation

Yulong Wang^{1(✉)}, Zhi Wu¹, and Yifeng Huang²

¹ State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, Beijing, China
wyl@bupt.edu.cn

² North China Electric Power University, Beijing, China

Abstract. In the field of deep learning and image recognition, to improve the accuracy of recognition, the neural model with a complex structure is usually selected as the training model. However, the model with a complex structure has the disadvantages of a large amount of calculation and time-consuming, which limits the ability of deep CNN to deploy on resource-limited devices like mobile phones. This paper presented a new logo recognition approach that is based on knowledge distillation, improving the recognition accuracy of a small model by knowledge transfer. At the same time, a bias neural network is introduced to increase the recognition accuracy of the target class. In this paper, we select ResNet-50 as the cumbersome network, ResNet-18 and VGG16 as small networks respectively. With only knowledge distillation, the average recognition accuracy of ResNet-18 and VGG16 have increased by 8% and 11% respectively. With the proposed bias neural network, the recognition accuracy of ResNet-18 and VGG16 further increased by 2%–10%. The recognition accuracy of the target class is within 5% of that of ResNet-50, which means the bias neural network with fewer layers and parameters is able to reach nearly the same recognition performance as the cumbersome network on target logo classes. And the experiments validate that the bias neural network can improve the accuracy of bias classes.

Keywords: Deep learning · Image recognition · Knowledge distillation
Bias neural network

1 Introduction

In recent years, the rise of e-commerce companies such as Alibaba and Amazon has made the brand logo an important element of the electronic economy market. Although the current research on image classification and recognition technology is relatively mature in technology, the identification of brand logos is still a challenge because of the variability of usage scenarios, the differences in the data samples, and the limited ability of deep CNN to deploy on resource-limited devices like mobile phones.

Single logo recognition is a special type of image classification task, which is becoming an urgent requirement for short-term web-based services [10]. But single logo recognition is still a difficult task. First of all, the samples in available training datasets usually contain a lot of noise. For example, in FlickrLogos-32 dataset [11],

there are more than 7,000 Logo sample data, only 810 images have a little noise, and the rest of the images have lots of interference and noise. To counter the negative impact of noise, we usually need a large neural network. Secondly, due to the development of modern mobile devices, the need for nesting recognition programs on smartphones with limited computation and storage resources is also increasing. Although conventional image recognition algorithms such as SIFT (Scale-invariant feature transform) [1, 2] and HOG (Histogram of Oriented Gradient) [3–5] can run smoothly in smartphones, they are sensitive to noises, thus sometimes have poor recognition performance in reality. CNN-based image recognition solutions [7] can learn abstract features from noisy samples. But the size of the models is usually too large to run in smartphones. Therefore, building a small-scale, high-performance convolutional neural model suitable for a resource-limited terminal is a problem worthy of study.

Generally, in order to improve the accuracy in the recognition task, a model with a complex structure is used, or different models are selected to be trained on the same dataset, and then their results are averaged as a final prediction value. However, such methods require a large amount of calculation. It is time-consuming and not suitable for the logo recognition under the resource limitation scenario. Hinton et al. [6] proposed the idea of knowledge distillation. They mainly expound a new idea of training neural models, namely selecting a model with a complex structure and high recognition accuracy as a cumbersome model, and then selecting the small model with low recognition accuracy. Then the two models are trained simultaneously on the same dataset, and the predicted value of the cumbersome model is transfer as knowledge to the small model so that the recognition accuracy of the small model is increased. We improved the knowledge distillation algorithm by optimizing the loss function of the cumbersome model.

Apart from a small model, the task of single logo recognition also needs a high accuracy on the target logo. Malls need to frequently carry out brand promotions and discounts, such as during the women's day, the cosmetics brand needs a lower misidentification rate as it is a hot season for cosmetics. During the Children's Day, snacks and toy brands are hot season products, and it's the peak period of such brand purchases. Therefore, such brand's logos should have a higher recognition rate than that of the daily routine, and it is more convenient for the customer to purchase the corresponding brand. Therefore, on the constraint of not expanding the structure of the original model, brand logos in different promotion periods need to have higher accuracy than daily, so as to increase the purchasing willingness of customers. Based on this requirement, after improving the small model as described above, the bias neural network has the ability to be customized to prefer the target logo.

The key contributions of our work are 1. It (Bias neural networks) is the first small neural network for single logo recognition that has nearly the same accuracy performance as a large neural network; 2. We carry out experiments to validate the effectiveness of the proposed bias neural network with a public available dataset.

The rest of paper is organized as follows. Section 2 overviews the existing works on the problem of knowledge distilling and image recognition. Section 3 presented the details of our approach. In Sect. 4 we carry out two experiments to evaluate the effectiveness of the proposed approach. Finally, Sect. 5 concludes the whole paper.

2 Related Works

Our work is based on the existing research works on neural network compression. Bucilua et al. [26] have shown that it is possible to compress the knowledge in a cumbersome model into a small model so that a deep cumbersome model can be simplified. They firstly demonstrate convincingly that the knowledge acquired by a large ensemble of models can be transferred to a single small model, which will be much easier to be deployed in devices with limited computation resource. Hinton et al. [6] develops Caruana’s approach further using a different compression technique that distills the knowledge of an ensemble of models into a single model. They mainly expound a new idea of training neural models, namely selecting a model with a complex structure and high recognition accuracy as a cumbersome model, and then selecting the small model with low recognition accuracy. Based on Geoffrey Hinton’s work, we improved the knowledge distilling algorithm by adding bias for the target log. And we achieve promising results on the FlickrLogos-32 dataset.

Various researches had been done for image classification, some of which are suitable for logo recognition. He et al. [27] have provided convincing theoretical and practical evidence for the advantages of utilizing a residual method for image recognition using ResNet-18, especially for object detection. The VGG network was introduced by Simonyan [28], which achieved very good performance at a relatively low computational cost. Based on their works, we use the biased distilled algorithm to improve the two networks.

3 Our Approach

3.1 Rectified Knowledge Distilling

In a typical neural network, the softmax layer produces the probabilities of each class according to Eq. 1

$$q_i = \frac{\exp(z_i)}{\sum_j \exp(z_j)} \quad (1)$$

Hinton et al. [6] used the concept of temperature [6] to weaken the difference between different classes, as shown in Eq. 2

$$q_i = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)} \quad (2)$$

Assume the gradient of cross entropy loss function of the soft target [6] is dc/ds_i , in which c is the cross-entropy loss function of soft target, S_i is the i -th component of the small model’s logit. Assume q_{1i} is the i -th component of soft target, q_{2i} is the i -th component of the small model’s softmax output. Assume t_i is the i -th component of the cumbersome model’s logit. So that the gradient is given by:

$$\frac{\partial C}{\partial s_i} = \frac{1}{T}(q_{2i} - q_{1i}) \tag{3}$$

From (1) (2) and (3) the gradient equals to

$$\frac{\partial C}{\partial s_i} = \frac{1}{T} \left(\frac{\exp(s_i/T)}{\sum_j \exp(s_j/T)} - \frac{\exp(t_i/T)}{\sum_j \exp(t_j/T)} \right) \tag{4}$$

The temperature is much greater compared with the magnitude of the logits, for the basic equivalent infinitesimal replacement, we can make an approximation

$$\exp(s_i/T) \approx 1 + s_i/T \tag{5}$$

From (4) and (5) we can approximate the gradient to

$$\frac{\partial C}{\partial s_i} \approx \frac{1}{T} \left(\frac{1 + s_i/T}{N + \sum_j s_j/T} - \frac{1 + t_i/T}{N + \sum_j t_j/T} \right) \tag{6}$$

in which, N is the number of classes. When using FlickrLogos-32 dataset for training, the parameter N is set to 32. Since N is much larger than $\sum_j s_j/T$ and $\sum_j t_j/T$, we can assume that the logits have been zero-meaned, which means:

$$\sum_j s_j/T = 0, \sum_j t_j/T = 0 \tag{7}$$

From (6) and (7) the gradient finally can be simplified to

$$\frac{\partial C}{\partial s_i} \approx \frac{1}{N \times T^2} (s_i - t_i) \tag{8}$$

So from (8) we can obtain the loss function by calculating the following gradient

$$C_k = \frac{1}{2N \times T^2} \sum_i (s_i - t_i)^2 \tag{9}$$

3.2 Loss Function for Small Model

We attempt to transfer the knowledge from a cumbersome model to a small model, so as to ensure the small model have a better performance. We use data labels to correct the value of loss function C_k . Assume C_h is the cross-entropy loss function of a data label, assume the i-th component of a data label is y_i . C_h is defined by:

$$C_h = -\frac{1}{n} \sum_j (y_i \ln q_{2i} + (1 - y_i) \ln(1 - q_{2i})) \tag{10}$$

Then we use C_h to combine with loss function C_k to obtain a new loss function C_w that reflects the correction by the data label, as shown in Eq. 11.

$$C_w = \alpha C_k + (1 - \alpha) C_h \quad (11)$$

in which, α is the factor that represents the data label's correcting effect in the training of a small model. If a low α means the data label's error correction dominates in the training process of the small model, otherwise the knowledge transformation dominates. In this paper, α is set to the value between 0.8 and 1 to carry out the experiments.

3.3 Bias Neural Network

Bias neural network, compared to the normal small model can have a much higher accuracy on some certain classes, and have similar recognition ability with the cumbersome model on target classes. Assume W is the bias vector and w_i is the i -th component of W . We can use the bias vector to bias the small model, which is defined by:

$$W = \langle w_1, w_2, \dots, w_N \rangle^T \quad (12)$$

Assume O_s is a one-hot encoding data label that denoted as

$$O_s = \langle y_1, y_2, \dots, y_n \rangle \quad (13)$$

From (12) and (13) we can obtain the dot product of bias vector and the data label whose result is

$$K = W \cdot O_s = w_1 y_1 + w_2 y_2 + \dots + w_n y_n \quad (14)$$

In order to increase the accuracy for classifying the target class, we can increase the corresponding value of W so that K will reflect the training preference for the target class. With K we can modify loss function C_w as follows:

$$C_w = K(\alpha C_k + (1 - \alpha) C_h) \quad (15)$$

It can be seen that when the target class is classified incorrectly, C_w will have a higher loss. So in the back-propagation process, the correction degree on the target class will be higher than that of other classes, thus the accuracy for recognizing the target class will be increased.

4 Experiment

We carried out two experiments and compared the results with ResNet-18 [27] and VGG16 [28]. We choose ResNet-50 as the cumbersome model in both experiments. We choose VGG16 and ResNet-18 as the small models in the first and second experiments respectively. After knowledge distillation, the average accuracy of VGG16 is higher than that of the original VGG16. Through a biased knowledge distillation, the

Table 1. Comparison of accuracy on VGG16

	Res- Net50	VGG16	Distilled VGG16	Ran- domly select 1 target class	Ran- domly select 2 target class	Ran- domly select 3 target class
Adidas	0.62	0.11	0.31	0.30	0.29	0.29
Aldi	0.90	0.50	0.80	0.75	0.78	0.74
Apple	0.93	0.62	0.68	0.63	0.64	0.72
Becks	0.78	0.72	0.76	0.78	0.77	0.71
Bmw	0.82	0.47	0.64	0.62	0.60	0.56
Carlsberg	0.90	0.50	0.80	0.83	0.89	0.88
Chimay	0.81	0.63	0.70	0.68	0.70	0.69
Cocacola	0.76	0.58	0.58	0.69	0.56	0.61
Corona	0.75	0.64	0.66	0.59	0.58	0.59
Dhl	0.87	0.43	0.83	0.82	0.81	0.86
Erdinger	0.66	0.38	0.38	0.37	0.39	0.36
Esso	0.80	0.60	0.73	0.75	0.72	0.74
Fedex	0.90	0.36	0.85	0.89	0.89	0.82
Ferrari	0.92	0.88	0.88	0.84	0.84	0.85
Ford	0.85	0.58	0.78	0.7	0.75	0.72
Fosters	0.85	0.74	0.85	0.81	0.70	0.73
Google	0.86	0.73	0.78	0.74	0.75	0.80
Guinness	0.86	0.53	0.76	0.56	0.84	0.61
Heineken	0.90	0.71	0.74	0.70	0.61	0.66
HP	0.73	0.53	0.63	0.55	0.54	0.57
Milka	0.69	0.51	0.63	0.59	0.57	0.58
Nvidia	0.54	0.47	0.51	0.49	0.47	0.47
Paulaner	0.76	0.15	0.63	0.62	0.60	0.54
Pepsi	0.46	0.43	0.45	0.45	0.40	0.39
Rittersport	0.95	0.38	0.85	0.84	0.81	0.78
Shell	0.62	0.56	0.60	0.55	0.58	0.57
Singha	0.90	0.45	0.87	0.85	0.86	0.85
Starbucks	0.84	0.69	0.73	0.51	0.59	0.41
Stellaartois	0.90	0.54	0.73	0.60	0.80	0.65
Texaco	0.99	0.25	0.62	0.60	0.56	0.57
Tsingtao	0.71	0.58	0.65	0.63	0.59	0.59
Ups	0.66	0.33	0.46	0.41	0.45	0.45
accuracy	0.79	0.61	0.72	0.68	0.70	0.68

bias neural network of VGG16 has obtained a higher recognition accuracy on the target class than that of the knowledge distillation network and the original network. Similarly, ResNet-18 has the same effect on bias neural network as VGG16.

4.1 VGG16, Distilled VGG16 and Biased Distilled VGG16

The first experiment compares the average recognition accuracy and each classes' accuracy of VGG16, distilled VGG16 and biased distilled VGG16. The result is shown in Table 1.

Comparing the average recognition accuracy of the original VGG16 and distilled VGG16, we can find that the average recognition accuracy increased from 0.61 to 0.72. Although after adding the bias the average recognition accuracy decreased 2%–4%, the recognition accuracy of the target classes increased 2%–11%. For example, the target class Stellaarfois's recognition accuracy increased from 0.73 to 0.80 and the class Guinness's recognition accuracy increased from 0.76 to 0.84. And we find that the recognition accuracy of those target classes can have the same recognition accuracy compared with the cumbersome model ResNet-50. For example, the class Guinness's accuracy on ResNet-50 is 0.86 and its accuracy on biased distilled VGG16 is 0.84. Therefore, the bias neural model can have the similar recognition ability on the target classes compare with the cumbersome model.

Besides that we found that when select 1 target the bias class's recognition accuracy increased 9% (Cocacola), when select 2 targets the bias classes' recognition accuracy increased 8% and 7% (Guinness and Stellaarfois) and when select 3 targets the bias classes' recognition accuracy increased 4%, 3% and 2% (Apple, Dhl, and Google). So that we guess that the with the increase in the number of bias classes the rise of the bias classes' recognition accuracy will decrease. And we will validate it in the future work. After using the bias method the recognition accuracy of the other classes decrease. For example, Milka's recognition accuracy decreased 4%, 5%, and 6% when randomly select 1 target, 2 targets, and 3 targets.

4.2 ResNet-18, Distilled ResNet-18 and Biased Distilled ResNet-18

Distilled ResNet-18 is the small model that transferred from the cumbersome model ResNet-50. In the second experiment, we train the cumbersome model ResNet-50 then transferred it to two ResNet-18 models, of which one uses bias and the other does not. The result is shown in Table 2.

Comparing the accuracy of the distilled ResNet-18 on each class, it can be seen that after introducing the bias, the accuracy of each target class on ResNet-18 is higher than that of the previous ones, and the accuracy rises in the fluctuation range of 2% to 9%. The average accuracy of the bias model is slightly lower than before when no bias is introduced. The average accuracy fluctuates between 70% and 73%, compared to the previous average accuracy of 75%, with a drop of 2% to 5%. For non-target classes, the recognition accuracy of each type decreases slightly after introducing the bias. However, the result shows that the recognition accuracy of the target class on distilled ResNet-18 is similar to that of the class on ResNet-50 after using the bias. For example, the recognition accuracy of Carlsberg on a biased distilled ResNet-18 is 88%, which

Table 2. Comparison of accuracy on ResNet-18

	ResNet-50	Res-Net-18	Dis-tilled Res-Net-18	Ran-domly select 1 target class	Ran-domly select 2 target class	Ran-domly select 3 target class
Adidas	0.62	0.50	0.55	0.55	0.53	0.55
Aldi	0.90	0.73	0.80	0.78	0.78	0.80
Apple	0.93	0.84	0.84	0.83	0.81	0.79
Becks	0.78	0.79	0.77	0.78	0.77	0.71
Bmw	0.82	0.53	0.64	0.64	0.60	0.56
Carlsberg	0.90	0.74	0.79	0.79	0.88	0.77
Chimay	0.81	0.68	0.67	0.67	0.63	0.72
Cocacola	0.76	0.60	0.66	0.66	0.65	0.61
Corona	0.75	0.57	0.63	0.64	0.64	0.63
Dhl	0.87	0.71	0.82	0.80	0.81	0.86
Erdinger	0.66	0.58	0.61	0.61	0.61	0.62
Esso	0.80	0.77	0.79	0.78	0.79	0.66
Fedex	0.90	0.78	0.86	0.87	0.86	0.82
Ferrari	0.92	0.81	0.82	0.82	0.82	0.87
Ford	0.85	0.72	0.73	0.79	0.72	0.72
Fosters	0.85	0.73	0.70	0.70	0.70	0.68
Google	0.86	0.76	0.86	0.86	0.83	0.84
Guinness	0.86	0.83	0.85	0.85	0.84	0.71
Heineken	0.90	0.87	0.88	0.89	0.85	0.86
HP	0.73	0.68	0.71	0.67	0.69	0.67
Milka	0.69	0.62	0.66	0.63	0.56	0.68
Nvidia	0.54	0.57	0.54	0.45	0.48	0.36
Paulaner	0.76	0.71	0.70	0.70	0.70	0.69
Pepsi	0.46	0.49	0.47	0.47	0.47	0.44
Rittersport	0.95	0.88	0.93	0.91	0.88	0.78
Shell	0.62	0.53	0.60	0.59	0.60	0.61
Singha	0.90	0.84	0.87	0.87	0.89	0.85
Starbucks	0.84	0.82	0.84	0.82	0.84	0.81
Stellaartois	0.90	0.79	0.83	0.82	0.80	0.85
Texaco	0.99	0.93	0.96	0.91	0.93	0.86
Tsingtao	0.71	0.70	0.70	0.67	0.69	0.67
Ups	0.66	0.76	0.74	0.75	0.74	0.75
accuracy	0.79	0.67	0.75	0.70	0.73	0.71

has a 90% recognition accuracy on the cumbersome network. Besides, it has the same effect on other target classes such as Dhl, Ferrari, Chimay, Ford, Milka and Singha.

Besides that we found that when select 1 target the bias class's recognition accuracy increased 6% (Ford), when select 2 targets the bias classes' recognition accuracy increased 9% and 2% (Carlsberg and Singha) and when select 3 targets the bias classes' recognition accuracy increased 5%, 4% and 5% (Chimay, Dhl, and Ferrari). And after using the bias method the recognition accuracy of the other classes decrease. For example, the Apple's recognition accuracy decreased 4%, 5%, and 6% when randomly select 1 target, 2 targets, and 3 targets. And Cocacola's recognition accuracy decreased 4% and 5% when randomly 2 targets and 3 targets.

With the two experiments, we find that the number of the bias classes will affect the recognition accuracy. And this part is not discussed in detail in this paper in the future work we will validate it.

5 Conclusion

In this paper, the knowledge gained from the cumbersome model is transformed to the small model using knowledge distillation. At the same time, this paper proposes a bias neural network to improve the accuracy of target classes. The experimental results verified that after adding the bias, each target class has a better recognition performance. This results also showed that the bias neural network with fewer layers and parameters has the similar recognition performance as the cumbersome network on target classes. The recognition accuracy is basically higher than the previous one when the bias is not added, and the average accuracy is slightly changed within 5% drop. After the bias is introduced, the recognition ability of the target classes in the small model is close to that of the cumbersome network.

Acknowledgement. The work was supported in part by the National High-tech R&D Program of China (863Program) (2015AA017201) and National Key Research and Development Program of China (2016QY01W0200). The authors are very grateful to the anonymous viewers of this paper.

References

1. Lipikorn, R., Cooharajanone, N., Kijsupapaisan, S., et al.: Vehicle logo recognition based on interior structure using SIFT descriptor and neural network. In: International Conference on Information Science, Electronics and Electrical Engineering, pp. 1595–1599. IEEE (2014)
2. Da, P., Ping, S.: A method of TV logo recognition based on SIFT. In: 3rd International Conference on Multimedia Technology (ICMT-13), pp. 1571–1579. Atlantis Press (2013)
3. Llorca, D.F., Arroyo, R., Sotelo, M.A.: Vehicle logo recognition in traffic images using HOG features and SVM. In: International Conference on Intelligent Transportation Systems, pp. 2229–2234. IEEE (2014)
4. Lu, F., Liu, Y., Zhang, R.: An improved HOG-based vehicle logo location and recognition method. *Study Opt. Commun.* **5**, 26–29 (2012)

5. Biswas, C., Mukherjee, J.: Logo recognition technique using sift descriptor, Surf descriptor and Hog descriptor. *Int. J. Comput. Appl.* **117**(22), 34–37 (2014)
6. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. *Comput. Sci.* **14**(7), 38–39 (2015)
7. Bianco, S., Buzzelli, M., Mazzini, D., et al.: Deep learning for logo recognition. *Neurocomputing* **245**, 23–30 (2017)
8. Shu-Kuo, S., Zen, C.: Robust logo recognition for mobile phone applications. *J. Inf. Sci. Eng.* **27**(2), 545–559 (2014)
9. Hichem, S., Lamberto, B., Giuseppe, S., Alberto, D.: Context-dependent logo matching and recognition. *IEEE Trans. Image Process.* **22**(3), 1018–1031 (2013)
10. Wang, Y., Yang, W., Zhang, H.: Deep learning single logo recognition with data enhancement by shape context. In: *The 2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE (2018)
11. FlickrLogos-32: <http://www.multimedia-computing.de/flickrlogos/>. Accessed 22 July 2018
12. Psyllos, A.P., Anagnostopoulos, C.N.E., Kayafas, E.: Vehicle logo recognition using a SIFT-based enhanced matching scheme. *IEEE Trans. Intell. Transp. Syst.* **11**(2), 322–328 (2010)
13. Liu, X., Zhang, B.: Automatic collecting representative logo images from the internet. *Tsinghua Sci. Technol.* **18**(6), 606–617 (2013)
14. Leonid, K., Joseph, S., Yochay, T., Asaf, T.: Fine-grained recognition of thousands of object categories with single-example training. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 965–974. IEEE (2017)
15. Ning, X., Zhu, W., Chen, S.: Recognition, object detection and segmentation of white background photos based on deep learning. In: *32nd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pp. 182–187. IEEE (2017)
16. Chen, R., Matthew, H., Lyudmila, M., Xiao, J., Liu, W.: Vehicle logo recognition by spatial-SIFT combined with logistic regression. In: *19th International Conference on Information Fusion (FUSION)*, pp. 1228–1235. IEEE (2016)
17. Rajalida, L., Nagul, C., Suppassara, K., Tavinee, I.: Vehicle logo recognition based on interior structure using SIFT descriptor and neural network. In: *2014 International Conference on Information Science, Electronics and Electrical Engineering*, pp. 1595–1599. IEEE (2014)
18. Apostolos, P., Christos-Nikolaos, A., Eleftherios, K.M.: A new method for Vehicle Logo Recognition. In: *2012 International Conference on Vehicular Electronics and Safety (ICVES 2012)*, pp. 261–266. IEEE (2012)
19. Apostolos, P., Psyllos, C.N., Anagnostopoulos, E.K.: Vehicle logo recognition using a SIFT-based enhanced matching scheme. *IEEE Trans. Intell. Transp. Syst.* **11**(2), 322–328 (2010)
20. Xia, L., Qi, F., Zhou, Q.: A learning-based logo recognition algorithm using SIFT and efficient correspondence matching. In: *2008 International Conference on Information and Automation*, pp. 1767–1772. IEEE (2008)
21. Sonawane, D.R., Apte, S.D.: Improved Context Dependent logo matching framework using FREAK method. In: *2016 IEEE International Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT)*, pp. 362–366. IEEE (2016)
22. Tang, S., Zhang, Y.D., Chen, H.: Scalable logo recognition based on compact sparse dictionary for mobile devices. In: *17th International Workshop on Multimedia Signal Processing (MMSP)*, pp. 1–6. IEEE (2015)
23. Leonardo, B., Guillermo, C.C., Pedro, S.: Real-time single-shot brand logo recognition. In: *30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pp. 134–140. IEEE (2017)

24. Afsoon, A.S., Alireza, D., Hasan, F., Mehran, Y.: Persian logo recognition using local binary patterns. In: 3rd International Conference on Pattern Recognition and Image Analysis (IPRIA), pp. 258–261. IEEE (2017)
25. Matheel, E., Abdulmunim, H.K.: Logo matching in Arabic documents using region based features and SURF descriptor. In: 2017 Annual Conference on New Trends in Information & Communications Technology Applications (NTICT), pp. 75–79. IEEE (2017)
26. Bucilua, C., Caruana, R., Niculescu-Mizil, A.: Model compression. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD 2006, pp. 535–541. ACM, New York (2006)
27. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778. IEEE (2016)
28. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2014)



LSTM Encoder-Decoder with Adversarial Network for Text Generation from Keyword

Dongju Park and Chang Wook Ahn^(✉)

School of Electrical Engineering and Computer Science,
Gwangju Institute of Science and Technology (GIST), Cheomdangwagi-Ro,
Buk-Gu, Gwangju 61005, Republic of Korea
cwan@gist.ac.kr

Abstract. Natural Language Generation (NLG), one of the areas of Natural Language Processing (NLP), is a difficult task, but it is also important because it applies to our lives. So far, there have been various approaches to text generation, but in recent years, approaches using artificial neural networks have been used extensively. We propose a model for generating sentences from keywords using Generative Adversarial Network (GAN) composed of a generator and a discriminator among these artificial neural networks. Specifically, the generator uses the Long Short-Term Memory (LSTM) Encoder-Decoder structure, and the discriminator uses the bi-directional LSTM with self-attention. Also, the keyword for input to the encoder of the generator is input together with two words similar to oneself. This method contributes to the creation of sentences containing words that have similar meanings to the keyword. In addition, the number of unique sentences generated increases and diversity can be increased. We evaluate our model with BLEU Score and loss value. As a result, we can see that our model improves the performance compared to the baseline model without an adversarial network.

Keywords: Text generation · Generative Adversarial Network
Natural Language Processing

1 Introduction

Automatically generating natural and contextual sentences is difficult, but it is important in Natural Language Processing (NLP) and Artificial Intelligence. For instance, Natural Language Generation (NLG) can be used when writing poems, novels, and letters. It also plays a significant role in NLP fields such as dialogue system [1], image captioning [2] and machine translation [3].

Recently, in the field of machine learning, Recurrent Neural Network Language Model (RNNLM) [4] is used for NLG work, and it is conceptually simple. However, RNN has a vanishing and exploding gradient issue that causes long-term dependency problems. The emergence of language model using Long Short-Term Memory (LSTM) [5], a variant of RNN, alleviates this problem. Nevertheless, these RNN variation based Language Models still have problems such as the *exposure bias* [6] due to the discrepancy between training and reasoning processes. The difference is that in the

learning process, the next word is predicted by looking at the previous ground truth word, but in the inference process, it is generated by the previously deduced word.

Generative Adversarial Network (GAN) [7] is used as a solution to mitigate the problems mentioned above. In general, the GAN consists of two artificial neural network models that compete against each other in the learning process. One network is *Discriminator* and it is aimed to distinguish whether the given data is real data or synthetic data. The other is *Generator* that synthesizes fine quality generating data to fool the discriminator. During learning, the generator creates synthesized data similar to real data from the latent variable and uses it as training data of the discriminator. The discriminator uses real data and synthesized data from the generator to train and then uses the gradient of the learning loss as a guide to update the parameters of the generator. The GAN was first introduced to deal with the problems of continuous data such as image synthesis and computer vision task, and has been very successful. For example, DCGAN [8] incorporating the convolution layer enables the vector arithmetic operation performed by the generator learned in DCGAN, and it stabilizes the learning in most situations. In addition, StarGAN [9] is a single model that enables image-to-image translation for multiple domains.

On the other hand, text data is, unlike image data, discrete so that it is difficult to transfer a gradient from the discriminator to the generator. A recent study was able to deal with discrete data with SeqGAN [10] using reinforcement learning to resolve the problem. Thereafter, various models for discrete data emerged, including RankGAN [11] for learning by using the relative ranking scores of data and LeakGAN [12] for long text generation.

In this paper, we propose a model for generating text based on a given word. Our work contributes to widening the diversity of text generated. In order to increase the diversity of the text generated from a single word, this model finds a word near to a given word and learns it together. In addition, this method generates a sentence containing words obtained from the main word but not the main word among the input words. The model consists of SeqGAN based LSTM Encoder-Decoder and we use GAN together to improve learning efficiency.

2 LSTM Encoder-Decoder with Adversarial Network

The basic structure of the proposed model consists of a generator that generates realistic text based on a given word, and a discriminator that can distinguish between generated and real data.

2.1 Word Embedding

Text data such as (w_1, \dots, w_T) consisting of T words are divided into words and then converted into real number vectors by skip-gram [13] method, learned by neural networks, a kind of Word2Vec. These words are embedded in the user-defined N dimensions, and the similarity between words can be measured by the distances among the words. In the proposed model, skip-gram is used to select two words with high similarity to a given word.

2.2 Generator

We choose to use the RNN Encoder-Decoder [14] architecture as the generator to generate text with the given word as conditional information. Also, our generator uses LSTM instead of RNN for both encoder and decoder. The encoder reads the input words, such as a sequence of vectors (x_1, \dots, x_T) , to form a context vector c by the last hidden state of the LSTM. The decoder is trained and inferred with the context vector c given by the encoder and all the formerly words $(y_1, \dots, y_{T'})$ it to predict the next word. That is, it can be expressed as follows:

$$p(y_1, \dots, y_{T'} | x_1, \dots, x_T) = \prod_{t=1}^{T'} p(y_t | c, y_1, \dots, y_{t-1}) \tag{1}$$

In the case of the input data of the encoder, one word given for learning and reasoning, two words similar to the word are added, and a total of three words are used as the input data of the encoder. The decoder starts with the last hidden state of the encoder and <START_TOKEN> . If the sentence consists of 20 or fewer words, it is filled with <PAD> (see Fig. 1).

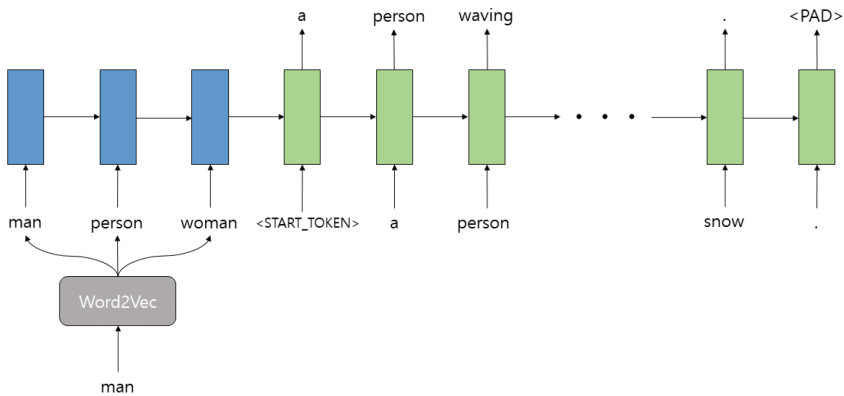


Fig. 1. The illustration of our generator model. The blue rectangles represent the LSTM layer of the encoder and the green are the LSTM layer of the decoder. <man, person, woman> are input data, and <a, person, waving, ..., snow, > are output data. (Color figure online)

2.3 Discriminator

The discriminator is an adversarial network of RNN Encoder-Decoder (generator). We use the bi-directional LSTM using self-attention [15] with one forward layer and one backward layer instead of the Convolutional Neural Network (CNN) used for text classification [16] in SeqGAN. In this model, bi-directional LSTM, hidden states H to be used in self-attention is obtained by concatenating hidden states from forward LSTM layers and backward LSTM layers. After, H is multiplied by the annotation matrix A , and a new sentence embedding matrix M is constructed as

$$M = AH. \tag{2}$$

Subsequently, it passes through fully-connected layers to determine whether it is real or fake text (see Fig. 2)

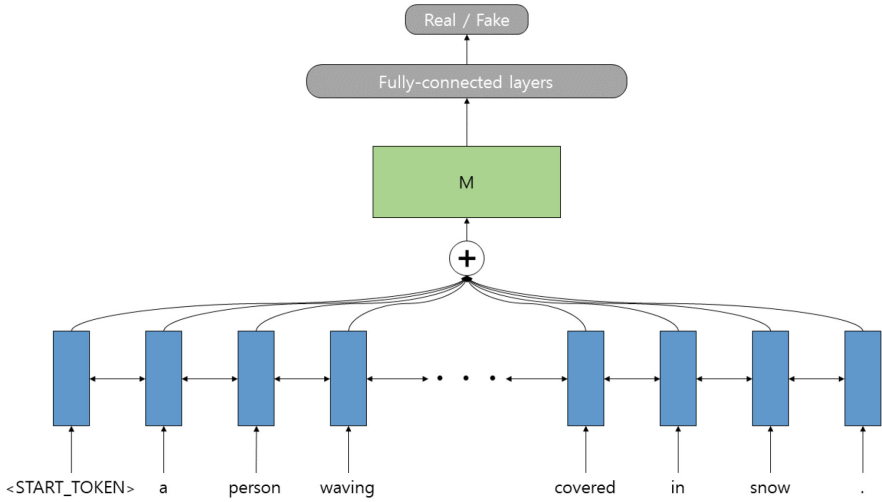


Fig. 2. The illustration of our discriminator model. The blue rectangles represent the hidden layers of the bidirectional LSTM. Circle with + denotes annotation matrix, and the green square is a sentence embedding matrix. In addition, the softmax function layer is used in the final layer to distinguish between real and fake texts. (Color figure online)

2.4 Policy Gradient Training

Discrete data such as text cannot be learned by the generator directly from the discriminator’s gradient. Thus, most models that deal with discrete data in GAN use the REINFORCE algorithm as the policy gradient [17]. The training method of the proposed model is similar to the learning method of SeqGAN which learns by REINFORCE algorithm that includes Monte Carlo Tree Search with the encoder part added.

The objective of the generator model is to generate text that maximizes reward. The objective function of the policy gradient is as follows:

$$J(\theta) = \mathbb{E}_{\pi_{\theta}}[r] = \sum_{s \in S} d(s) \sum_{a \in A} \pi_{\theta}(s, a) Q(s, a) \tag{3}$$

r is the reward, s is the state, $d(s)$ is the probability of reaching the state, a is the action, π is the policy, Q is the action-state value, and θ is policy parameter. The gradient of Eq. (3) can be defined as:

$$\begin{aligned}\nabla_{\theta}J(\theta) &= \sum_{s \in S} d(s) \sum_{a \in A} \pi_{\theta}(s, a) \nabla \log \pi_{\theta} Q(s, a) \\ &= \mathbb{E}_{\pi_{\theta}}[\nabla_{\theta} \log \pi_{\theta}(s, a) Q(s, a)]\end{aligned}\quad (4)$$

We can use this gradient to update the parameters of the generator as follows:

$$\theta \leftarrow \theta + \alpha_k \nabla_{\theta} J(\theta) \quad (5)$$

where α_k is the learning rate at k -step.

3 Experiments

3.1 Datasets

We use COCO Dataset [18] as data for learning and evaluating the proposed model. The datasets are made up of image and text pairs of data for image captioning. In these data pairs, 20000 texts consisting of 15 to 20 words are set as training sets and 3000 as evaluation sets. Also, we set a keyword for each sentence. For convenience, we choose the first noun in the sentence and pair each sentence. In total 7428 words are used for training and evaluation, and a word dictionary is created after embedding in a vector space using a skip-gram.

3.2 Evaluation Measure

BLEU Score [19] is used to evaluate our model and Baseline model as LSTM Encoder-Decoder using only Maximum Likelihood Estimation (MLE). The BLEU Score is originally designed for machine translation, but we use it to compare the similarity of the generated sentence with the ground-truth sentence. The BLEU score calculation method is as follows:

$$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-\frac{c}{r})} & \text{if } c \leq r \end{cases} \quad (6)$$

Then,

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^N w_n \log p_n\right) \quad (7)$$

The BP in Eq. (6) is used as a Brevity Penalty to eliminate the case where the higher the sentence length is, the higher the score is. r is the length of the sentence in the dataset, and c is the length of the generated sentence by the model. Equation (7) denotes the final BLEU score, where w_n is the weight for n -grams and p_n represents the precision for the corresponding gram.

We calculate the BLEU Score from 2-gram to 4-gram based on the evaluation dataset

3.3 Training Setting

First, we embed all the words in the dataset into the vector space using the skip-gram model. They are used not only as word embedding vectors for the generator and the discriminator but also for finding words similar to the keyword. Then, pre-train the generator and discriminator before implementing the adversarial training. When learning the generator, for convenience, select the first noun in a given sentence and designate it as a keyword, and use Word2Vec to get the two words that are used as input to the encoder along with the keyword. The target data is a sentence extracted from the keyword. In the case of the discriminator, it learns to classify by using the generator-synthesized sentence and ground-truth data. Sometimes the generator generates new data for the discriminator learning. Thereafter, adversarial training is carried out with two pre-training models.

4 Experimental Results

Our experimental results are divided into three. The first is the number of unique sentences. Second, we compare the loss of our model trained with GAN and the baseline model learned only with MLE. Finally, the sentences generated from the two models used above are evaluated based on the BLEU Score.

4.1 Number of Unique Sentences

The result of comparing the number of unique sentences according to the number of input words, using the second and third words obtained from the keyword together with the keyword makes more unique sentences than learning a single word. In addition, the use of three words with shuffle is better than its exclusion. In other words, this method can be said that the diversity of generated sentences is widened as the result of comparing the number of unique sentences among the generated sentences.

Furthermore, when shuffling three words for learning and reasoning, those words appear similar in generated sentences, and the sentences that contain no words are also slightly reduced with this method. In contrast, when learning with one word and with three words, the first word is overwhelming, and the second and third words extracted from the first word using Word2Vec are rarely shown. (We do not include it if it comes with the first word). Tables 1 and 2 show the experimental results.

Table 1. The number of unique sentences among the total 20000 sentences generated by the number of input words and the rate of increase in the number of the unique sentences compared to one word.

Number of input words	Number of unique sentences	Rate of increase
One word	15385	–
Three words	16418	6.71%
Three words (Shuffle)	17713	15.13%

Table 2. Percentage of words in sentences generated by the number of words entered

Number of input words	First word (keyword)	Second word	Third word	None
One word	85.24%	0.22%	0.12%	14.42%
Three words	84.85%	0.41%	0.27%	14.47%
Three words (Shuffle)	31.07%	28.70%	26.54%	13.69%

4.2 Learning Curves

We compare the baseline model that we learn using only our model and MLE with Negative Log-Likelihood (NLL). The baseline model learns 250 epochs only with MLE, and our model pre-training 150 epochs before adversarial training. As a result, the two models are similar to each other up to 150 epochs, but after that, we confirm that our model shows the lower losses than the baseline model (see Fig. 3).

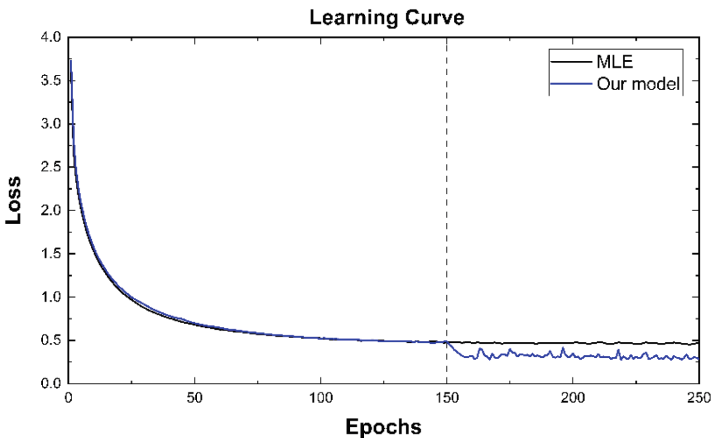


Fig. 3. The learning curve of the two models. The vertical dash line is the end of the pre-training of the proposed model.

4.3 BLEU Score

We measure the BLEU Score (2-gram, 3-gram, and 4-gram) using the test data. The measurement results show that the model using the GAN has the higher score. Table 3 shows the results.

Table 3. The BLEU Score of two models

Method	BLEU-2	BLEU-3	BLEU-4
MLE	0.7958	0.6120	0.4359
Our model	0.8182	0.6255	0.4378

In addition, Table 4 shows the result of the generated text according to the input words. A bold word is a word contained in the generated sentence.

Table 4. Example of the generated text according to the input words.

Input word	Generated text
People/ men /women	- Two men playing a video game in a park bench with a book to it
Restroom/urinals/ toilet	- A toilet and sink next to each other by the small window in the bathroom
Birthday/ cake /chocolate	- A cake with a side of a cake lit candles while sliced leaves on a cake
Fruits / vegetables/stir	- Two fruits , one of them on a plate with a knife and fork
Cheese/pizza/ broccoli	- Broccoli , tomato, cucumber, onion and meat in a plastic container on top of an office
Doubles/apartment/ streets	- Two red streets sign stand on a wall, a lamp, has a lamp base
Racket /ball/ tennis	- A tennis player wearing a blue outfit stretches up high with his racket on a tennis court
Business/ balcony /towels	- A balcony with chairs and a table is seen through a glass door

5 Conclusion

In this paper, we introduced an LSTM Encoder-Decoder model with the adversarial network for text generation from the keyword. Experimental results show that our model using GAN has the higher BLEU score and the lower loss than the MLE model alone. We also found that when we find words similar to keywords, shuffle them with keywords, and use them as input data, we increase the variety of sentences generated. Furthermore, by using this method, not only the sentences containing the keyword but also the sentences containing the similar words obtained from the keyword are generated. For future work, we plan to expand the model so that the relationship between the input words can be entered together, or the model can be inferred directly so that it will work well to generate various sentences we want.

Acknowledgements. This work was supported by Global University Project (GUP) grant funded by the GIST in 2018. Also, this work was supported by the NRF funded by MEST of Korea (No. 2015R1D1A1A02062017).

References

1. Serban, I.V., Sordoni, A., Bengio, Y., Courville, A.C., Pineau, J.: Building end-to-end dialogue systems using generative hierarchical neural network models. In: AAAI, pp. 3776–3784 (2016)
2. Xu, K., et al.: Show, attend and tell: neural image caption generation with visual attention. In: International Conference on Machine Learning, pp. 2048–2057 (2015)
3. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint [arXiv:1409.0473](https://arxiv.org/abs/1409.0473) (2014)
4. Mikolov, T., Karafiát, M., Burget, L., Černocký, J., Khudanpur, S.: Recurrent neural network based language model. In: Eleventh Annual Conference of the International Speech Communication Association (2010)
5. Sundermeyer, M., Schlüter, R., Ney, H.: LSTM neural networks for language modeling. In: Thirteenth annual Conference of the International Speech Communication Association (2012)
6. Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N.: Scheduled sampling for sequence prediction with recurrent neural networks. In: Advances in Neural Information Processing Systems, pp. 1171–1179 (2015)
7. Goodfellow, I., et al.: Generative adversarial nets. In: Advances in neural information processing systems, pp. 2672–2680 (2014)
8. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint [arXiv:1511.06434](https://arxiv.org/abs/1511.06434) (2015)
9. Choi, Y., Choi, M., Kim, M., Ha, J.-W., Kim, S., Choo, J.: StarGAN: unified generative adversarial networks for multi-domain image-to-image translation. arXiv preprint [arXiv:1711.09020](https://arxiv.org/abs/1711.09020) (2017)
10. Yu, L., Zhang, W., Wang, J., Yu, Y.: SeqGAN: sequence generative adversarial nets with policy gradient. In: AAAI, pp. 2852–2858 (2015)
11. Lin, K., Li, D., He, X., Zhang, Z., Sun, M.-T.: Adversarial ranking for language generation. In: Advances in Neural Information Processing Systems, pp. 3155–3165 (2017)
12. Guo, J., Lu, S., Cai, H., Zhang, W., Yu, Y., Wang, J.: Long text generation via adversarial training with leaked information. arXiv preprint [arXiv:1709.08624](https://arxiv.org/abs/1709.08624) (2017)
13. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
14. Cho, K., et al.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint [arXiv:1406.1078](https://arxiv.org/abs/1406.1078) (2014)
15. Lin, Z., et al.: A structured self-attentive sentence embedding. arXiv preprint [arXiv:1703.03130](https://arxiv.org/abs/1703.03130) (2017)
16. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint [arXiv:1408.5882](https://arxiv.org/abs/1408.5882) (2014)
17. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.* **8**, 229–256 (1992)
18. Chen, X., et al.: Microsoft COCO captions: data collection and evaluation server. arXiv preprint [arXiv:1504.00325](https://arxiv.org/abs/1504.00325) (2015)
19. Papineni, K., Roukos, S., Ward, T., Zhu, W.-J.: BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp. 311–318. Association for Computational Linguistics (2002)



Quantum Algorithm for Crowding Method

Jun Suk Kim and Chang Wook Ahn^(✉)

School of Electrical Engineering and Computer Science,
Gwangju Institute of Science and Technology (GIST), Cheomdangwagi-ro,
Buk-gu, Gwangju 61005, Republic of Korea
{junsuk89, cwan}@gist.ac.kr

Abstract. The purpose of this paper is to introduce a simple, basic, yet intuitive method to exploit Grover’s search algorithm on crowding method, a part of general evolutionary algorithm. Specifically, Grover’s algorithm’s capability of “finding multiple needles in a haystack” provides a new, quantum way to properly select a parent individual out of the population that is the closest to its child individual in each niche. We avoid meticulously analyzing the mathematical procedure; rather, we would like to provide a set of concise intuition to help further motivate quantum computing researchers to continue the work. Conclusively, we prove that Grover’s algorithm indeed reduces the upper bound of time complexity required for the crowding method operation. Although our solution does not provide a quadratic or exponential speedup, the fact that quantum adaptation brings changes to existing classical algorithm is worth noticing, and it can attract more computing algorithm researchers to the field of quantum computing.

Keywords: Quantum computing · Grover’s search algorithm
Crowding method

1 Introduction

Quantum computing has evolved from quantum physics, a study of the tiny worlds in every matter. Its core principle is the Schrodinger’s equation that states the wave function of the system, denoted as ψ . A wave function is basically the overall representation of a certain state of quantum system. In quantum computing, this state, or a set of qubits, is represented as $|\psi\rangle$. A qubit is a quantum analogue of the classical computer’s bit, but while a bit can only be either 0 or 1, a qubit contains both of the probabilities of being 0 and 1. This is the key attribute of qubits which leads to the quantum advantages such as quantum superposition and entanglement, and ultimately, quantum parallel computation.

In spite of the aforementioned features, designing quantum algorithms is as tricky as the quantum principles. One main severity that every quantum algorithm suffers is the infamous quantum physics phenomenon, the wave function collapse. Quantum superposition and entanglement enable the quantum box to set problems in multiple states with probabilities and compute them in parallel. As soon as a measurer “opens” the box, however, every but one quantum state disappears, leaving the measurer with

only single outcome. More specifically, only one state remains with its probability of 1, and all the others become eliminated due to their probabilities shifted to 0. This process is irreversible, meaning that the collapsed states will never come back to lives.

Although complicated and non-intuitive, quantum computing has been carefully studied as a stimulus that can significantly enhance the existing computational methods, including evolutionary algorithm. It is unlikely to entirely replace classical computers in every field, but its superiority over several particular types of problem can help researchers resolve them much faster and more efficiently. For example, Patel proposes the quantum optimizing means for evolutionary algorithm, [1] and Nowotniak and Kucharki claim that quantum-inspired genetic algorithm provides the better propagation of building blocks. [2] Hopping on the series of these continuous efforts, we suggest a simple quantum application to evolutionary algorithm, particularly on crowding method, one of the important computations in evolutionary algorithm utilized to deal with premature convergence problems.

2 Quantum Background: Grover's Search Algorithm

In 1996, Grover first introduced his famous quantum search algorithm, Grover's algorithm. [3] Its main role is to utilize quantum superposition in order to find a wanted element out of an unordered array faster than any classical search algorithms do. In other words, it is a practical utilization of so-called quantum parallel computation. Although, unlike Deutsch's or Simon's algorithm, Grover's algorithm only shows a quadratic speedup, its applicability presumably belongs to the highest necessity.

The textbook written by Yanofsky and Mannucci [4] provides a concise and easy way to understand the overall steps of Grover's algorithm, which are briefly discussed here. Essentially, the whole procedure is divided into two parts: phase inversion and inversion about the mean. These two tricks seem rather useless if they are considered one after the other, but when combined, they turn into a powerful tool for searching.

2.1 Phase Inversion

Suppose we have the following four states (Table 1).

Table 1. 4 states with corresponding probability amplitudes

State	00	01	10	11
Probability amplitude	$\frac{1}{\sqrt{4}}$	$\frac{1}{\sqrt{4}}$	$\frac{1}{\sqrt{4}}$	$\frac{1}{\sqrt{4}}$

We want to select $|10\rangle$. Without any manipulation, each state shares an equal probability, $\frac{1}{4}$. As a result, measuring this system leads to observing any one of these states in random. It would require tremendous amount of luck to observe the wanted $|10\rangle$ each time the system is measured. Phase inversion improves the situation by

specifying the desired state’s probability. By cleverly constructing the algorithm’s circuit, the states after the measurement can be written as

$$|\psi\rangle = (-1)^{f(\mathbf{x})} |\mathbf{x}\rangle \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right] = \begin{cases} -1|\mathbf{x}\rangle \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right], & \text{if } \mathbf{x} = \mathbf{x}_0 \\ +1|\mathbf{x}\rangle \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right], & \text{if } \mathbf{x} = \mathbf{x}_0 \end{cases} \quad (1)$$

Refer to [4] for the through mathematical approach and the circuit design. Now the states look slightly different (Table 2).

Table 2. States after phase inversion

State	00	01	10	11
Probability amplitude	$\frac{1}{\sqrt{4}}$	$\frac{1}{\sqrt{4}}$	$-\frac{1}{\sqrt{4}}$	$\frac{1}{\sqrt{4}}$

Note that the sign for $|10\rangle$ has changed from positive to negative. This, however, is not enough to distinguish it from others. A quantum circuit uses the Hadamard matrices to implement its qubit’s superposition, and it requires the probability amplitude of each state to be squared to represent its actual probability after the measurement. Here, because $(-\frac{1}{2})^2$ and $(+\frac{1}{2})^2$ are the same ($\frac{1}{4}$), $|10\rangle$ is not distinguishable from other states. This is exactly why we need another process, inversion about the mean.

2.2 Inversion About the Mean

The role of this part is to amplify the state that flipped in phase inversion, thus making it possible to identify it after measuring. The process is done in the following steps:

1. Find the average of the probability amplitude of every state involved.
2. Flip the probability amplitudes to the other side with respect to the average. For example, if one state is 8 and the average is 3, switch the state to $-8 + 2 \cdot 3 = -2$.

Doing so will change not only the sign of the state, but also its amplitude. This process wouldn’t work if the sign hadn’t been changed, and that is why both tricks are needed. Repeat the loop of phase inversion and inversion about the mean enough so that we can assure that the desired state will be observed at the end. After just one loop, our example states have the following amplitudes (Table 3).

Table 3. States after inversion about the mean.

State	00	01	10	11
Probability amplitude	0	0	1	0

Notice that if finding $|10\rangle$ is done classically, it takes 4 times of checking in the worst case. Grover claims that given m states, the algorithm should repeat at least \sqrt{m}

times to assure $\approx 100\%$ chance to observe the state we want. This is quadratically faster compared to the classical counterparts, with which one has to go over the entire unordered list to find an element in the worst case (that is, m times). Figure 1 shows the visual conceptualization of the overall procedure.

Grover’s algorithm is very versatile, and it’s regarded as one of the most potential quantum solutions in the future once a quantum computer is built. In fact, many have researched its generalization and modification already. For example, Ahuja proved that Grover’s algorithm can be used to find the maximum with unchanged time complexity [5]. Also, Nielson and Chuang proved that the algorithm can find multiple t states, not just one, within $\frac{\pi}{4}\sqrt{\frac{m}{t}}$ loops [6]. Grover’s Algorithm is therefore highly adaptable to many existing tools, including evolutionary algorithms, for further enhancing their capabilities.

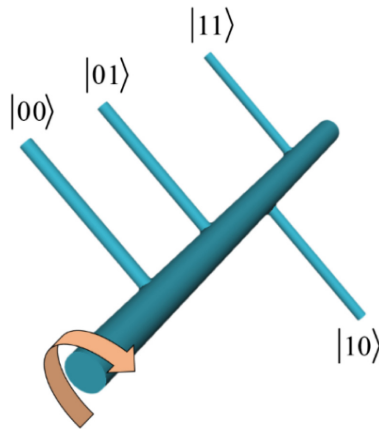


Fig. 1. 3D illustration of Grover’s “phase rotation.”

3 Proposed Approach

3.1 Premature Convergence Problems

In evolutionary algorithm, one of the issues to be carefully dealt with is premature convergence. Real world problems often consist of sets of noticeably “good” individuals that eventually become dominant in the entire population as generations flow. They help converge to solutions - either global or local – relatively fast, but at the same time risk discarding possibilities of other sets of solutions expected to be as good as or even better than themselves [7]. In order to prevent that from taking place, population diversity needs to be preserved throughout the algorithm, and one of the safety locks to guarantee such diversity is adapting crowding method (see Fig. 2).

Crowding method, or often called crowding function, is a means to alleviate the crowdedness of packed individuals by taking every individual of the problem into

consideration [8]. Although there are several variants under the same banner, the core mechanics shares the following procedure:

1. Create a set consisting of randomly chosen individuals out of the population (call it a niche).
2. Create a new individual. Usually, this is done in either of two ways. Create it randomly, or create as an offspring of two parent individuals by performing selection, crossover, and mutation. Make sure that at least one of them is a member of the niche.
3. Replace the member of the niche which is the closest in terms of genotypes to the new individual with it.
4. Confirm the replacement if new fitness evaluation shows the better result.
5. Repeat 1 to 4 for every niche created.

What crowding method conducts is self-evident: it disperses individuals into wider areas, thus providing higher chances of surviving to potential solutions that do not possess good individuals in the initial stage.

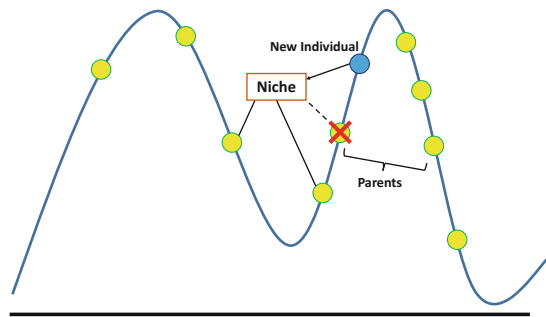


Fig. 2. Crowding method.

3.2 Quantum Adaptation

In order to show how quantum computation can help improve solving the problem above, we first identify the time complexity of crowding method in classical manner. Suppose we have a population of n individuals that need to be processed with crowding method, and we decide to put m individuals in every niche. As mentioned, the method embraces the entire population, dealing with every niche one by one, which requires $\frac{n}{m}$ times of calculation that sets an upper bound $O(\frac{n}{m})$. For each niche, an existing individual that is the closest and replaced by a new individual needs to be identified, and this requires m times of calculation, setting an upper bound $O(m)$. Overall, the entire operation runs under an upper bound of $O(n)$.

We can adapt Grover's algorithm to lower the upper bound. Remember that Grover's algorithm is specialized in finding a specific element out of a set. Therefore, it can be used on finding a parent individual among multiple individuals in a niche that is the closest to a new offspring, pulling down the upper bound from $O(m)$ to $O(\sqrt{m})$. As

a result, the upper bound for the entire operation is now set $O(\frac{n}{\sqrt{m}})$ instead of $O(n)$. The table below lists the maximum number of operations required for 5 example cases comparing classical and quantum computations for crowding method (Table 4).

Table 4. Classical vs. quantum crowding method.

n	m	Classical	Quantum
20	4	20	10
128	16	128	32
1026	36	1026	171
60000	400	60000	3000
10000000	2500	10000000	200000

3.3 Discussion

What can be inferred from our theory above is that the efficiency of our quantum computation depends on the number of m , i.e., the number of individuals in each niche. Figure 3 depicts how m could affect the quantum dominance over the classical counterpart with a fixed amount of n . Seemingly, the more the number of individuals we allocate to each niche, the more quantum advantage we would achieve.

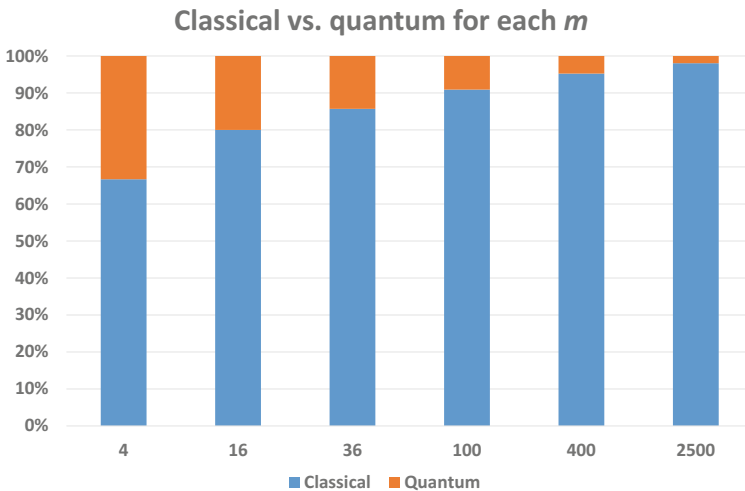


Fig. 3. Comparison of the required maximum amount of operations in percentages for classical and quantum computations with different values of m under an arbitrary, fixed value of n . Notice that the bigger m is, the less the quantum cost becomes with respect to the classical cost.

However, what needs to be done to properly determine the size of each niche, m , is not clear at this stage. In our evaluation above, we showed that the total classical computational cost is independent of m . This indicates that the size of a niche is not a

primary concern for the efficiency of the classical method. It apparently does matter in the quantum method, but unrestrictedly increasing m does not seem to be a good idea. Suppose an extreme case in which $m = n$, i.e., the entire population is a niche itself. Although the upper bound is pulled down to $O(\sqrt{n})$, i.e. boosting the computation to the maximum, this is a bad solution indeed. Creating niches is worthwhile only if particular individuals of similar genotypes can be distinguished, and setting the whole population as a niche makes such distinction virtually impossible. Creating a lot of niches with small m would not help either obviously because it drastically decreases the quantum efficiency. Consequently, we have decided to not make a hasty conclusion now and to admit that further studies are needed on determining proper niche size for the quantum method.

After all, we have shown that the quantum aid can reduce the required number of computation in crowding method, and it is safe to mention that rooms for improvement still exist. Our solution suggests the possibility of quantum adaptation to this particular field, although it is not quadratically or exponentially faster than the classical methods. We have come to concurrence that more fundamental changes are needed in order to increase the speed of the operation far more greatly. Conducting more studies on the applicability of other existing quantum algorithms or even newly developed quantum algorithms could be noteworthy.

4 Conclusion

As we expected, quantum computing has proven to boost the processes of crowding method in securing population diversity. Precisely, we adapted Grover's search algorithm to finding the closest parent individual to its offspring individual for each niche group. We also anticipate that more quantum methodologies can be discovered in this particular area to bring impressive speedups in future.

Acknowledgment. This work was supported by Global University Project (GUP) grant funded by the GIST in 2018. Also, this work was supported by the NRF funded by MEST of Korea (No. 2015R1D1A1A02062017).

References

1. Patel, A.: Optimisation of quantum evolution algorithms. arxiv preprint [arXiv:1503.01429](https://arxiv.org/abs/1503.01429) (2015)
2. Nowotniak, R., Kucharski, J.: Building blocks propagation in quantum-inspired genetic algorithm. arxiv preprint [arXiv:1007.4221](https://arxiv.org/abs/1007.4221) [cs.NE] (2010)
3. Grover, L.: A fast quantum mechanical algorithm for database search. In: Proceedings of the 28th Annual ACM Symposium on Theory of Computing, pp. 212–219. ACM Press, New York (1996)
4. Yanofsky, N., Mannucci, M.: Quantum Computing for Computer Scientists. Cambridge University Press, New York (2013)
5. Ahuja, A., Kapoor, S.: A quantum algorithm for finding the maximum. arxiv preprint [arXiv: quant-ph/9911082](https://arxiv.org/abs/quant-ph/9911082) (1999)

6. Nielsen, M., Chuang, I.: Quantum Computation and Quantum Information. Cambridge University Press, New York (2000)
7. Dick, G., Whigham, P.: Spatially-structured sharing technique for multimodal problems. *J. Comput. Sci. Technol.* **23**(1), 64–76 (2008)
8. Bruno, S., Krahenbuhl, L.: Fitness sharing and niching method revisited. *IEEE Trans. Evol. Comput.* **2**(3), 97–106 (1998)



Random Repeatable Network: Unsupervised Learning to Detect Interest Point

Pei Yan and Yihua Tan^(✉)

National Key Laboratory of Science and Technology on Multi-spectral
Information Processing, School of Automation,
Huazhong University of Science and Technology, Wuhan 430074, China
yhtan@hust.edu.cn

Abstract. This paper presents a Random Repeatable Network (RRN) which is an entirely unsupervised training framework for interest point detection. The existing learning based methods make tradeoff between the unsupervised level and the approximation degree to the objective of repeatability, while our RRN model trains a convolutional neural network whose loss function is directly based on point repeatability without relying on any initial interest point detector. In terms of point repeatability under perspective transform or illumination change, we propose a novel loss function with regularization term of repeatability, which is optimized by an effective iterative algorithm. Experiments demonstrate our model achieves better performance on test data compared to some state-of-the-art interest point detector.

Keywords: Unsupervised learning · Interest point · Repeatability
Deep network

1 Introduction

Interest point detection is the first step in many computer vision tasks such as simultaneous localization and mapping, structure-from-motion, camera calibration, and image matching. Interest points are a sparse set of image locations which are insensitive to both perspective distortion and illumination changes.

Classical detector such as Harris [1], SIFT [2] and FAST [3] explicitly define criteria to extract the points whose two-dimensional gray scale have abrupt changes or have the large curvatures in the edge curve of an image. When the images are acquired outdoors, such approach may have poor performance because of inappropriate parameters. On the contrary, TaSK [4], TILDE [5] use learning based methods to train interest point detectors, but their potentials rely on other initial interest point detectors heavily. More recently, some approaches like Quad-network [6] and SuperPoint [7] that train models by fully unsupervised or self-supervised learning, and LIFT [8] and SuperPoint that use deeper network to construct detectors. Though these models are suitable for complex situation, their optimization objective has no direct connection to point repeatability.

The main difficulty of formulating interesting point detection as machine learning problem is that we could hardly define some pixels as interest point given a set of train

images. Instead of using human supervision, we present a fully unsupervised optimization objective to train a convolutional neural network which could detect repeatable points. The key intuition is that a detector could judge any pixel as an interest point given an image so long as the detector can extract this point from the same scene with some perspective transforms and illumination changes. So we could start from a very poor but adjustable detector, and improve its detection ability based on the interest point set extracted by itself.

2 Related Work

Traditional hand-crafted interest point detectors are typically subdivided into two groups: one is designed to detect corner and that the other one detects blob-like image structure. Corner detectors such as Harris [1], FAST [3], Forstner [17], have a high spatial precision in the 2D image plane but are not scale invariant typically. Blob structure detectors such as SIFT [2], MSER [9] and SURF [10], extract interest point in scale space which give themselves being scale invariance, making them be used widely in stereo matching and object tracking. The major drawbacks of these methods are that they cannot be easily adapted to different context. In many situations they have to keep a large amount of candidate to prevent failures from missing interest points, which leads to high false positive and complex post-processing.

Learning based detectors are more flexible and adjustable to different applications, and their core problem is how to discriminate positive samples (interest point) from negative samples (background). TaSK [4] used Forstner [17] as the initial interest point detector, and considered the points with high repeatability as positive samples on a pre-aligned training image set. Obviously, TaSK is limited by the ability of initially used interest point detector. TILDE [5] is a similar method that took SIFT [2] as initial interest point detector, and designed a more complex loss function to obtain more robust results. LIFT [8] went even further on both the way of collecting train set and the scope of the model application. This approach judges an interest point by a classical Structure of Motion (SfM) system, so pre-aligned training image sets are not necessary. Furthermore, LIFT sequentially trained the network to implement descriptor computation, orientation estimation and point detection, which can more possibly overcome the limit of classical SfM system. Quad-network [6] is a well-known model that was trained with a fully unsupervised method. The most remarkable achievement of Quad-network is that it doesn't rely on any initial interest point detector, but drive the training by an unsupervised loss function based on ranking constraints. SuperPoint [7] can also be thought as an unsupervised approach, which trained a detector with an auto-generated synthetic shapes dataset, and then fine-tuned with a real image set.

All above learning-based methods make tradeoff between the unsupervised level and the approximation degree to the objective of repeatability. In another word, model do not rely on initial interest point detector are more difficult to describe point repeatability directly. This paper proposes a novel unsupervised approach to train a convolutional neural network whose loss function is based on point repeatability directly. Considering the training process seemed like judge interest point randomly, we name this detector as Random Repeatable Network (RRN).

3 Learning Random Repeatable Network

3.1 Overview of Our Approach

RRN is motivated by the key intuition that interest point should be defined by detector itself, rather than defined by people. In other word, if a point could be detected repeatedly under some perspective distortion and illumination changes, it is no doubt this point is an interest point for the detector. So the remaining problems are: (1) how to decide point repeatability given a detector, and (2) how to improve this detector to detect more repeatable interest point.

For the first problem, we need an appropriate train image set. We could calculate the interest point response of each pixel given an image using CNN [11–13], then the core requirement is to judge whether these responses are concordant between different images with some perspective distortion or illumination change. Therefore, we required the known correspondence between several images in train image set which are captured under the same scene. As we known there are several feasible ways to achieve this purpose, such as acquiring aligned image by stationary cameras, warping existing image with geometric transformation, re-projecting image with known 3D structure data, and so on. Clearly, the first way represents the same scene with illumination changes, and the last two reflect different perspective transform.

Now we could solve the second problem directly. Benefiting from the adjustable CNN model, we could select those highly repeatable points as positive samples (interest point), and other points should be negative samples (background), then used some gradient descent based method to train our model.

Our RRN model is a fully-convolutional neural network [14]. Figure 1 showed the architecture of RRN and the overall training process.

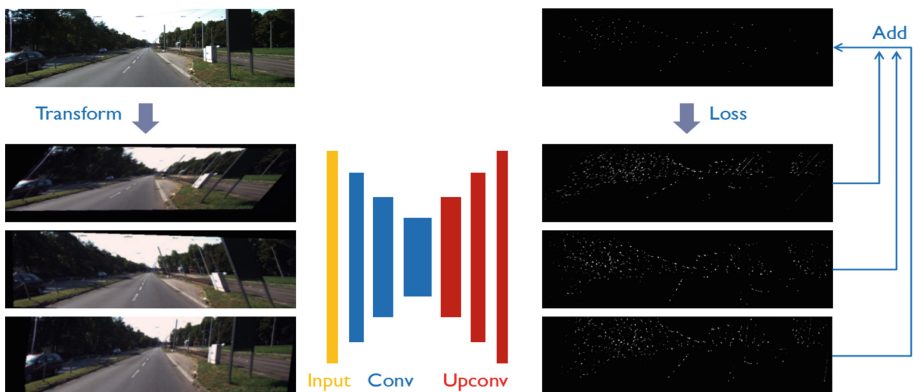


Fig. 1. Overview of the training of Random Repeatable Network. First, reference image is transformed to generate several new images which are fed into RRN. Then the output response maps would be transformed back and accumulate. The final response map is used to calculate the loss.

3.2 Formulation and Solution Process

Here we first introduce some relate notations. Assume the training image set contains n images. We choose m images with known correspondence from the training image set which are represented as I_1, I_2, \dots, I_m . Then assume the m images are transformed from a virtual reference image RI (suppose RI contains enough information) and $I_j = T_j(RI)$, $I_j = T_j^{-1}(RI)$. Here both T_j and T_j^{-1} represent some transformation. Note that T_j and T_j^{-1} are known because the correspondence between m images is known, and the RI could be chosen arbitrarily. The output response map of I_j given by RRN is p_j , and we always restrict the value of any pixel in p_j is between 0 and 1. Assume there is a ground truth binary image y_j whose pixel value equal to 1 if and only if the correspond point in I_j is an interest point. Clearly, I_j, p_j, y_j have the same size.

Given the training image set, if for any image I_i we know the ground truth y_i , and we calculate p_i by RRN, then in the view of maximum likelihood we get the loss function:

$$L_d = \sum_i (y_i \ln p_i + (1 - y_i) \ln(1 - p_i)) \tag{1}$$

Of course y_i is unknown, but from the assumption of RRN we could estimate y_i use RRN itself. More precisely, by choosing I_1, I_2, \dots, I_m with known correspondence, and using current RRN we first obtain p_1, p_2, \dots, p_m . Now we set a fixed threshold A (for example, $A = 0.5$), and note the Non-Maximum Suppression operation as function NMS , then we could calculate the repeat number of each point in reference image RI :

$$Rep(RI) = \sum_j^m T_j^{-1}(NMS(p_j) > A) \tag{2}$$

For simplicity we directly define the logical expression return 1 if the expression is true, otherwise return 0. Then we could choose a threshold N , and only the points whose repeatable number are large than N would be considered as the interest points. Now we could have the estimation of y_j as \hat{y}_j :

$$\hat{y}_j = T_j(Rep(RI) > N) \tag{3}$$

For most of applications, we typically have enough priori about how many interest points we expected appear in an image, so we could define another loss function by the number of interest points in RI :

$$L_r(RI) = f_{priori}(\|Rep(RI) > N\|_1) \tag{4}$$

We divide the training image set into K group, which makes sure that in any group G_k the correspondence between each pair of images are known. Finally, we could complete the loss function:

$$L = L_d + \lambda L_r(RI) \quad (5)$$

$$L_d = \sum_k^K \sum_j^{m_k} (\widehat{y}_{kj} \ln p_{kj} + (1 - \widehat{y}_{kj}) \ln(1 - p_{kj})) \quad (6)$$

$$L_r(RI) = \sum_k^K f_{\text{priori}} \left(\left\| \sum_j^{m_k} T_{kj}^{-1} (NMS(p_{kj}) > A) > N_k \right\|_1 \right) \quad (7)$$

Here the way to estimate \widehat{y}_{kj} is same as (3). It is very hard to minimize the loss function directly because of the regularization term, and the selection of coefficient λ is not trivial. Fortunately, considering the form of f_{priori} in most applications, we could divide the problem into two subproblems and propose an effective iterative algorithm to minimize the entire loss function.

Subproblem 1: Calculate p by RRN and Minimize L_r

In the most applications, we typically expect to extract interest point sparsely and representatively, so the simplest priori function should be:

$$f_{\text{priori}}(\text{num}) = \begin{cases} 0, N_{\min} \leq x \leq N_{\max} \\ +\infty, \text{otherwise} \end{cases} \quad (8)$$

This means we could only accept the point number between $[N_{\min}, N_{\max}]$, and now the parameter λ could be ignored. Now only the selection of N_k would impact L_r (keep in mind that we have fix the value A). So we just need go through the available value of N_k , and choose the best combination corresponding to the minimum of L_r . The computational complexity is $O(n^2)$ because the maximum of N_k is n . Typically, $N_k \ll n$, so this process would be very fast.

Subproblem 2: Estimate \widehat{y} and Implement Gradient Descent

On the basis of the solution of subproblem 1 and equal (3), \widehat{y} could be estimated, then the gradient of L could be calculated. Because the gradient of L_r always equals to 0, we just need differentiate L_d , which is well known as the cross entropy loss function. Then any optimization algorithm such as random gradient descent could be used to update the parameters of the network.

In summary, the entire training process of RRN is described as below: (1) initialize the parameters of RRN, (2) solve subproblem 1, (3) optimize subproblem 2, (4) repeat (2) and (3) until convergence.

4 Experiments

4.1 Implementation Details

Our RRN model was a fully-convolutional neural network. It contains six convolutional layers with total stride of 8, and then follows three transposed convolutional layers which make sure the output has the same size as the input image. RRN is trained on KITTI [18] trainset by Matlab Neural Network Toolbox with a single GTX 1080 GPU. We fix our hyperparameters $A = 0.5$, $N_{\min} = 50$, $N_{\max} = 150$. In every batch, we

warp each training image with 10 random perspective projection so that the correspondence between any two of these ten images are known. With the training set we can calculate the gradient of the loss function which is the key of our iterative method. It deserves consideration that our training process seems more like online learning because we always generate new warped images for every batch. We set learning rate as $1e-2$ initially, and decrease it by half after every five epochs. In our experiment the training process converges after thirty epochs.

To form the test data set, we warp every image in KITTI with five random perspective projection and record the corresponding projection matrix. All the test images with the size of 664×200 are downsampled from KITTI training set and test set. These warped images and projection parameters construct our test benchmark.

4.2 Quantitative Result

We compare RRN to the state-of-the-art point detection algorithms, namely BRISK [15], SURF [10], SIFT [2], FAST [3] and KAZE [16]. Evaluation metric is chosen as average repeatable number (ARN), which is similar to the repeatability metric in SuperPoint [7]. ARN is defined as (9). Here the meaning of Rep , RI , K is same as in equal (2) (7), and we fix $m = 5 \forall k \in [1, K]$ so that $ARN \in [1, 5]$. Specially we use a correct distance of 2 pixels to judge points correspondence.

$$ARN = \frac{1}{K} \sum_k^K (\|Rep(RI_k)\|_1 / \|Rep(RI_k)\|_0) \quad (9)$$

All compared algorithms are implemented in Matlab. The evaluation was performed on Intel i7-7700k CPU and GTX 1080 GPU

Table 1 gives the comparison results of several classical detectors and our model with ARN metric on To be fair, all algorithms follow Non-Maximum Suppression with seven pixels, and we only select 100 points with the largest response if any algorithm extract more than 100 points on one image.

Table 1. The comparison with ARN metric and processing speed.

Algorithm	ARN on train set	ARN on test set	Speed/fps
BRISK [15]	2.20	2.18	4.6
SURF [10]	1.94	1.94	162.8
SIFT [2]	2.32	2.32	5.6
FAST [3]	2.55	2.53	1539.4
KAZE [16]	2.34	2.30	28.6
RRN (ours)	2.63	2.64	7.7

Our method achieves better performance compared to the other state-of-the-art algorithms. This results show that RRN that is randomly initialized and updated by the alternative iterative algorithm can finally produce a very powerful detector adapting to perspective projection. Considering the success was achieved by a full unsupervised

way, our RRN model should have more potential to be applicable to the case with more complex transformation and illumination change.

4.3 Qualitative Results

First some qualitative examples of RRN are shown in Fig. 2. These images are transformed from KITTI test set with random perspective projection. From Table 1 we know it's not very easy to extract repeatable points from them.



Fig. 2. The interest points extracted by RRN from the images transformed from KITTI test set with random perspective projection. Note these points are highly repeatable, which shows RRN can adapt itself to perspective projection.

Though our RRN is “simply” trained to adapt to perspective projection, it can extract repeatable points from the stereo images and video sequences, as shown in Figs. 3 and 4. Although the test images are not connected with the training data, these results show RRN is suitable to the applications such as stereo matching and object tracking. Especially, not belonging to KITTI dataset, the stereo image pair in the bottom row of Fig. 3 contains more perspective distortion.



Fig. 3. The interest points extracted by RRN from stereo image pair in KITTI test set (top) and no belong to KITTI (bottom). Note RRN has never seen the correspondence between stereo images, which demonstrates the generalization ability of RRN on stereo images.



Fig. 4. The interest points extracted by RRN from image sequence in KITTI test set. Note RRN has never seen the correspondence between the sequential images, and the difference between the sequential images is more significant than the stereo images or the above images with perspective projection, which demonstrates the generalization ability of RRN on sequential images.

4.4 Discussion

Being initialized randomly and optimized with the loss function with regularization of repeatable number, RRN is capable of surprisingly learning how to extract repeatable interest point. The ability can be interpreted from the viewpoint of reinforcement learning because the training of RRN is similar to the process of exploration and exploitation.

Starting with random initialization, RRN selected the pixels with higher responses as the interest points, then NMS keep some of them as the crucial points. At the beginning moment the response difference between the interest point and the background is very small, but this indicates that RRN is in the process of exploration to find the optimization direction. Experiments reflect this small response difference is enough to make RRN tend to the preference. The more significant the preference is, the more proportion exploitation had. So RRN will be stable in the last stage of training. It is worth mentioning that we always generate new warped images for every batch as mentioned in 4.1, which made sure RRN be continuously doing exploration. This process is beneficial to kept RRN from being overfitting over train image set, and the results in 4.3 demonstrate the generalization ability of RRN.

5 Conclusion

We have presented a fully-convolutional neural network architecture for unsupervised learning interest point detection with a novel loss function, and proposed an effective iterative optimization algorithm. Experimental results demonstrate our RRN model can extract interest point with higher repeatability compared to several state-of-the-art detectors, and have potential to be applied to the fields such as stereo matching and object tracking. Future work will investigate whether RRN can be trained as a point descriptor, which will extend the application range of RRN much more widely.

Acknowledgments. This research was partially supported by National Foundation of China under Grants 41371339 and The Fundamental Research Funds for the Central Universities No. 2017KFYXJJ179.

References

1. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proceedings of Fourth Alvey Vision Conference, pp. 147–151 (1988)
2. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **60**(2), 91–110 (2004)
3. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 430–443. Springer, Heidelberg (2006). https://doi.org/10.1007/11744023_34
4. Strecha, C., Lindner, A., Ali, K., Fua, P.: Training for task specific keypoint detection. In: Denzler, J., Notni, G., Süße, H. (eds.) DAGM 2009. LNCS, vol. 5748, pp. 151–160. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03798-6_16
5. Verdie, Y., Yi, K., Fua, P., et al.: TILDE: a temporally invariant learned detector. In: 2015 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5279–5288. IEEE, New York (2015)
6. Savinov, N., Seki, A., Ladicky, L., et al.: Quad-networks: unsupervised learning to rank for interest point detection. In: 2017 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)
7. DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperPoint: self-supervised interest point detection and description. arXiv preprint [arXiv:1712.07629](https://arxiv.org/abs/1712.07629) (2017)
8. Yi, K.M., Trulls, E., Lepetit, V., Fua, P.: LIFT: learned invariant feature transform. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9910, pp. 467–483. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46466-4_28
9. Nistér, D., Stewénius, H.: Linear time maximally stable extremal regions. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008. LNCS, vol. 5303, pp. 183–196. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88688-4_14
10. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006). https://doi.org/10.1007/11744023_32
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
12. Nair, V., Hinton, G.E.: Rectified linear units improve restricted Boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-2010), pp. 807–814 (2010)
13. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv preprint [arXiv:1502.03167](https://arxiv.org/abs/1502.03167) (2015)
14. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation 2015. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440. IEEE, New York (2015)
15. Leutenegger, S., Chli, M., Siegwart, R.Y.: BRISK: binary robust invariant scalable keypoints. In: 2011 IEEE International Conference on Computer Vision, pp. 2548–2555. IEEE, New York (2011)
16. Alcantarilla, P.F., Bartoli, A., Davison, A.J.: KAZE features. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7577, pp. 214–227. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33783-3_16

17. Förstner, W., Dickscheid, T., Schindler, F.: Detecting interpretable and accurate scale-invariant keypoints. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 2256–2263. IEEE, New York (2009)
18. Fritsch, J., Kuhl, T., Geiger, A.: A new performance measure and evaluation benchmark for road detection algorithms. In: 16th International IEEE Conference on Intelligent Transportation Systems, pp. 1693–1700. IEEE, New York (2013)



An Orthogonal Genetic Algorithm with Multi-parent Multi-point Crossover for Knapsack Problem

Xinchao Zhao¹(✉), Jiaqi Chen¹, Rui Li¹, Dunwei Gong²,
and Xingmei Li³

¹ School of Science, Beijing University of Posts and Telecommunications,
Beijing 100876, China

zhaoxc@bupt.edu.cn

² School of Information and Control Engineering,
China University of Mining and Technology, Xuzhou 221116, China

³ School of Economics and Management,
North China Electric Power University, Beijing 102206, China

Abstract. According to the inherent feature of knapsack problem, a multi-parent multi-point crossover operation (MP2X) is proposed, which is implanted with orthogonal experimental design method. The aim of implementing orthogonal experimental design method to MP2X operation is to fully utilizing the inherent information from multiple component of multiple individuals. Based on MP2X operation and orthogonal design method, a genetic algorithm variant (MPXOGA) is proposed in this paper. The simulation results on classic knapsack instances show that MPXOGA is better than several other solvers, including Hybrid Genetic Algorithm (HGA), Greedy Genetic Algorithm (GGA), Greedy Binary Particle Swarm Optimization Algorithm (GBPSOA) and Very Greedy PSO (VGPSO) in the ability of finding optimal solution, the efficiency and the robustness.

Keywords: Knapsack problem · Genetic algorithm
Multi-parent Multi-point crossover

1 Introduction

Knapsack problem (KP) problem [1] is a combination optimization problem of NP-hard. It has important applications in system processing and database allocation, resource allocation, and investment decisions in the industrial and financial fields. At present, the exact algorithms such as branch and bound method and graph theory method for solving knapsack problems and the approximate algorithms such as hybrid genetic algorithm (HGA) and greedy genetic algorithm (GGA) have been widely studied and applied in academic circles, and have been successfully applied to many problems. This article focuses on a solution to the knapsack problem.

Genetic algorithm is an effective method to solve difficult problems based on Darwin evolutionary theory of biological population evolution. Chromosomes with low fitness are basically eliminated by means of natural selection, chromosome crossing,

and genetic variation. After several iterations of this cyclical selection, the population and individuals that are ultimately retained are highly self-contained.

In this paper, orthogonal genetic operators are used to enhance the ability of the algorithm to search finely, and the probability of searching for the optimal solution is improved. The algorithm is simple to implement and has good results. The calculation results show that the algorithm can effectively solve the large-scale 0/1 knapsack problem.

2 Background Knowledge

2.1 Mathematical Model of Knapsack Problem

The knapsack problem [2] is described as: Given the volume and value of a group of n objects, that is, a given object has its own weight w_i and value c_i , and set the backpack's weight to C . Within the defined load-bearing range, how to select the loaded objects maximizes the total value of the objects loaded into the backpack.

The specific mathematical description is as follows: Given a backpack with a maximum capacity of C , $C > 0$, n objects, the weight and value are expressed as w_i , c_i ($0 \leq i \leq n$), $w_i \geq 0$, $c_i \geq 0$, respectively. It is required that the objects loaded into the bag have the maximum total value under the condition that the maximum weight of the backpack is not exceeded. That is, a d -ary vector satisfies the following conditions:

$$\max f(x) = \sum_{i=1}^n c_i x_i \quad (1)$$

$$\text{s.t.}, \quad \sum_{i=1}^n w_i x_i \leq M \quad (2)$$

$$x_i = 0 \text{ or } 1, i = 1, 2, \dots, n \quad (3)$$

Formula (1) is the total value of the objects loaded into the bag at this time. Equations (2) and (3) are constraints, and WX is the total capacity of the bag loaded at this time, which must meet the total weight of the backpack. Equation (3) shows that when $x_i = 1$, the i -th object is loaded into the packet; when $x_i = 0$, the i -th object is not loaded into the packet.

2.2 Traditional Genetic Algorithm

Population is the basic composition of biological evolution, carried out in a group-driven manner. A chromosome is a sequence of alleles. Different species in nature undergo natural selection, and the criteria for the survival of the fittest are to evolve the survival of the fittest and to screen out biological populations and individuals with better adaptability and natural changes with a certain probability. Among them, Reproduction is a common characteristic of all life; variation ensures that any living system can reproduce itself in the positive entropy world; competition and choice are unavoidable conclusions for the ever-expanding population confined to a limited area.

The genetic algorithm [3] directly uses the value of the objective function as the search information, and it is more convenient than the traditional optimization algorithm to require the derivative of the objective function and other information. Genetic algorithms often perform multi-point search in the solution space at the same time, avoiding the situation that it is easy to fall into Local optimum. Genetic algorithms use adaptive probabilistic search techniques to increase the flexibility of the search process. After several generations, the algorithm converges to the best chromosome, which hopefully represents an optimal or suboptimal solution to the problem. In past few years, the GA community has turned much of its attention to the optimization and industrial engineering, resulting in a fresh development of research and applications (Fig. 1).

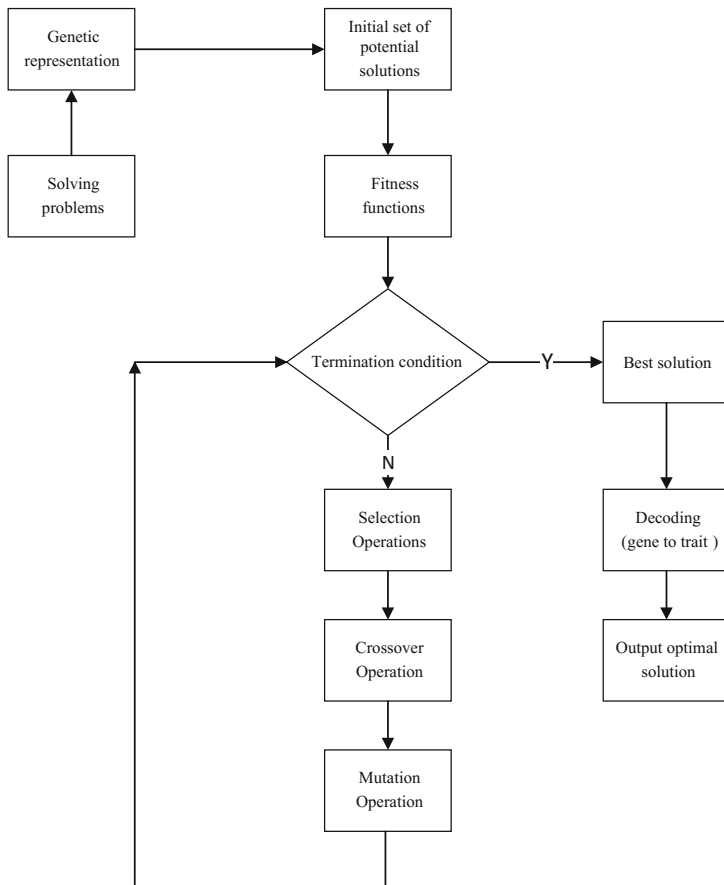


Fig. 1. The flowchart of genetic algorithm

2.3 Previous Research

The greedy transformation method adopted by the hybrid genetic algorithm (HGA) proposed in [4] solves the problem of numerical optimization of backpacks. He et al. [5] proposed a new definition of greedy transformation, and combined this method with genetic algorithm to form a new hybrid genetic algorithm: Greedy Genetic Algorithm (GGA). The VGPSOA proposed in [6] combines the particle swarm optimization algorithm with the greedy thought, and proposes a more greedy hybrid particle swarm optimization algorithm for solving the 0/1 knapsack problem. Ye et al. [7] proposed a hybrid particle swarm algorithm proposed by GBPSOA to solve the 0/1 knapsack problem, which improves the search ability and the probability of finding the optimal solution.

2.4 Orthogonal Design

GA is prone to “premature convergence” in the search process. In response to this phenomenon, people combine genetic algorithms with some traditional optimization methods in the problem domain, such as the steepest descent method and the conjugate gradient method, which improves the efficiency of the algorithm to some extent [8–11]. Another question to consider is: How do sample test points around a given point? The orthogonal design method provides the answer to this question. By using the properties of the orthogonal design method, the optimal combination group is obtained which is regarded as the initial population of the genetic algorithm, and then iterated by the genetic algorithm, thereby improving the speed at which the genetic algorithm converges to the global optimal solution.

The implementation of orthogonal design [12] is usually done by an orthogonal table. The orthogonal table can be defined as: orthogonal table, the matrix is $H = (h_{ij})_{n \times m}$, a type of orthogonal table $L_n(r_1 \times \dots \times r_m)$ if

- (i) $\forall j(j = 1, \dots, m), h_{ij} \in \{0, 1, \dots, r_j - 1\}, i = 1, 2, \dots, n$.
- (ii) Each $\forall j(j = 1, \dots, m)$ in $h_{ij}(i = 1, 2, \dots, n)$ different non-negative integer $k(k \in \{0, 1, \dots, r_j - 1\})$ occurs for the same number of times n/r_j .
- (iii) $\forall j_1, j_2(1 \leq j_1, j_2 \leq m)$ In a sub-array consisting of H 's j_1, j_2 columns, the number of occurrences of each pair of different non-negative integer numbers is equal to $n/(r_{j_1} \bullet r_{j_2})$, where n, m, r_1, \dots, r_m are positive integers.

In actual experimental situations, the general experimental system has F factors. When each factor has Q levels, there are Q^F combinations. If a comprehensive combination experiment is performed, a group experiment is performed. But when Q and F are large, it is impossible to do Q^F group experiments. Orthogonal experimental design is an effective method to solve multi-factor and multi-level experimental problems. It uses orthogonal tables to arrange a few experiments and can find the best or better experimental conditions, so it is widely used for optimization. $L_M(Q^F)$ represents an orthogonal table with F factors and Q levels, where L represents the Latin square and M represents

the horizontal combination number. There are M lines in $L_M(Q^F)$, each line represents a horizontal combination. To apply an orthogonal table $L_M(Q^F)$, just needing to select M combinations to do the experiment, where M is generally much smaller than Q^F .

2.5 Development

In order to solve a class of important algorithms for knapsack problems, genetic algorithms have made great progress in recent years, resulting in such hybrid genetic algorithms (HGA), greedy genetic algorithm (GGA), quantum genetic algorithm (QGA), viral infection genetic algorithm, small world algorithm, etc. The genetic algorithm is a random global search and optimization method that mimics the evolutionary mechanism of biological evolution in nature. It draws on Darwin's progress and Mendel's genetics. Its essence is an efficient, parallel, global search method that automatically acquires and accumulates during the search process. Knowledge of the search space and adaptively control the search process to find the optimal solution. Because the algorithm has excellent performances such as simplicity, versatility and robustness, it has been widely used in many fields such as combinatorial optimization, machine learning, planning and artificial life, and has become one of the key applications of intelligent computing.

3 Orthogonal Genetic Algorithm for Knapsack Problem

3.1 Research Motivation

In recent years, many scholars have applied the robust genetic algorithm to the solution of the 0-1 knapsack problem, and have received good results in the quality of problem solving. In the traditional genetic algorithm, the population converges slowly, and it is easy to fall into local convergence. This paper proposes a multi-parent multi-point orthogonal genetic operator, which preserves the diversity of the population in the later stage, avoids the premature convergence of the population into local convergence, and improves the solution quality and algorithm efficiency of the problem.

Traditional methods for solving knapsack problems include dynamic programming, analytics, and exhaustive methods. Dynamic programming algorithms are usually used to solve problems with certain optimal solution properties. In this kind of problem, there may be many feasible solutions. Each solution corresponds to a value. This paper hopes to find the optimal solution (maximum value or minimum value).

3.2 Multi-parent Multi-point Crossover

Multi-parental crossing helps to improve the performance of the genetic algorithm. With the introduction of multiple parents in the crossover operation, the possibility of

the super-individual copying itself to the child is reduced, which means that more diverse solution space is brought about. Search results, thereby reducing the risk of genetic algorithm precocity. In this technique, the child is derived from three randomly chosen parents. Each factor is divided into Q levels, so the intersection operation of the parent individuals in the N dimension is transformed into the t factor and the Q level experiment problem. Then the orthogonal table $L_M(Q^F)$ is constructed, and the orthogonal design experiment is arranged to generate the M children. On behalf of individuals where $F = t$. When the number of levels is $Q = 2$, the adaptive orthogonal crossover operator becomes a multi-point crossover operator, and the location of the factor division of the parental individual is the location of the crossover operation. In the multi-point crossover, the crossover combinations have many ways to exist. As the number of intersections increases, the number of combinations will increase sharply. The number of intersections is the number of factors in the orthogonal table, and the position of the intersection is the factor division of the parent. Position, then construct the orthogonal table, and carry out orthogonal experiment design to produce representative combinations to generate offspring individuals, which greatly improves the search efficiency. The proposed algorithm may require different orthogonal arrays for different optimization problems. Although many orthogonal arrays have been tabulated in the literature, it is impossible to store all of them for the proposed algorithm. We will only need a special class of orthogonal arrays $L_M(Q^F)$, where Q is odd and $M = Q^J$, where J is a positive integer fulfilling.

$$N = \frac{Q^J - 1}{Q - 1} \quad (4)$$

In this subsection, we design a simple permutation method to construct orthogonal arrays of this class. We denote the column of the orthogonal array $[a_{i,j}]_{M \times N}$ by a_j , $j = 1, 2, (Q^3 - 1)/(Q - 1) + 1, \dots, N$. Columns a_j for $j = 1, 2, (Q^{J-1} - 1)/(Q - 1) + 1$ are called the basic columns, and the others are called the non-basic columns. We first construct the basic columns, and then construct the non-basic columns. The details are Algorithm 1.

Algorithm 1:

Construction of $L_M(Q^F)$:

Step 1: Select the smallest J fulfilling $(Q^J - 1)/(Q - 1) \geq N$

Step2: If $(Q^J - 1)/(Q - 1) = N$, then $N' = N$ else $N' = (Q^J - 1)/(Q - 1)$.

Step 3: Execute Algorithm 1 to construct the orthogonal array

Construction of Orthogonal Array:

Step3. 1: Construct the basic columns as follows:

FOR k=1 TO J DO

BEGIN

$$j = \frac{Q^{k-1} - 1}{Q - 1} + 1$$

FOR i=1 TO Q^j DO

$$a_{i,j} = \left[\frac{i-1}{Q^{j-k}} \right] \bmod Q;$$

END

Step 3.2: Construct the non-basic columns follows:

FOR K=2 TO J DO

BEGIN

$$j = \frac{Q^{k-1} - 1}{Q - 1} + 1;$$

FOR S=1 TO j-1 DO

FOR t=1 TO Q-1 DO

$$a_{j+(s-1)(Q-1)+t} = (a_s \times t + a_j) \bmod Q;$$

END

Step 3.3: Increment by one for all

$$1 \leq i \leq M \text{ and } 1 \leq j \leq N$$

Step 4: Delete the last $N' - N$ columns of $L_{Q^J}(Q^{N'})$ to get $L_M(Q^N)$ where $M = Q^J$

3.3 Correction

Based on the above analysis, the algorithm can be further enhanced to be more efficient [13].

Make a greedy correction to the non-feasible solution X in the iteration: 1 Descending all the items in the solution vector X to 1 in descending order of price/performance ratio Preface, the item with the worst price/performance ratio is taken out from the backpack, so the allele at the corresponding position changes from 1 to 0, and continues to judge whether the changed chromosome satisfies the condition of the capacity constraint of the problem. If the constraint is not satisfied, the above is repeated. Continue operation until the capacity constraint of the problem is satisfied;

the correction ends when the infeasible solution is just corrected to the feasible solution. The proposed correction strategy further enhances the algorithm's fine search ability and improves the efficiency of searching for the optimal solution. The current search path is modified or updated in an indefinite degree to generate descendant individuals with higher fitness values, and the search efficiency of the algorithm is improved. Once the operation of the algorithm is modified, it can achieve the traditional maintenance of the genetic algorithm and search for the purpose of path diversity.

3.4 Orthogonal Genetic Algorithm

Given a backpack with a maximum capacity of C , $C > 0$, n objects, the weight and value are expressed as w_i , c_i ($0 \leq i \leq n$), $w_i \geq 0$, $c_i \geq 0$, respectively. The details of the overall algorithm are as follows.

Orthogonal Genetic Algorithm

Step (1) Initialization

Randomly create an initial generation of N binary strings $p = \{x_1, x_2, x_3, \dots, x_N\}$, and initialize the generation number gen to 0.

Step (2) Calculate fitness values

Calculate individual objective function values and fitness values in a population.

Step (3) Construction of $L_M(Q^N)$

Execute Algorithm 1 to construct $L_M(Q^N)$

Step (4) Population Evolution

WHILE (stopping condition is not met) DO BEGIN

Step 4.1: Orthogonal Crossover

Each chromosome is selected for crossover with a corresponding probability.

Step 4.2: Mutation

Each chromosome undergoes mutation with $z \in (l, u)$ probability p_m . To perform mutation on a chromosome, randomly generate an integer $j \in (1, N)$ and a real number $z \in (l, u)$, and then replace the j -th component of the chosen chromosome by z to get a new chromosome.

Step 4.3: Selection

Among the chromosomes and those generated by crossover and mutation, select the new chromosomes with the least cost to form the next generation.

Step 4.4: Increment the generation number gen .

END

4 Experiments and Performance Measures

4.1 Algorithms Instances

In order to facilitate comparison with similar algorithms, this paper directly uses the backpack example widely used in the literature to solve the performance of the algorithm.

According to the comparative analysis, it is assumed that the item size, weight set, value set and maximum load capacity of the backpack are n , W , V , and M , respectively.

Example 1: $M = 878$, $n = 20$,

$V = \{92, 4, 43, 83, 84, 68, 92, 82, 6, 44, 32, 18, 56, 83, 25, 96, 70, 48, 14, 58\}$,

$W = \{44, 46, 90, 72, 91, 40, 75, 35, 8, 54, 78, 40, 77, 15, 61, 17, 75, 29, 75, 63\}$.

Example 2: $M = 1\ 000$, $n = 50$,

$V = \{220, 208, 198, 192, 180, 180, 165, 162, 160, 158, 155, 130, 125, 122, 120, 118, 115, 110, 105, 101, 100, 100, 98, 96, 95, 90, 88, 82, 80, 77, 75, 73, 72, 70, 69, 66, 65, 63, 60, 58, 56, 50, 30, 20, 15, 10, 8, 5, 3, 1\}$,

$W = \{80, 82, 85, 70, 72, 70, 66, 50, 55, 25, 50, 55, 40, 48, 50, 32, 22, 60, 30, 32, 40, 38, 35, 32, 25, 28, 30, 22, 50, 30, 45, 30, 60, 50, 20, 65, 20, 25, 30, 10, 20, 25, 15, 10, 10, 4, 4, 2, 1\}$.

Example 3: $M = 6\ 718$, $n = 100$,

$V = \{597, 596, 593, 586, 581, 568, 567, 560, 549, 548, 547, 529, 529, 527, 520, 491, 482, 478, 475, 475, 466, 462, 459, 458, 454, 451, 449, 443, 442, 421, 410, 409, 395, 394, 390, 377, 375, 366, 361, 347, 334, 322, 315, 313, 311, 309, 296, 295, 294, 289, 285, 279, 277, 276, 272, 248, 246, 245, 238, 237, 232, 231, 230, 225, 192, 184, 183, 176, 174, 171, 169, 165, 165, 154, 153, 150, 149, 147, 143, 140, 138, 134, 132, 127, 124, 123, 114, 111, 104, 89, 74, 63, 62, 58, 55, 48, 27, 22, 12, 6\}$,

$W = \{54, 183, 106, 82, 30, 58, 71, 166, 117, 190, 90, 191, 205, 128, 110, 89, 63, 6, 140, 86, 30, 91, 156, 31, 70, 199, 142, 98, 178, 16, 140, 31, 24, 197, 101, 73, 169, 73, 92, 159, 71, 102, 144, 151, 27, 131, 209, 164, 177, 177, 129, 146, 17, 53, 164, 146, 43, 170, 180, 171, 130, 183, 5, 113, 207, 57, 13, 163, 20, 63, 12, 24, 9, 42, 6, 109, 170, 108, 46, 69, 43, 175, 81, 5, 34, 146, 148, 114, 160, 174, 156, 82, 47, 126, 102, 83, 58, 34, 21, 14\}$.

The above algorithm is implemented in Matlab. T is the number of iterations, and each algorithm runs independently for each instance.

The GGA algorithm and the HGA algorithm have an iteration number of 500 and a population size of 100, using single-point crossing and single-point variation, in which the crossover probability and the mutation probability are 0.5 and 0.1, respectively. The parameters in GBPSOA and VGPSO are $\omega = 0.8$, $c1 = 1.3$, $c2 = 2.8$, $\Delta t = 300$. In order to verify the efficiency of the MPXOGA algorithm proposed in this paper, we set the number of iterations to 200 to achieve a better solution with fewer runs. The population size is 100, and simulations are performed at a small population size, further confirming the efficiency of the algorithm. The output of the optimal solution is compared with the results of other algorithms in Table 1.

4.2 Numerical Results

According to the results of different algorithms for solving the same example [14], it can be seen from Table 1 that the first example finds the same result because of the smaller scale. Through the experimental results of different scale KP problem examples, it can be seen that with the increase of the size of the KP problem, the performance of the HGA and GBPSOA algorithms deteriorates drastically, and the gap between the performance of the GGA and VGPSO algorithms is larger and larger, so

Table 1. Experimental comparison with other algorithms

Algorithm	Instances 1			Instances 2			Instances 3		
	C/W	Size	T	C/W	Size	T	C/W	Size	T
GBPSOA	1042/878	30	500	3075/997	30	700	24864/6651	50	1000
HGA	1037/874	100	500	3103/1000	100	500	26487/6178	100	500
GGA	1042/878	100	500	3112/1000	100	500	26559/6717	100	500
VGPSO	1042/878	30	200	3103/1000	20	300	26559/6717	50	500
MPXOGA	1042/878	100	200	3112/1000	100	200	26559/6717	100	200

HGA and GBPSOA is not well suited for solving large-scale backpack problems. For the second example, GGA and MPXOGA get the best calculation results, but from the calculation cost, the iteration scale of MPXOGA is small. For the third example, GGA, VGPSO, and MPXOGA all achieve the same optimization results, and the computational cost required by MPXOGA is only about 40% of the other three algorithms. In summary, it can be seen from this set of experiments that the orthogonal genetic algorithm MPXOGA proposed in this paper is fast, efficient and stable.

5 Conclusion

According to the inherent feature of knapsack problem, a multi-parent multi-point crossover operation (MP2X) is proposed, which is implanted with orthogonal experimental design method. MP2X operation can make full use of the inherent information from multiple component of multiple individuals. Orthogonal experimental design is an effective method to solve multi-factor and multi-level experimental problems. Based on MP2X operation and orthogonal design method, a genetic algorithm variant (MPXOGA) is proposed in this paper.

The simulation results on classic knapsack instances show that MPXOGA is better than several solvers, including Hybrid Genetic Algorithm (HGA), Greedy Genetic Algorithm (GGA), Greedy Binary Particle Swarm Optimization Algorithm (GBPSOA) and Very Greedy PSO (VGPSO) in the ability of finding optimal solution, the efficiency and the robustness. MPXOG is more effective for high dimensional problems and the computational cost required by MPXOG is small than others.

In the future, along this line of thinking, we will further explore how to integrate other reasonable heuristic strategies into the genetic algorithm to more effectively solve NP-hard problems such as SAT and TSP.

Acknowledgements. This research is supported by National Natural Science Foundation of China (71772060, 61375066). We will express our awfully thanks to the Swarm Intelligence Research Team of BeiYou University.

References

1. Garey, M.R., Johnson, D.S.: *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Co., New York (1979)
2. Lim, T.Y., Al-Betar, M.A., Khader, A.T.: Taming the 0/1 knapsack problem with monogamous pairs genetic algorithm. *Expert Syst. Appl.* **54**, 241–250 (2016)
3. Patvardhan, C., Bansal, S., Srivastav, A.: Parallel improved quantum inspired evolutionary algorithm to solve large size quadratic knapsack problems. *Swarm Evol. Comput.* **26**, 175–190 (2016)
4. Zhan, Z.H., Li, J., Cao, J., Zhang, J., Chung, H.S.: Multiple populations for multiple objectives: a coevolutionary technique for solving multi-objective optimization problems. *IEEE Trans. Syst. Man* **43**(2), 445–463 (2012)
5. He, Y.C., Liu, K.Q., Zhang, C.J.: Greedy genetic algorithm for solving knapsack problem and its application use. *Comput. Eng. Des.* **28**(11), 2655–2657 (2007)
6. Zhao, X.C., Yang, T.T.: Very greedy particle swarm optimization algorithm for knapsack problem. *Comput. Eng. Appl.* **45**(36), 32–34 (2009)
7. Ye, Y.C., Che, L.X., He, B.: A hybrid particle swarm optimization algorithm for solving 0/1 knapsack problem. *J. Changsha Electr. Power Coll.* **21**(4), 87–90 (2006)
8. Li, Y.L., Zhan, Z.H., Gong, Y.J., Chen, W.N., Zhang, J.: Differential evolution with an evolution path: a deep evolutionary algorithm. *IEEE Trans. Cybern.* **45**(9), 1798–1810 (2015)
9. Leung, Y.W., Wang, Y.P.: An orthogonal genetic algorithm with quantization for global numerical optimization. *IEEE Trans. Evol. Comput.* **5**(1), 41–53 (2001)
10. Yang, Q., Chen, W.N., Li, Y., Chen, C.L., Xu, X.M.: Multimodal estimation of distribution algorithms. *IEEE Trans. Cybern.* **47**(3), 636–650 (2017)
11. Wang, Y., Liu, H., Cai, Z., Zhou, Y.: An orthogonal design based constrained evolutionary optimization algorithm. *Eng. Optimiz.* **39**(6), 715–736 (2007)
12. Montgomery, D.C.: *Design and Analysis of Experiments*, 3rd edn. Wiley, New York (1991)
13. Kalashnikov, V.V., Pérez-Valdés, G.A., Tomasgard, A., Kalashnykova, N.I.: Natural gas cash-out problem: bilevel stochastic optimization approach. *Eur. J. Oper. Res.* **206**(1), 18–33 (2010)
14. Li, H., Jiao, Y.C., Zhang, L.: Orthogonal genetic algorithm for solving quadratic bilevel programming problems. *J. Syst. Eng. Electron.* **21**(5), 763–770 (2010)



Cooperative Co-evolution with Principal Component Analysis for Large Scale Optimization

Guangzhi Xu^{1,2}(✉), Xinchao Zhao², and Rui Li^{1,2}

¹ Automation School, Beijing University of Posts and Telecommunications,
Beijing 100876, China

xgzegw@163.com

² School of Science, Beijing University of Posts and Telecommunications,
Beijing 100876, China

Abstract. This paper attempts to address the problem of large scale optimization and high dimensional optimization using principal component analysis (PCA) strategy with differential evolution (DE) based on Cooperative Co-evolution (CC) framework. Decomposition problem is a major obstacle for large-scale optimization problems. The aim of this paper is to propose effective dimension decomposition method of PCA strategy for capturing the main information among dimensions. PCA strategy can measure most of the contribution information of dimension and uses it for identifying main dimension to guide them to group the most promising subcomponents in CC framework. Then each subcomponents can be solved using an evolutionary optimizer to find the optimum values. The experimental results show that this new technique is more effective than some existing grouping methods.

Keywords: Large-scale optimization · Cooperative co-evolution
Problem decomposition · Principal component analysis

1 Introduction

Evolutionary algorithms (EAs) have widely used to solve many optimization problems and show advantageous performance in a variety of difficult optimization tasks [1–3]. However, the performance of EAs would drop dramatically as the dimensionality of problems is increasing [4, 5]. Many larger scale problems exist in real world, for example, the design of airfoil where thousands of variables are required to represent the complex shape of an aircraft wing [6]. This problem makes a serious challenge to EAs.

There are a number of researchers have proposed approaches for solving large-scale problems. One main approach of solving this problem is to decompose the original large-scale problem into a set of smaller and simpler sub problems, which is called Cooperative Coevolution (CC) proposed by Potter and De Jong [7]. In recent work, the decision variables are divided into k s -dimensional subcomponents each of which is evolved separately in a round-robin fashion (where $k \times s$ equals to the total number of dimensions n) [8]. Random Grouping [9] and Delta Grouping [10] are two major attempts in capturing interacting variables in common subcomponents. Cooperative

Co-evolution with Variable Interaction Learning (CCVIL) [11] is another grouping technique recently proposed. Mohammad propose an automatic decomposition strategy called differential grouping that can uncover the underlying interaction structure of the decision variables and form subcomponents such that the interdependence between them is kept to a minimum [12].

A major difficulty in applying CC is the choice of a promise decomposition strategy. The focus of this paper is to propose a new decomposition method based on cooperative co-evolution framework for large-scale global optimization, using principal component analysis decomposition. The algorithm is denoted as DECC-P, which guides the dimensions to from promising subcomponents according the each of contribution to solution. It extracts the principal information as much as possible among all dimensions of solution and inherits to decompose the dimensions according the contribution. DECC-P provides a novel decomposition mechanism and a new approach to get quality groups in the large scale problems by analyzing the information of the solutions.

The research aspects can be described as follows:

- (1) to provide a theoretical foundation for PCA strategy, which is equally applicable to both traditional and evolutionary optimization algorithms;
- (2) to design an decomposition mechanism and to combine it with CC framework;
- (3) to show the beneficial in solving large-scale global optimization problems with up to 1000 decision variables;

The rest of the paper is organized as follows. In Sect. 2, a review of cooperative co-evolutionary algorithms and principal component analysis is given. In Sect. 3, PCA strategy based on CC framework is proposed. Section 4 outlines the benchmark problems used to evaluate the performance of PCA Strategy. Finally, the performance of the DECC-P algorithm is compared to other state-of-the-art decomposition methods, and then the effectiveness in improving the optimization performance is investigated. Sect. 5 summarizes and concludes the paper.

2 Related Work

2.1 Cooperative Co-evolutionary Algorithms

Cooperative Co-evolution (CC) [7] is proposed as a means of decomposing large-scale problems into smaller sub-problems. Therefore, the decomposed subcomponents can be solved using an evolutionary optimizer in a round-robin fashion. Finally, all solutions to sub-problems are merged to form one global solution for the original problem. However, one critical technique of CC is that how to divide the decision variables more effectively. Literature [12] propose differential grouping algorithm, which can be used to identify and group the interacting variables into common subcomponents. It is mathematics method to divide the decision variables into suitable group with knowledge of the structure of the problem. In this paper, the knowledge is available through using PCA strategy, which focuses on extracting the principal variables and revealing the contribution among the variables on statistically. With this prior knowledge, the decomposition technique divides the variables according the proportion of the main information on statistically.

$$\text{Var}(Y_i) = a_i' \sum a_i, i = 1, 2, \dots, n \tag{3}$$

$$\text{Cov}(Y_i, Y_j) = a_i' \sum a_i, i, j = 1, 2, \dots, n \tag{4}$$

The effect of PCA is to transform a set of original exemplars X_i into several new exemplars which substitute almost all information. Here, Y_1 is called the first principal component. The variance $\text{Var}(Y_i)$ is selected for charactering the amount of information that the principal component carries. For avoiding infinity of $\text{Var}(Y_i)$, an orthogonal restriction on a_i , namely, $a_i a_i' = 1$, is used in this paper. When the first principal component can't reflect enough information of the entire exemplars, another principal component Y_i is used. However, Y_i and Y_j may contain the common information about the entire exemplars, which is not what we want. Therefore, an additional constraint is given as $\text{Cov}(Y_i, Y_j) = 0$. For getting the PCA coefficient vector, this problem with two constraints can be solved by Lagrange multiplier method.

It is obvious that coefficients $a_{11}, a_{21} \dots a_{s1}$ represent the contribution on first dimension from PCA transformation function. The definition of contribution of each dimension is as follows, which can be used to divide the dimensions into different groups based on contribution.

$$c_j = \sum_{i=1}^s |a_{ij}|, j = 1, 2, \dots, n \tag{5}$$

If the value of c_j is large it means that the corresponding dimension j contributes large proportion on all exemplars information in statistics. On the contrary, the small value indicates contribution is little.

Algorithm 1: PCA strategy with CC

```

1 pop ← {1, 2, ..., s}
2 dims ← {1, 2, ..., n}
3 PCA cycles ← {2}
4 groups ← {}
5 for i ← 1 to PCA cycles do
6   aij ← PCA(dims)
7   cj ← ∑|aij|
8   groups ← rank(cj) //grouping stage
9   (best, bestval) ← min(func(pop))
10  for j ← 1 to groups do
11    indicies ← groups [j]
12    subpop ← pop[:, indicies]
13    subpop ← optimizer (best, subpop, FE)
14    pop[:, indicies] ← subpop
15    (best, bestval) ← min(func(pop)) //optimization stage.
16  end for
17 end for

```

3.2 PCA Strategy with CC

This section explains how PCA strategy is used in a CC framework for solving large-scale global optimization problems. Algorithm 1 shows that how the CC framework is applied in the PCA strategy. Note that the algorithm has two major stages: a grouping stage and an optimization stage. During the grouping stage the dimensions are ranked in ascending order by the PCA strategy, and each subcomponent is formed with 100 dimensions according the rank. Note that the dimensions can be divide into 10 sub-components if total dimensions is 1000. In ordering to take better advantage of exemplar information, the second time PCA strategy is used on latter half procedure. In the optimization stage the subcomponents that are formed in the grouping stage are optimized in a round-robin fashion for a predetermined number of cycles. The optimizer function can be any numerical optimization algorithm. In this paper, the SaNSDE [16] is used for subcomponents optimizer, which is short for (DECC-P).

4 Experimental Results and Analysis

In order to evaluate the performance of DECC-P a set of 10 benchmark functions is used in this paper. These benchmark functions were proposed for the IEEE CEC'2010 special session on large-scale global optimization and the associated competition [17]. The CEC'2010 benchmark functions are classified into the five classes, each of them is described as follows:

- (1) separable functions (f1, f3);
- (2) single-group m -nonseparable functions (f4, f5);
- (3) $\frac{n}{2m}$ -group m -nonseparable functions (f10, f11);
- (4) $\frac{n}{m}$ -group m -nonseparable functions (f15, f17);
- (5) nonseparable functions (f19, f20);

where n is the dimensionality of the problem and m is the number of variables in each nonseparable subcomponent. For this research, n and m are set to 1000 and 50, respectively. The subcomponent optimizer used in this paper is SaNSDE, a variant of differential evolution (DE) [18]. SaNSDE self-adapts the crossover rate and the scaling factor of DE. The population size is 50 as suggested in [16]. In the experimental studies, DECC-P is compared with four classical CC variants DECC-DG [12], MLCC [19], DECC-D [10] and DECC-DML [10]. All experimental results are based on 25 independent runs for each algorithm. The maximum number of generations was set to 60000. Table 1 and Fig. 1 summarize the experimental results.

It can be observed from Table 1 and Fig. 1 that these results of DECC-P have the ability to find the superior solutions among the state-of-the-art algorithms for most of the CEC'2010 benchmark functions.

The first class f1 and f3 are separable functions and four methods almost obtain the competitive results. However, it is not necessary to identify the variables as fully separable and placed them into one separable group for ECC-DG. The PCA strategy is more promise for obtaining well performance for f3. It is clear that PCA strategy is efficient and accuracy for separable functions.

For the second class f4 and f5, they are one nonseparable subcomponent with 50 variables. In this class, all five methods almost obtain the same results, especially DECC-P and DECC-DG achieve the best solutions on f4 and f5 respectively. The reason for this is that, the all 50 nonseparable variables were correctly grouped into a common subcomponent for DECC-DG.

For the functions f10 and f11, they contain ten nonseparable subcomponents, each with 50 variables and one separable subcomponent with 500 variables. Both of two benchmarks, especially for f11, DECC-P and DECC-DML achieve significant dominance to other competitors. The performance reliability and robustness of five algorithms are almost the same in functions f10.

The f15 and f17 have 20 nonseparable subcomponents. It can be seen that DECC-P is better than other four algorithms on f15 and achieves promise solution on f17. The DECC-DG has a high reliability and optimization ability on both functions, which stems from accuracy grouping. DECC-P works well on functions f19 and f20 where all the variables interact with each other. The other algorithms are worse than DECC-P in terms of the searching solution for function f19. This because its underlying PCA strategy is grouping the principal variables into same subcomponent. It is clear that DECC-P works efficient in most cases and achieves overall well performance compared with other state-of-the-art algorithms.

Table 1. Comparison of DECC-P with Other Algorithms on the CEC'2010 Benchmark Functions

Functions		DECC-P	DECC-DG	MLCC	DECC-D	DECC-DML
f1	Mean	1.03E-23	5.47E+03	1.53E-27	1.01E-24	1.93E-25
	Sta	1.53E-23	2.02E+04	7.66E-27	1.40E-25	1.86E-25
f3	Mean	8.31E-14	1.67E+01	9.88E-13	1.81E-13	1.18E-13
	Sta	1.82E-14	3.34E-01	3.70E-12	6.68E-15	8.22E-15
f4	Mean	1.00E+12	4.79E+12	9.61E+12	3.99E+12	3.58E+12
	Sta	5.13E+11	1.44E+12	3.43E+12	1.30E+12	1.54E+12
f5	Mean	2.04E+08	1.55E+08	3.84E+08	4.16E+08	2.98E+08
	Sta	9.82E+07	2.17E+07	6.93E+07	1.01E+08	9.31E+07
f10	Mean	2.80E+03	4.52E+03	3.43E+03	1.16E+04	1.25E+04
	Sta	2.29E+02	1.14E+02	8.72E+02	2.68E+03	2.66E+02
f11	Mean	2.13E-13	1.03E+01	1.98E+02	4.76E+01	1.80E-13
	Sta	1.80E-14	1.01E+00	6.98E-01	9.53E+01	9.88E-15
f15	Mean	5.10E+03	5.88E+03	7.11E+03	1.53E+04	1.54E+04
	Sta	3.33E+02	1.03E+02	1.34E+03	3.92E+02	3.59E+02
f17	Mean	1.86E+05	4.01E+04	1.59E+05	9.03E+05	6.54E+06
	Sta	1.02E+04	2.85E+03	1.43E+04	5.28E+04	4.63E+05
f19	Mean	1.71E+06	1.74E+06	1.36E+06	1.33E+07	1.59E+07
	Sta	1.24E+05	9.54E+04	7.35E+04	1.05E+06	1.72E+06
f20	Mean	1.80E+03	4.87E+07	2.05E+03	9.91E+02	9.91E+02
	Sta	1.05E+02	2.27E+07	1.80E+02	2.61E+01	3.51E+01

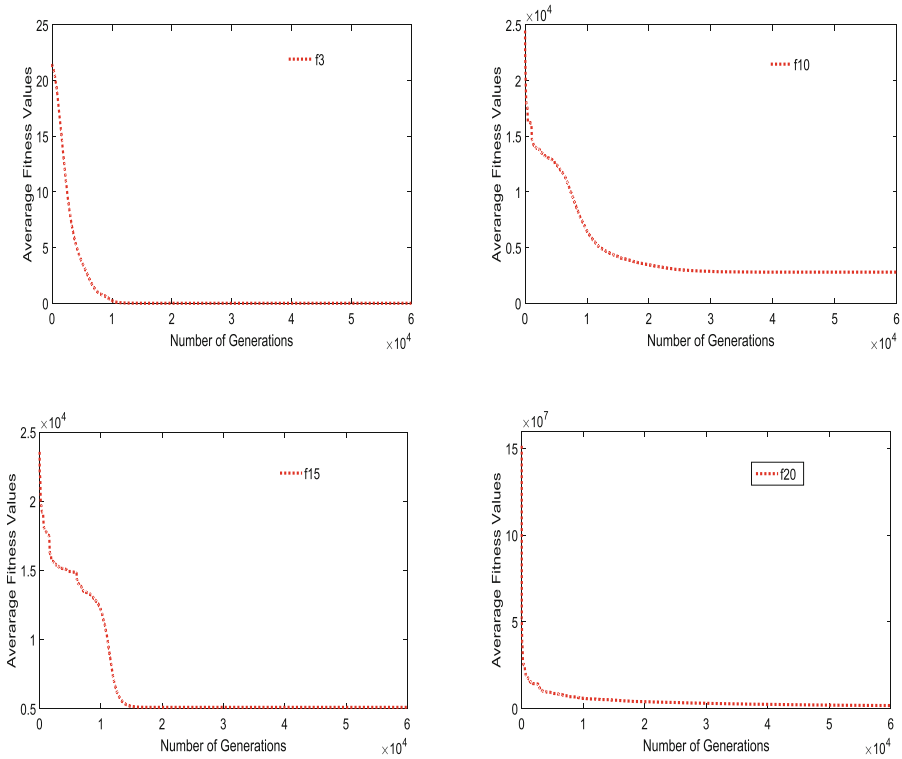


Fig. 1. Convergence curves of DECC-P algorithm on CEC'2010 benchmark functions f3, f10, f15 and f20.

Generally speaking, DECC-P performs significantly better than DECC-DG, MLCC, DECC-D and DECC-DML on functions f3, f4, f10, f15, f19. On functions f1, f5 and f17, DECC-D performed equally well compared with MLCC, DECC-D and DECC-DML.

5 Conclusion

In this paper, we have proposed principal component analysis based on cooperative co-evolution framework to solve large scale optimization problems, which is an automatic way of decomposing an optimization problem into a set of smaller problems.

The proposed decomposition procedure is utilizing the contribution information of variables to divide the dimensions into different groups more effectively. The results have shown that it is capable of PCA strategy for the majority of the benchmark functions. In order to evaluate the performance of DECC-P on optimization problems, a comparative study with DECC-DG, MLCC, DECC-D and DECC-DML algorithms is conducted and the experimental results showed that DECC-P is superior to other algorithms in terms of searching solution for most large-scale functions. However,

distributing reasonable computational resources to each of the subcomponents is the extending study in the future investigations. In addition, the research of high performance evolutionary optimizer is possible to make it scale better to high dimensional problems as a subcomponent optimizer in CC framework with PCA strategy.

References

1. Kazimipour, B., Salehi, B., Jahromi, M.Z.: A novel genetic-based instance selection method: Using a divide and conquer approach. In: 2012 16th CSI International Symposium on Artificial Intelligence and Signal Processing (AISP), pp. 397–402. IEEE (2012)
2. Qin, A.K., Raimondo, F., Forbes, F., Ong, Y.S.: An improved CUDA-based implementation of differential evolution on GPU. In: the 2012 Conference on Genetic and Evolutionary Computation, pp. 991–998 (2012)
3. Peng, H., Wu, Z.: Heterozygous differential evolution with Taguchi local search. *Soft Comput.* **19**(11), 3273–3291 (2015)
4. Mahdavi, S., Shiri, M.E., Rahnamayan, S.: Metaheuristics in largescale global continues optimization: a survey. *Inf. Sci.* **395**, 407–428 (2014)
5. Kazimipour, B., Li, X., Qin, A.K.: Why advanced population initialization techniques perform poorly in high dimension? In: Dick, G., et al. (eds.) SEAL 2014. LNCS, vol. 8886, pp. 479–490. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-13563-2_41
6. Vicini, A., Quagliarella, D.: Airfoil and wing design through hybrid optimization strategies. *AIAA J* **37**, 634–641 (1999)
7. Potter, M.A., De Jong, K.A.: A cooperative coevolutionary approach to function optimization. In: Davidor, Y., Schwefel, H.-P., Männer, R. (eds.) PPSN 1994. LNCS, vol. 866, pp. 249–257. Springer, Heidelberg (1994). https://doi.org/10.1007/3-540-58484-6_269
8. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Network, vol. 4, pp. 1942–1948 (1995)
9. Yang, Z., Tang, K., Yao, X.: Large scale evolutionary optimization using cooperative coevolution. *Inf. Sci.* **178**, 2986–2999 (2008)
10. Omidvar, M.N., Li, X., Yao, X.: Cooperative co-evolution with delta grouping for large scale non-separable function optimization. In: Proceedings of IEEE Congress on Evolutionary Computation, pp. 1762–1769, July 2010
11. Chen, W., Weise, T., Yang, Z., Tang, K.: Large-scale global optimization using cooperative coevolution with variable interaction learning. In: Schaefer, R., Cotta, C., Kołodziej, J., Rudolph, G. (eds.) PPSN 2010 Part II. LNCS, vol. 6239, pp. 300–309. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15871-1_31
12. Omidvar, M.N., Li, X., Mei, Y., et al.: Cooperative co-evolution with differential grouping for large scale optimization. *IEEE Trans. Evol. Comput.* **18**(3), 378–393 (2014)
13. Xie, J., Chen, W., Zhang, D., et al.: Application of principal component analysis in weighted stacking of seismic data. *IEEE Geosci. Remote Sens. Lett.* **14**(8), 1213–1217 (2017)
14. Chu, W., Gao, X.G., Sorooshian, S.: Fortify particle swam optimizer (PSO) with principal components analysis. In: 2011 IEEE Congress on Evolutionary Computation, pp. 1644–1648 (2011)
15. Kuznetsova, A., Pons-Moll, G., Rosenhahn, B.: PCA-enhanced stochastic optimization methods. In: Pinz, A., Pock, T., Bischof, H., Leberl, F. (eds.) DAGM/OAGM 2012. LNCS, vol. 7476, pp. 377–386. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32717-9_38

16. Yang, Z., Tang, K., Yao, X.: Self-adaptive differential evolution with neighborhood search. In: *Evolutionary Computation*, pp. 1110–1116. IEEE (2008)
17. Tang, K., Li, X., Suganthan, P.N., Yang, Z., Weise, T.: Benchmark functions for the CEC 2010 special session and competition on large-scale global optimization. *Nature Inspired Computation and Applications Laboratory* (2009)
18. Zhang, S.X., Zheng, S.Y., Zheng, L.M.: An efficient multiple variants coordination framework for differential evolution. *IEEE Trans. Cybern.* **47**(9), 2780–2793 (2017)
19. Yang, Z., Tang, K., Yao, X.: Multilevel cooperative coevolution for large scale optimization. In: *IEEE Congress on Computational Intelligence*, pp. 1663–1670. IEEE (2008)



HCO-Based RFID Network Planning

Jun Wang¹, Jinsong Chen², Qianying Liu³(✉), and Jia Liu⁴

¹ Network Engineering, South China Normal University,
Guangzhou 510630, China

² Department of Information Management,
National Yunlin University of Science and Technology, Douliu,
Yunlin 64002, Taiwan

³ College of Management, Shenzhen University, Shenzhen 518060, China
Qianying_Liu@163.com

⁴ Department of Computer Science, University of Surrey,
Guilford GU2 7XH, UK

Abstract. This work presents the application of Hydrologic Cycle Optimization (HCO) for RFID network planning (RNP). An integrated model is presented in this paper to evaluate the RNP's fitness which lays emphasis on coverage, load balance, interference and economic efficiency of the RFID system. The fitness function based on this integrated model uses the power of the tag received from every reader replacing the previous one—distance to calculate the coverage and interference. This substitute makes our model accurately reflects the real situation. HCO algorithm is used to find the optimal position and power of the readers with the minimum value of the fitness function based on the model above. The solution of RNP is optimized by searching for the best value of the parameters (position and power) which are mathematically denoted as a vector whose length is $3N$ (N is the readers' count). The encoding of this vector consists of the coordinates of each reader and their radiated power. The first $2N$ length is the coordinates of each reader, and the rest is their power. In the proceeding of finding the optimal position and power, the four factors mentioned above are considered and the best individual will be tracked. To demonstrate the effectiveness and efficiency of HCO, we make a comparison among HCO, PSO, GA, SA-ES. As the result indicates, the HCO algorithms has the best performance of RNP among all the algorithms both the best and the worst situation.

Keywords: Hydrologic Cycle Optimization · Swarm Intelligence
RFID network planning

1 Introduction

The Radio Frequency Identification (RFID) technology is a famous wireless application which has been widely used in many fields such as ticketing, payment, passports, car keys, etc. [1, 2] For example, Ni et al. [3] use the active RFID to handle the indoor location sensing problem. Pala et al. [4] create a smart parking system based on the RFID.

A complete RFID system consists of readers and transponders (so-called tag). The principle of RFID system is that the reader sends a certain frequency of infinite wave energy to the transponder, driving its circuit to send the internal ID code which will be received by the reader later. The advantages of the transponder are that it is free of battery, no contact, no card, so it is not afraid of dirt, and the chip password is the only one in the world that cannot be copied, with high security and long life [1].

For the purpose of managing the large-scale network of RFID readers in an optimal fashion, the RFID network planning (RNP), as a vital problem should be addressed when deploys the RFID application [5]. However, due to the fact that it has lots of objectives (coverage, load balance, interference, etc.), RNP is a sophisticated and challenging problem.

In the past few years, many researchers used population-based algorithms to solve optimization problems. Among these algorithms, the representative ones are Genetic Algorithm (GA) [6], Genetic Programming (GP) [7], Swarm Intelligence algorithms inspired by animal colonies (Particle Swarm Optimization (PSO) [8], Artificial Bee Colony (ABC) algorithm [9], etc.) However, their performance is not very well when the optimization problems are complex [10]. As to RNP, in [11], an improved PSO is utilized to cope with RNP by Gong et al. Ma et al. [12] proposed a cooperative multi-objective artificial colony algorithm and exploited it to gain the solution of this problem. In addition, Guan et al. [13] proved the applicability of the Genetic approach (GA) based RNP.

Hydrologic cycle optimization (HCO) is a novel algorithm, which simulates the water cycle on Earth. It has been demonstrated its superiority in some problems compared to the PSO, ABC, etc. [10] In this paper, aiming to make the deployment of RFID systems more reasonable and economical, we combine the model in [14] with [15] and propose an integrated model as the fitness function, which pays more attention to coverage, load balance, interference and economic efficiency of the RFID system. Due to the fact that the receiving quality actually directly depends on the radiated power of reader, this integrated model use power to calculate the interference (proposed by Chen et al.), replacing the relevant part in [14], which makes this model more accurate to reflect the reality in evaluating the RNP. In this study, the solution of RNP based on the integrated model is encoded to a vector (so-called water individual). The population consists of these water individuals will be updated by the HCO algorithm which includes four operations—flow, infiltration, evaporation, and precipitation after multi iterations. And we evaluate the position of the population and track the best water individual in every iteration. In the experimental section, the experimental results are compared with Particle Swarm Optimization (PSO), Genetic Algorithm (GA) and the Self-adaptive ES (SA-ES). From the experimental results, we can see that the HCO outperforms other algorithms and have higher efficiency. In other words, the HCO algorithms consume less time while getting better performance. The detailed experimental data and results are given in Sect. 5.

The remainder of this paper is organized as follows. The RNP problem model is introduced in Sect. 2. Sections 3 and 4 give the brief introduction of the HCO algorithm and its application for RNP. Section 5 is the experiment part which compares the performance of HCO, PSO, GA and SA-ES, in solving the RNP. Finally, we draw the conclusion in Sect. 6.

2 RFID Network Planning (RNP) Problem

The wireless communication between readers and tags holds the key to the RFID system [13]. There are many factors to affect the deployment of the readers. Users always are expected to obtain better services, which is one of the most important factors to affect users making decisions. However, as a service provider, they pay more attention to cost-benefit ratio. Meanwhile, they should guarantee the service quality. In this paper, we comprehensively consider the two side and propose an integrated model based on the model proposed in [14] and [15]. The following Eqs. (1), (3), (4) are the same as [14], and (2) comes from [15]. This model attaches more attention to the coverage, load balance, interference and economical efficiency. The coverage factors can guarantee the availability of the users. To minimize the interference among readers, the service quality will be enhanced. To the service provider, a good load balance in the system will extend machine life and save energy, cutting the cost in some way. The purpose of the economical efficiency is to cost less money to provide better performance. In many models, distance between the reader and tag is used to calculate the above objectives. However, the quality of the information received by the tag from the reader and the distance are not in the same series, which may account for inaccuracy when we evaluate every water individual. Therefore, in our integrated model, we use the power of the tag received by every reader to calculate the coverage and interference because the received power determines the quality of the received information. The variable declaration of the mathematical model shows in Table 1 [14].

2.1 Network Coverage

In most RFID network applications, coverage is often the most important part among the whole objectives. In most cases, full coverage, which requires all tag being received by the readers, is usually a hard requirement. We use the difference between the minimum power required for the communication between reader and tag P_d (in this

Table 1. Variable declaration.

Variable	Declaration
M	The number of readers
P_i	The power level of i^{th} reader
P_k	The minimum power required for the communication between reader and tag
P_{best}	The biggest power the tag received from all the reader
P_d	The threshold power of tags
R_i	The position of i^{th} reader
C_k	The assigned tags number to reader k
N_t	The number of tags
$center_k$	The position of k^{th} cluster center
R_k	The position of k^{th} best served reader
f_i	The function for the i^{th} objective
w_i	The weight of the i^{th} objective

study is -15 dBm) and its best-received power P_i of i_{th} tag covered by reader R_b as a measurement of coverage. Equations (1), (3)–(5) is from reference [14]. Equation (2) is from reference [15].

$$f_1 = \sum_{i=1}^{N_r} (P_i^r - P_d) \quad (1)$$

2.2 Interference

Interference is that there are overlaps coverage between readers. In other words, the tag can receive power from at least two readers. However, these power received from other readers exerts a negative influence on the tag. Therefore, we use the power to evaluate the interference level.

$$f_2 = \left(\sum_{k=1}^M P_k \right) - P_{best} \quad (2)$$

2.3 Load Balance

It is a great energy waste if some readers are under high load while others are low. What's more, readers which are always under high load are prone to go wrong. Accordingly, load balance also plays an important part in the deployment of the RFID system. This objective function is mainly defined by the number of tags served by the k^{th} reader.

$$f_3 = \prod_{k=1}^M \left(\frac{1}{C_k} \right) \quad (3)$$

2.4 Economic Efficiency

Service providers always pay more attention to the economic efficiency to cut the costs. Therefore, we also take the economic efficiency into consideration. It's a useful way to denote the economic efficiency by reducing the distance between readers and tags.

$$f_4 = \sum_{k=1}^M (dist(R_k - center_k)) \quad (4)$$

2.5 Combined Measure

In this study, the final fitness function for RNP consists of these four objectives.

$$f_5 = \sum_{i=1}^4 w_i f_i; w_1 + w_2 + w_3 + w_4 = 1 \quad (5)$$

3 HCO Algorithm

This algorithm simulates the hydrologic cycle action on Earth. water on the earth move to different places through a series of physical effects, but the total amounts of water almost has no change [16]. These physical effects include flow, infiltration, evaporation and precipitation. Flow step allow water to flow to the lower places. The function of infiltration is to enable water to move towards its neighbors. In the evaporation and precipitation steps, water move to other places in a wide range. Hydrologic cycle algorithm simulates the proceeding of hydrologic cycle action, taking the above four factors into consideration [14]. The detailed introduction is given as follow, Tables 2, 3 and 4 is from reference [10].

3.1 Flow

The reason why water can move and gather to lower areas mainly relies on this step. In the HCO algorithm, each individual randomly chooses another individual whose position is better than it to move. The pseudocode of this parts is defined as follow:

Table 2. The pseudocode of flow step in HCO

For each individual X_i in the population
Select another individual X_j which its fitness is better than X_i randomly
$X_{try} = X_i + (X_j - X_i) * rand(1, Dimension)$
While the new position is better && max flow times not reached
$X_i = X_{try}$
Flow another time using the equation above
End
End

3.2 Infiltration

Most of the population based algorithm are likely to fall into the local optima. In this step, water individual moves towards its neighbors, even if its neighbor's position is higher than it [10]. This make it possible for this algorithm to exploit the neighborhood and escape from the local optima. The pseudocode of this parts is defined as follow:

Table 3. The pseudocode of Infiltration step in HCO

For each individual X_i in the population
Select another individual X_j randomly
Select sd dimensions randomly to form a vector SD
$X_{i,SD} = X_{i,SD} + (X_{j,SD} - X_{i,SD}) * 2 * (rand(1, sd) - 0.5)$
End

3.3 Evaporation and Precipitation

Evaporation and precipitation enable the water to move to other places in a wide range, even the higher places. In this step, individuals are probably evaporated with the possibility P_{eva} . Then, the evaporated water individual then takes the precipitation action and has two destinations to choose—a random position in the searching space or the best position of its neighborhood. The pseudocode of this parts is defined as follow:

Table 4. The pseudocode of Evaporation and precipitation step in HCO

For each individual X_i in the population
IF $rand < P_{eva}$.
IF $rand < 0.5$
Move X_i to another position randomly
Else
Move X_i to neighborhood of best position so far using Gauss mutation
End
End
End

4 HCO Algorithm for RNP Problem

The integrated model mentioned above mathematically denotes RNP. Then, by the above four operations, the HCO algorithm update the position of the population (the solutions) in every iteration and enable population to gather to the better position. We track the best water individual (solution) in every iteration and obtain the global best position in the end of the program. The calculation methods of these objectives have been mentioned above. The vital elements in this part are displayed as follow. Equation (6) is from reference [11].

4.1 Encoding

In RNP, the position and energy of each reader need to be mathematically represented for optimization. In our experiments, each water individual is represented by a vector of dimension $D = 3M$, where M is the number of readers. [8] The first $2M$ dimensions are used to represent the coordinates of each reader, and the last M dimensions are used to represent the energy of each reader. Its formula is as follows:

$$X_i = [x_i^1, y_i^1, x_i^2, y_i^2, \dots, x_i^M, y_i^M, p_i^1, p_i^2, \dots, p_i^M] \quad (6)$$

4.2 Initialization

At the very beginning to implement the program, we need to initialize some of the required parameters. First, we randomly generate 100 coordinates in a 30 by 30

two-dimensional space, which will simulate the position of the tag in reality. [12] Then, the necessary parameters for HCO operation, such as the population size, the number of iterations, the maximum flow time, etc., need to be initialized. The relevant values and results are shown in the V section.

4.3 Fitness Evaluation

The fitness function is shown in (5). Our goal is to minimize the value of the function by the HCO algorithm. However, the method to calculate the coverage value (as shown in (1)) is a maximization problem. In order to turn it into a minimization problem, we convert the energy to its negative. Load balance, economic efficiency, and interference calculation methods are shown in (2), (3) and (4). The specific steps for each water individual to be evaluated are as follows:

- (a) Firstly, we are supposed to set the value of w_1, w_2, w_3, w_4 , etc., and separate the coordinates and power of readers from population.
- (b) Calculate the distance between all the tags with all the readers, the power every tag received from every reader. In this part, to change the problem to a minimization problem, the power is converted to negative.
- (c) Calculate the distance between all the reader with $center_k$.
- (d) Calculate the f_1, f_2, f_3, f_4, f_5 by (1), (2), (3), (4), (5).

4.4 The Main Function

This function is to solve the RNP by HCO algorithms, optimizing the fitness function and saving the results. Therefore, it is the most important part of this study. Firstly, we should initialize the relevant parameters. Then, the HCO algorithms runs to optimize the fitness function. The calculation step of the HCO is listed in Tables 2, 3, 4. Meanwhile, we should take the records of the best fitness value and the index of the best water individual in every iteration. Finally, we should save the results, including the distributed sketch of the reader and the best fitness value in every run. The pseudocode of this part is given in Table 5.

5 Experiments and Results

In this session, a series of computation is carried out to test the characteristics of HCO. In this experiment, based on the model in [14], the working area (shown in Fig. 1) including 100 tags takes up an area of 30 m by 30 m. In addition, with the radiated power varying from 0.1 to 2 W, 9 readers are deployed to serve them.

In this paper, we make a comparison of the performance of RNP among HCO, PSO, GA, and SA-ES. The control parameters of the GA and the SA-ES are same as [15]. The parameters of HCO are listed in Table 5. Table 6 shows the performance of all algorithms on RNP. The objects we comparing include the best value, the worst value, the average value and the average running time. As the Figs. 2 and 3 indicate, it is obvious that the HCO algorithm outperforms other algorithms. From Table 7, we can

Table 5. The pseudocode of the main function

Initialization.
 Record the current time.
 Set parameters $Dim, PopSize, MaxFlowTimes, M, N_t, w1, w2, w3, w4, etc.$
 Deploy the tags and reader in the working area randomly.
 Get the initial Population randomly and calculate its fitness, the record the $Fvalue_{best}$ and $Individual_{best}$ using the equation (5) and (6).
End Initialization

While $l < Iterations$
For ($r=1:S$)
 Execute the Flow operation using the algorithm in Table 1.
 Update the $Fvalue_{best}$ and $Individual_{best}$ using the equation (5).
End FOR

For($r=1:S$)
 Execute the Infiltration operation using algorithm in Table 2.
 Update the $Fvalue_{best}$ and $Individual_{best}$ using the equation (5).
End For

For($r=1:S$)
 Execute the Evaporation and precipitation operation using the algorithm in Table 3.
 Update the $Fvalue_{best}$ and $Individual_{best}$ using the equation (5).
End For

Save the $Fvalue_{best}$ of this iteration into result vector. Using the equation as follow:
 $result(l) = Fvalue_{best}$

End While
 Calculate the time this program consumes by the time we have recorded before.
 Form the position and field strength picture of readers by the $Individual_{best}$ we have recorded before.

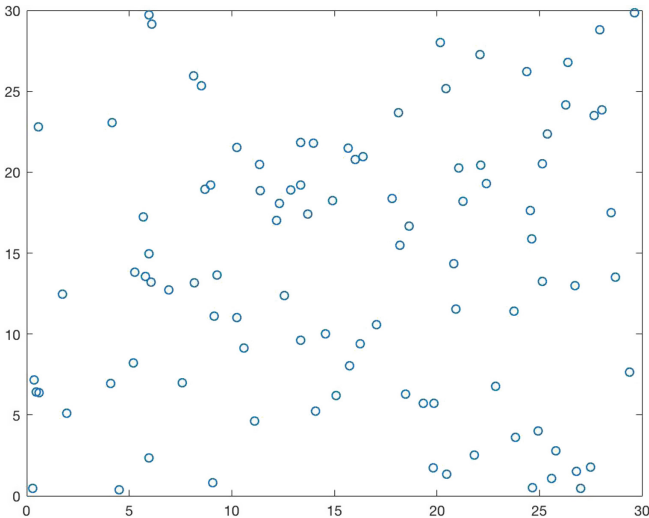


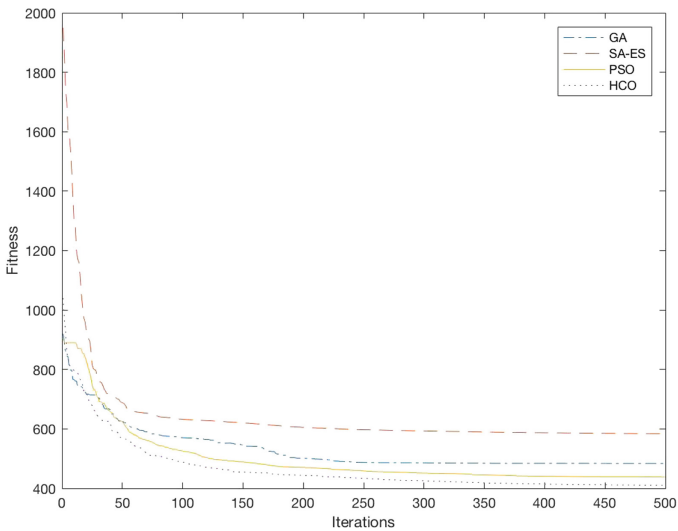
Fig. 1. The working area

Table 6. Parameters of HCO

Parameters	Value
Dim	27
M	9
N_i	100
P_{ed}	0.25
P_d	-15
$CoordinateRange$	(0, 0)–(30, 30)
$PopSize$	10
$Iterations$	500
$Runtime$	5
w_1	0.3
w_2	0.2
w_3	0.2
w_4	0.3

Table 7. Performance of all algorithms

Objective function	HCO	PSO	SA-ES	GA
Best	356.8600	404.4226	496.7077	400.3008
Worst	411.8724	502.7327	751.8296	592.5708
Mean	380.2991	437.8498	583.8184	483.7341
Running time (s)	43.2658	51.3603	7.6880	73.2095

**Fig. 2.** The mean fitness value of all algorithms over five runs

also see that the HCO algorithm is the best in both the best and the worst situation. In addition, HCO spends the second least time in getting the best performance. Although the SA-ES spends least time, it is the worst among all the algorithms.

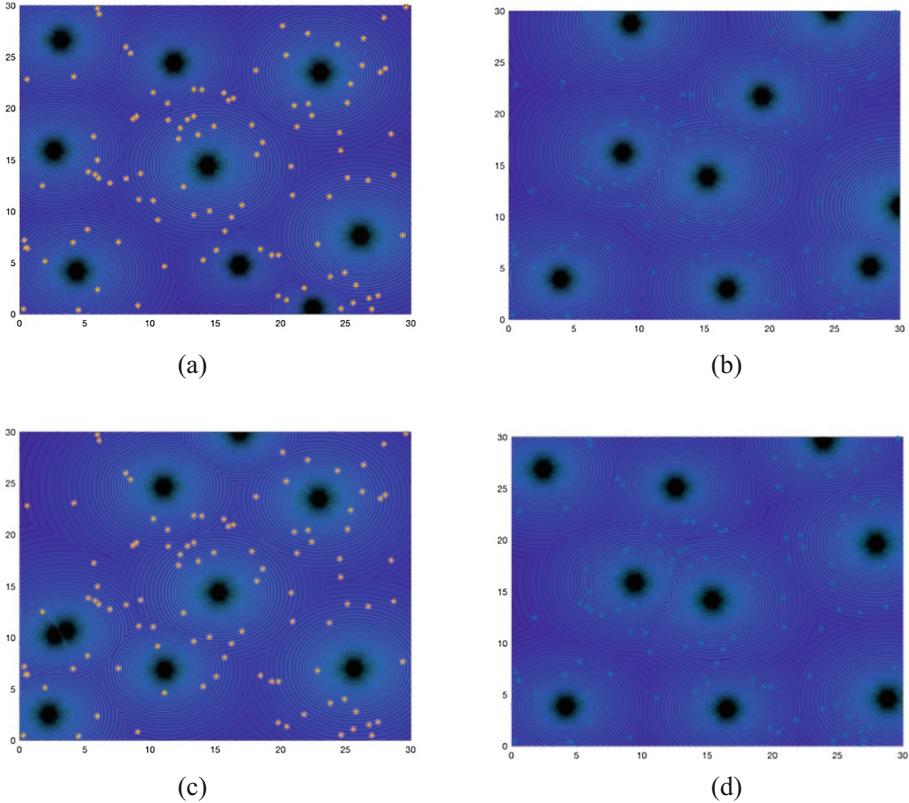


Fig. 3. The position and its radiated range of readers (a): HCO, (b): PSO, (c): SA-ES, (d): GA

6 Conclusions

In this paper, an integrated model which combines the model in [14] with [15] is proposed to denote the RNP as a mathematical form. In this model, we take the coverage, load balance, interference and economical efficiency into consideration. To reflect the real situation more accurately, we use power to calculate the coverage and interference. The HCO algorithm is utilized to optimize RFID network planning. This algorithm simulates the water cycle on Earth and doesn't need many parameters. In the experimental part, we take a comparison of the performance of RNP among HCO, PSO, GA and SA-ES algorithm. The experimental results demonstrate the effectiveness and efficiency of HCO algorithm. In the future work, we will study the hybrid model which combines the HCO with other algorithms.

References

1. Duroc, Y., Tedjini, S.: A key technology for Humanity. *Compt. Rend. Phys.* **19**(1), 64–71 (2018)
2. Ma, L., Chen, H., Hu, K., et al.: Hierarchical artificial bee colony algorithm for RFID network planning optimization. *Sci. World J.* **2014**, 1–22 (2014)
3. Ni, L.M., Liu, Y., Lau, Y.C.: LANDMARC: indoor location sensing using active. In: *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*, pp. 407–415. IEEE (2003)
4. Pala, Z., Inanc, N.: Smart parking applications using RFID technology. In: *1st Annual RFID Eurasia*, pp. 1–3. IEEE (2007)
5. Chen, H., Zhu, Y., Hu, K.: Network planning using a multi-swarm optimizer. *J. Netw. Comput. Appl.* **34**(3), 888–901 (2011)
6. Holland, J.: Genetic algorithms. *Sci. Am.* **267**(1), 66–72 (1992)
7. Koza, J.R., Poli, R.: Genetic programming. In: Burke, E.K., Kendall, G. (eds.) *Search Methodologies*. Springer, Boston (2005). https://doi.org/10.1007/0-387-28356-0_5
8. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: *Proceedings of the 1995 IEEE International Conference on Neural Networks*, pp. 1942–1948. IEEE (1995)
9. Dervis, K., Bahriye, A.: A comparative study of artificial bee colony algorithm. *Appl. Math. Comput.* **214**(1), 108–132 (2009)
10. Yan, X., Niu, B.: Hydrologic cycle optimization part I: background and theory. In: Tan, Y., Shi, Y., Tang, Q. (eds.) *ICSI 2018 Part I. LNCS*, vol. 10941, pp. 341–349. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93815-8_33
11. Gong, Y.J., Shen, M., Zhang, J., et al.: Optimizing RFID network planning by using a particle swarm optimization algorithm with redundant reader elimination. *IEEE Trans. Industr. Inf.* **8**(4), 900–912 (2012)
12. Ma, L., Hu, K., Zhu, Y., et al.: Cooperative artificial bee colony algorithm for multi-objective RFID network planning. *J. Netw. Comput. Appl.* **42**, 143–162 (2014)
13. Guan, Q., Liu, Y., Yang, Y.: Genetic approach for network planning in the RFID systems. In: *Sixth International Conference on Intelligent Systems Design and Applications*, pp. 567–572. IEEE (2006)
14. Gu, Q., Yin, K., Niu, B., Chen, H.: RFID networks planning using BF-PSO. In: Huang, D. S., Ma, J., Jo, K.H., Gromiha, M.M. (eds.) *Intelligent Computing Theories and Applications, ICIC 2012, LNCS*, vol. 7390, pp. 181–188. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31576-3_24
15. Chen, H.N., Zhu, Y.L., Hu, K.Y.: Networks planning using a multi-swarm optimizer. *J. PLA Univ. Sci. Technol. Nat. Sci. Edit.* **9**(5), 413–416 (2008)
16. Schlesinger, W.H., Bernhardt, E.S.: *The global water cycle*. In: *Biogeochemistry*, 3rd edn. Academic Press (2013)



Cuckoo Search Algorithm Based on Individual Knowledge Learning

Juan Li^{1,2}, Yuan-Xiang Li^{1(✉)}, and Jie Zou²

¹ School of Computer, Wuhan University, Wuhan 430072, China
yxli@whu.edu.cn

² School of Information Engineering,
Wuhan Technology and Business University, Wuhan 430065, China

Abstract. Cuckoo search (CS) is a one of the most efficient evolutionary for global optimization, and widely applied to solve diverse real-world problems. Despite its efficiency and wide use, CS suffers from premature convergence and poor balance between exploitation and exploration. To cope with these issues, a new CS extension based on individual knowledge learning (IKL-CS) is proposed. In this study, knowledge learning based on individual history is introduced into the CS algorithm. Individuals are constantly adjusted and optimized to use their historical knowledge in the optimization process, and communicate with each other to use their own knowledge. The accuracy and performance of the proposed approach are evaluated by eighteen classic benchmark functions. Statistical comparisons of our experimental results showed that the proposed IKL-CS algorithm made an appropriate trade-off between exploration and exploitation. Comparing the proposed I-PKL-CS with various evolutionary CS algorithms, the results demonstrated that IKL-CS is a competitive new type of algorithm.

Keywords: Cuckoo search algorithm · Individual knowledge
Global optimization

1 Introduction

Metaheuristic algorithms [1] have been used extensively to solve complex and highly non-linear optimization problems. Most of these methods were inspired by natural or physical processes [2, 3], and include genetic algorithms (GAS) [4], particle swarm optimization (PSO) [5], differential evolution (DE) [6, 7], krill herd (KH) [8–15], ant colony optimization (ACO) [16], biogeography-based optimization (BBO) [17], artificial bee colony (ABC) [18], elephant herding optimization (EHO) [19], monarch butterfly optimization (MBO) [20–22], earthworm optimization algorithm (EWA) [23], Brain storm optimization (BSO) [24], moth search (MS) algorithm [25] and the cuckoo search algorithm (CS) [26–29].

The CS algorithm is a simple and effective global optimization algorithm developed by Yang and Deb [30]. CS combines the idea of obligate brood parasitism, as exemplified by some cuckoo species, with Lévy flights. CS can explore a search space better than algorithms based on uniform and Gaussian distributions. CS has achieved good

performance on many optimization problems, and has been applied successfully to diverse fields such as electrical power systems [31], software engineering [32].

Yang and Deb [33] compared CS with PSO in over 100 trials for selected objective functions. This number of objective function evaluations would not be feasible for practical engineering problems that have costly objective functions.

Chandrasekaran [34] successfully applied CS to multi-objective scheduling problems. An improved version of CS was applied by Valian et al. [35] to reliability optimization problems. They proposed a modified cuckoo search algorithm that involved the addition of information between the top eggs.

Under certain conditions, CS may not have the ability to escape from local optima. To overcome this drawback and enhance the search ability of CS algorithms, a strategy was provided by [36] that adjusted the parameters using a self-adaptive method.

Li et al. [37] enhanced the exploitation ability of the cuckoo search algorithm by using an orthogonal learning strategy. Their experimental results showed that the method was very effective for the estimation of chaotic systems and for continuous function optimization problems.

Agrawal et al. [38] found the optimal thresholds that used the cuckoo search algorithm for multi-level thresholds in an image. An improved discrete version of CS was presented by Ouaarab et al. [39] to solve the famous traveling salesman problem.

It should be noted, however, that these methods were improved only with regard to some local aspects, and they could face issues arising from local optima and slow convergence speeds. Overall, researchers have encountered difficulties achieving the balance between exploration and exploitation.

In this paper, we present an improved CS algorithm called IKL-CS that adopts self-adaptive learning strategies. The previous algorithm focused only on the best individual for history individuals, ignoring ordinary individuals in the population, thus could not use useful information for all individuals, which is easy to be trapped by local optima. This research differs from other similar work insofar as the advantage of learning based on individual historical knowledge is that it not only considers the historical optimal solution of individual, but also considers all history solutions of individual, and adjusts their weights self-adaptively through knowledge learning rate and knowledge potential. We proposed learning model based on individual history knowledge (I-KL). Because of the previous algorithm focused only on the best individual in history individuals, useful information of all individuals could not be used. In I-KL model, through the learning of historical knowledge experience, the new position of individual is determined by knowledge learning rate and potential energy, which might improve the local search ability of the algorithm.

2 Cuckoo Search (CS)

The cuckoo search algorithm is an evolutionary algorithm inspired by the obligate brood parasitism of some cuckoo species that lay their eggs in the nests of other host birds. The algorithm is created by combining a model of this behavior with the prin-

ciples of Lévy flights. In the cuckoo search process, nest location is initialized randomly in feasible solution space, and the fitness of each nest location is calculated. CS is based on three idealized rules:

- (1) Each cuckoo lays one egg at a time, and places it in a randomly chosen nest.
- (2) The best nests with the highest quality eggs (solutions) will be carried over to the next generations.
- (3) The number of available host nests is fixed, and a host can discover an alien egg with the probability $P_a \in [0, 1]$. If the alien egg is discovered, the nest is abandoned, and a new nest is built in a new location.

Yang used D -dimensional vector $X_i = (x_{i,1}, x_{i,1}, \dots, x_{i,D})$, $1 \leq i \leq n$ to indicate the position of the number i nest. Yang used Lévy flights (based on random walks) to produce offspring. A Lévy flight is performed as follows:

$$X_i^{t+1} = x_i^t + a \otimes \text{levy}(\lambda) \quad (i = 1, 2, \dots, n), \quad (1)$$

$$a = a_0 \otimes (x_j^t - x_i^t), \quad (2)$$

after partial solutions are discarded, a new solution with the same number of cuckoos is generated by using Eq. (3).

$$X_i^{t+1} = x_i^t + r(X_m^t - X_n^t), \quad (3)$$

where r generates a random number between -1 and 1 , X_m^t and X_n^t are random solutions for generation t .

3 Learning Model Based on Individual Knowledge (I-KL)

Knowledge learning refers to the process in which individuals in a group communicate with each other and share their knowledge with other members. The knowledge learning proposed in this paper includes two aspects: the individual learning of its historical knowledge and the learning of population knowledge. The principle of learning based on individual historical knowledge is to study individual historical experience knowledge in each generation so as to obtain a better location of the nest. Learning based on individual historical knowledge can be considered as a decision process which is a high-level cognitive process. The advantage of learning based on individual historical knowledge is that it not only considers the historical optimal solution of individual, but also considers all history solutions of individual, and adjusts their weights self-adaptively through knowledge learning rate and knowledge potential.

The knowledge acquired through the historical learning is used to guide the evolution of subsequent individuals in order to better explore new areas.

The purpose of individual interaction is to determine the useful knowledge by the individual or group, so as to fully transfer, share, and utilize knowledge. When individuals learn or make decisions, they are more likely to refer to the people with abundant knowledge and to be influenced by the closer knowledge experience. It is common knowledge that different individuals have different levels of knowledge. For a same cuckoo nest, different iterations have different levels of knowledge. Therefore, the concept of knowledge potential or knowledge level can be defined as follows.

$$\varpi[s + \Delta s] = \varpi[s] + \Delta s \tag{4}$$

where $\varpi[s]$ is the knowledge potential energy of decision-making individuals, Δs is levels of knowledge. The formula shows that, with the growth of learning time, the knowledge potential of the decision individual is increased, the knowledge level is related to the original knowledge and external knowledge. The knowledge potential of the decision individual is influenced by the interaction learning, the longer the interaction, the higher the potential of knowledge. In addition, when the decision-makers have the potential difference, the individuals with low knowledge individuals are more likely to be affected by the individuals with higher potential energy, and the influence size can be represented by the learning rate. For cuckoo individuals, the knowledge potential energy can be expressed by the fitness value of its position. The knowledge learning rate is defined as follows: Let x_j^t is the position of cuckoo individual j in the t -th generation, the corresponding knowledge level is $f(x_j^t)$. The following vector to define knowledge: set the location of the first cuckoo individual j t generation that the corresponding level of knowledge. In other words, the adaptive value of individual is better than that another one, and its knowledge potential is also larger. Conversely, the knowledge potential of the cuckoo is smaller. The knowledge learning rate δ_i^t is definite in Eq. (5).

$$\delta_i^t = \frac{e^{score_i^t}}{e^{score_1^t} + e^{score_2^t} + \dots + e^{score_m^t}} \tag{5}$$

$$score_j^t = \begin{cases} 1, & \text{if } (f_{worst} = f_{best}) \\ \frac{f_{worst}^t - f_j^t}{f_{worst}^t - f_{best}^t}, & \text{others} \end{cases} \tag{6}$$

where δ_i^t is the knowledge learning rate of individual i in the t -th generation. f_j^t is the fitness value of individual j in the t -th generation. f_{worst}^t and f_{best}^t represent the optimal and worst fitness values for generation t , respectively. m is the number of cuckoo nests. It can be seen that the better the fitness value of individual is, the greater its knowledge potential $Score$ is. Conversely, the worse the fitness value of individual is, the smaller its knowledge potential $Score$ is. An example of knowledge potential and knowledge level, where $Score_{x_1}$ represents the knowledge potential of individual x_1 , $Score_{x_2}$ represents the knowledge potential of individual x_2 , f_{x_1} indicates is the fitness value of individual x_1 , f_{x_2} indicates is the fitness value of individual x_2 . It can be seen that the

fitness value f_{x_1} of the individual x_2 is smaller than the fitness value f_{x_2} of the individual x_2 , and the knowledge potential energy $Score_{x_1}$ of the individual x_1 is higher than the knowledge potential energy $Score_{x_2}$ of the individual x_2 . Thus, in the cuckoo algorithm based on individual historical knowledge, the new position determined by historical knowledge is as follows.

$$p_i^t = \sum_{j=1}^t \delta_i^{t-j} x_i^{t-j} \tag{7}$$

where p_i^t is new position determined of individual i by historical knowledge for generation t , δ_i^{t-i} is the knowledge learning rate of individual i for generation $t - i$, f_j^{t-i} is the fitness value of individual j for generation $t - i$.

The standard cuckoo search algorithm only pays attention to the best position in history, ignores the history of general individuals, which cannot use all useful information carried by individual, so the algorithm can easily fall into the local optimal. Comparing the standard CS, the advantage of learning based on individual historical knowledge is that it not only considers the historical optimal solution of individual, but also considers all history solutions of individual, and adjusts their weights self-adaptively through knowledge learning rate and knowledge potential. The knowledge acquired through the historical learning is used to guide the evolution of subsequent individuals in order to better explore new areas. Comprehensive Eqs. (1)–(7), individual historical knowledge learning strategies were added to the Lévy flights random components to ensure that individuals not only refer to the optimal value in the Lévy flights process, but also refer to individual historical experience, which guides direction of Lévy flights and get a better new solution. Cuckoo algorithm based on individual history knowledge learning using Eq. (8) generates a new solution.

$$x_{g+1,i} = x_{g,i} + a_0 \frac{\phi \times \mu}{|\nu|^{1/\beta}} (x_{g,i} - x_{g,best}) \cdot rand_1 + (x_{g,i} - p_i^t) \cdot rand_2 \tag{8}$$

where α_0 is a scaling factor, $\alpha_0 = 0.01$, and $x_{g,best}$ represents the best solution obtained so far. $x_{g,i}$ is individual i for generation t , p_i^t is new position determined of individual i by historical knowledge for generation i , $rand_1$ and $rand_2$ generate random number between 0 and 1. The structure of the IKL-CS algorithm are described in Algorithm 1.

Algorithm 1: IKL-CS algorithm

Input: Population size, NP; Maximum number of function evaluations, MAX_FES,LP

- (1) Randomly initial position of parasitic nest, FES=NP;
 - (2) Calculate fitness value for each solution in each nest;
 - (3) **while** FES<MAX_FES **do**
 - (4) **for** i=1 to NP
 - (5) generate knowledge potential energy $score'_i$ by using Eq. (6);
 - (6) generate knowledge learning rate ξ'_i by using Eq. (5);
 - (7) study based on individual historical knowledge and make a new position p'_i by using equation Eq. (7);
 - (8) generate x_i^{t+1} as new solution by using Eq. (8);
 - (9) **if** $f(x'_i) > f(x_i^{t+1})$
 - (10) Replace x_i^t with x_i^{t+1}
 - (11) End if
 - (12) **end if**
 - (13) choose candidate solution x_i^t ;
 - (14) **if** $f(x'_i) > f(x_i^{t+1})$
 - (15) Replace x_i^t with new solution x_i^{t+1} ;
 - (16) **end if**
 - (17) **end for**
 - (18) throw out a fraction(p_a) of worst nests;
 - (19) **for** each abandoned nest $k \in c$ **do**
 - (20) **for** each $i \in n$ **do**
 - (21) generate solution k_i^{t+1} using Eq. (3);
 - (22) **if** $f(x'_i) > f(x_i^{t+1})$
 - (23) replace x_i^t with new solution x_i^{t+1} ;
 - (24) **end if**
 - (25) **end for**
 - (26) **end for**
 - (27) rank the solution and find the current best;
 - (28) **end while**
 - (29) output the optimal solution X^*
-

4 Experimental and Results

To verify the performance of IKL-CS algorithm, we test our algorithms on eighteen different global optimization problems. Among the eighteen functions, F1–F5 were unimodal functions, F6–F11 were multimodal functions with many local minima,

F12-F14 were shifted unimodal functions, and F15–F18 were shifted multimodal functions. A brief description of these benchmark problems is listed in Table 1. The experiments were carried out on a P4 Dual-core platform with a 1.75 GHz processor and 4 GB memory, running under the Windows 7.0 operating system. The algorithms were developed using MATLAB R2017a. The maximum number of iterations, population size, and the number of runs was set to 30,000, 15, and 30, respectively. The probability that foreign eggs were found was $P_a = 0.25$, with the probability $P = 0.3$.

Table 1. Brief description of eighteen functions

Type	F	Name	Search range	Optimum
Unimodal	F1	Sphere	[-100, 100]	0
	F2	Rosenbrock	[-30, 30]	0
	F3	Step	[-100, 100]	0
	F4	Elliptic	[-100, 100]	0
	F5	Schwefel2.22	[-10, 10]	0
Multimodal	F6	Ackley	[-32, 32]	0
	F7	Rastrigin	[-5.12, 5.12]	0
	F8	Griewank	[-600, 600]	0
	F9	Schwefel2.26	[-500, 500]	0
	F10	Generalized Penalized1	[-50, 50]	0
	F11	Generalized Penalized2	[-50, 50]	0
Shifted unimodal	F12	Shifted Sphere	[-100, 100]	-450
	F13	Shifted Schwefels problem 1.2	[-100, 100]	-450
	F14	Shifted rotated high conditioned elliptic function	[-100, 100]	-450
Shifted multimodal	F15	Shifted Rosenbrock	[-100, 100]	390
	F16	Shifted rotated Ackleys	[-32, 32]	-140
	F17	Shifted rotated Griewanks	[-600, 600]	0
	F18	Shifted rotated Rastrigin	[-5.12, 5.12]	-330

In this experiment, we compare the performance of IKL-CS with CS by conducting some experiments on eighteen well-known benchmark functions. IKL-CS is a version of CS improved by the learning model based on individual history knowledge. The statistical results of each function was chosen as its fitness function. To examine the algorithm scalability, 30-dimensional and 50-dimensional benchmark functions are employed. Averages and standard deviations of optimal solutions over 30 runs achieved by CS, IKL-CS at $D = 30$ and 50 are compared in Table 2. The best results for ten algorithms are shown in boldface.

Table 2. Comparison of IKL-CS with CS for $D = 30$ and 50

F	D = 30		D = 50	
	CS	IKL-CS	CS	IKL-CS
F1	1.19E-26 ± 0.03E-26	0.00E+00 ± 0.00E+00	1.89E-20 ± 4.05E-19	0.00E+00 ± 0.00E+00
F2	5.81E+01 ± 3.22E+01	0.10E-05 ± 0.12E-05	4.33E+01 ± 8.01E+01	5.90E-02 ± 3.01E-01
F3	3.01E+00 ± 1.21E-01	2.55E-199 ± 1.12E+189	4.52E+00 ± 1.21E+00	6.92E-111 ± 2.12E-112
F4	2.38E-62 ± 1.07E-62	7.11E-233 ± 6.09E-220	8.43E-22 ± 4.08E-19	3.78E-78 ± 3.83E-80
F5	2.99E-03 ± 5.20E-03	9.17E-110 ± 5.15E-98	3.86E-01 ± 5.29E-01	1.02E-99 ± 3.87E-98
F6	5.43E-01 ± 3.03E-01	2.55E-15 ± 2.91E-15	5.43E-01 ± 3.03E-01	2.55E-15 ± 2.91E-15
F7	8.91E+01 ± 1.22E+01	1.52E+01 ± 1.22E+01	8.91E+00 ± 1.22E+00	5.87E+01 ± 1.87E+01
F8	5.87E-02 ± 1.75E-03	0.00E+00 ± 0.00E+00	3.98E-01 ± 8.97E-01	0.00E+00 ± 0.00E+00
F9	3.47E+03 ± 3.39E+02	3.25E+03 ± 1.14E+02	9.02E+03 ± 4.77E+03	1.75E+03 ± 8.14E+03
F10	2.87E+00 ± 1.15E+00	1.12E-07 ± 3.13E-07	8.91E+00 ± 3.27E+00	3.81E-07 ± 3.13E-07
F11	5.51E-03 ± 4.46E-02	3.43E-08 ± 4.87E-08	2.48E+01 ± 2.98E+01	3.77E-07 ± 5.77E-07
F12	5.68E-02 ± 0.00E-02	5.88E-15 ± 1.71E-14	1.98E-02 ± 3.88E-02	7.89E-10 ± 2.88E-11
F13	1.38E-02 ± 1.03E-01	3.78E-12 ± 7.11E-12	3.78E+00 ± 1.03E+00	5.28E-01 ± 3.99E-01
F14	4.38E+06 ± 2.99E+06	6.65E+05 ± 5.87E+05	5.84E+07 ± 1.90E+07	2.99E+06 ± 9.01E+06
F15	4.54E+01 ± 1.41E+00	5.12E+01 ± 1.01E+01	9.98E+01 ± 2.32E+00	7.22E+01 ± 9.12E+01
F16	3.22E+00 ± 2.42E+00	5.89E-14 ± 1.33E-14	6.03E+00 ± 5.88E+00	2.11E-09 ± 4.53E-09
F17	3.26E-01 ± 3.14E-01	0.00E+00 ± 0.00E+00	8.78E-01 ± 8.11E-01	0.00E+00 ± 0.00E+00
F18	6.27E+01 ± 3.28E+00	0.00E+00 ± 0.00E+00	2.78E+02 ± 8.11E+00	3.78E+01 ± 2.83E+02

- (1) **The results for the 30-D problems:** As can be seen from Table 2, IKL-CS achieved better results than the CS algorithms for the majority of test functions. The statistical solutions of the IKL-CS, in terms of the average and standard deviations of optimal solutions, are better than other algorithms for F2–F7, F12–F16. For functions F10 and F11, IKL-CS did not reach theoretical optima 0, while IKL-CS converged to better solutions, and IKL-CS converged to theoretical optima for four functions (F1, F8, F17, and F18). These findings demonstrated that the improvement of IKL-CS can balance exploration and exploitation efficiently through a novel combination of both methods.
- (2) **The results for the 50-D problems:** The experiments are conducted on the 50-D problems and results are presented in Table 2. From the results, it can be clearly observed that all 50-D functions become more difficult than their 30-D counterparts, and the results are not as good as in 30-D cases. The IKL-CS algorithms perform better than CS variants on most of the unimodal and multimodal problems experimented in this experiment. For functions F1, F8, and F17, IKL-CS converged to theoretical optima 0. About the IKL-CS to solve multimodal problems, it is based on the search experience for individual historical knowledge. So the larger search space is achieved. Due to this, the statistical solutions of the IKL-CS are better than CS algorithms for the majority of test function except F9–F11, which is proved that IKL-CS has strong robustness.

5 Conclusions

In this study, we introduced a novel IKL-CS algorithm based on individual knowledge learning strategies to solve multi-dimension function optimization problems. To balance the exploration and exploitation of the algorithm, the learning model of population knowledge improved the convergence rate and expanded the search area, which was helpful for enhancing the global exploration ability of IKL-CS. learning model of individual knowledge is not only considers the historical optimal solution of individual, but also considers all history solutions of individual. To verify the performance of IKL-CS, we employed eighteen benchmark test functions from the literature. The results showed that the proposed IKL-CS algorithm clearly outperformed the standard CS algorithm.

Acknowledgements. This work was supported by the scientific research project of Hubei Provincial Department of Education (No. B2017314), National Natural Science Foundation of China (No. 61672391), Innovation team of the Provincial Education Department (No. T201631), and Hubei provincial teaching research project (No. 2016446).

References

1. Wang, G.G., Tan, Y.: Improving metaheuristic algorithms with information feedback models. *IEEE Trans. Cybern.* (2017)
2. Wang, G.G., Cai, X., Cui, Z., Min, G., Chen, J.: High performance computing for cyber physical social systems by using evolutionary multi-objective optimization algorithm. *IEEE Trans. Emerg. Top. Comput.* (2017)
3. Wang, G.G., Chu, H.C.E., Mirjalili, S.: Three-dimensional path planning forUCAV using an improved bat algorithm. *Aerosp. Sci. Technol.* **49**, 231–238 (2016)
4. Deb, K.: An introduction to genetic algorithms. *Sadhana* **24**(4–5), 293–315 (1999)
5. Lim, W.H., Isa, N.M.: Bidirectional teaching and peer-learning particle swarm optimization. *Inf. Sci.* **280**(4), 111–134 (2014)
6. Rahnamayan, S., Tizhoosh, H.R., Salama, M.M.A.: Opposition-based differential evolution. *IEEE Trans. Evol. Comput.* **12**(1), 64–79 (2008)
7. Jia, G.B., Wang, Y., Cai, Z.X., Jin, Y.C.: An improved (l+k)-constrained differential evolution for constrained optimization. *Inf. Sci.* **222**, 302–322 (2013)
8. Wang, G.G., Guo, L., Gandomi, A.H., Hao, G.S., Wang, H.: Chaotic krill herd algorithm. *Inf. Sci.* **274**, 17–34 (2014)
9. Wang, G.G., Gandomi, A.H., Alavi, A.H.: Stud krill herd algorithm. *Neurocomputing* **128** (5), 363–370 (2014)
10. Wang, H., Yi, J.H.: An improved optimization method based on krill herd and artificial bee colony with information exchange. *Memet. Comput.* **10**(2), 177–198 (2018)
11. Wang, G.G., Gandomi, A.H., Alavi, A.H., Hao, G.S.: Hybrid krill herd algorithm with differential evolution for global numerical optimization. *Neural Comput. Appl.* **25**(2), 297–308 (2014)
12. Wang, G.G., Gandomi, A.H., Alavi, A.H.: An effective krill herd algorithm with migration operator in biogeography-based optimization. *Appl. Math. Model* **38**(9–10), 2454–2462 (2014)

13. Wang, G.G., Guo, L., Wang, H., Duan, H., Liu, L., Li, J.: Incorporating mutation scheme into krill herd algorithm for global numerical optimization. *Neural Comput. Appl.* **24**(3–4), 853–871 (2014)
14. Yi, J.H., Wang, J., Wang, G.G.: Improved probabilistic neural networks with self-adaptive strategies for transformer fault diagnosis problem. *Adv. Mechabucal Eng.* **8**(1), 1–13 (2016)
15. Wang, G.G., Gandomi, A.H., Alavi, A.H.: A chaotic particle-swarm krill herd algorithm for global numerical optimization. *Kybernetes* **42**(6), 962–997 (2013)
16. Zhang, Z., Feng, Z.: Two-stage updating pheromone for invariant ant colony optimization algorithm. *Expert Syst. Appl.* **39**(1), 706–712 (2012)
17. Wang, G.G., Guo, L., Duan, H., Wang, H., Liu, L.: Hybridizing harmony search with biogeography based optimization for global numerical optimization. *J. Comput. Theor. Nanosci.* **10**(10), 2318–2328 (2013)
18. Yildiz, A.R.: A new hybrid artificial bee colony algorithm for robust optimal design and manufacturing. *Appl. Soft Comput.* **13**(5), 2906–2912 (2013)
19. Wang, G.G., Deb, S., Gao, X.Z., Coelho, L.D.S.: A new metaheuristic optimization algorithm motivated by elephant herding behavior. *Int. J. Bio-Inspired Comput.* **8**(6), 394–409 (2016)
20. Wang, G.G., Deb, S., Cui, Z.: Monarch butterfly optimization. *Neural Comput. Appl.* 1–20 (2015)
21. Wang, G.G., Deb, S., Zhao, X.C., Cui, Z.H.: A new monarch butterfly optimization with an improved crossover operator. *Oper. Res.* 1–25 (2017)
22. Feng, Y., Wang, G.G., Deb, S., Lu, M., Zhao, X.: Solving 0-1 knapsack problem by a novel binary monarch butterfly optimization. *Neural Comput. Appl.* **28**(7), 1619–1634 (2017)
23. Wang, G.G., Deb, S., Coelho, L.S.D.: Earthworm optimization algorithm: a bio-inspired metaheuristic algorithm for global optimization problems. *Int. J. Bio-Inspired Comput.* (2015)
24. Wang, Y., Gao, S., Yu, Y., Xu, Z.: The discovery of population interaction with a power law distribution in brain storm optimization. *Memet. Comput.* 1–23 (2017)
25. Wang, G.G.: Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems. *Memet. Comput.* 1–14 (2016)
26. Wang, G.G., Deb, S., Gandomi, A.H., et al.: Chaotic cuckoo search. *Soft Comput.* **20**(9), 3349–3362 (2016)
27. Cui, Z., Sun, B., Wang, G., et al.: A novel oriented cuckoo search algorithm to improve DV-Hop performance for cyber-physical systems. *J. Parallel Distrib. Comput.* **103**, 42–52 (2017)
28. Wang, G.G., Gandomi, A.H., Yang, X.S., Alavi, A.H.: A new hybrid method based on krill herd and cuckoo search for global optimization tasks. *Int. J. Bio-Inspired Comput.* **8**(5), 286–299 (2016)
29. Wang, G.G., Gandomi, A.H., Zhao, X., Chu, H.C.: Hybridizing harmony search algorithm with cuckoo search for global numerical optimization. *Soft Comput.* **20**(1), 273–285 (2016)
30. Yang, X.S., Deb, S.: Cuckoo search via Lévy flights. In: *World Congress on Nature & Biologically Inspired Computing*, vol. 71, no. 1, pp. 210–214 (2009)
31. Nguyen, T.T., Vo, D.N.: Modified cuckoo search algorithm for short-term hydrothermal scheduling. *Int. J. Electr. Pow. Energy Syst.* **65**, 271–281 (2015)
32. Srivastava, P.R., Khandelwal, R., Khandelwal, S., Kumar, S., Ranganatha, S.S.: Automated test data generation using cuckoo search and tabu search (CSTS) algorithm. *Int. J. Inteli. Syst.* **21**(2), 195–224 (2012)
33. Yang, X.S., Deb, S.: Engineering optimization by cuckoo search. *Int. J. Math. Model. Numer. Optim.* **1**(4), 330–343 (2010)
34. Chandrasekaran, K., Simon, S.P.: Multi-objective scheduling problem: hybrid approach using fuzzy assisted cuckoo search algorithm. *Swarm Evol. Comput.* **5**, 1–16 (2012)

35. Valian, E., Tavakoli, S., Mohanna, S., Haghi, A.: Improved cuckoo search for reliability optimization problems. *Comput. Ind. Eng.* **64**(1), 459–568 (2013)
36. Li, X.T., Yin, M.H.: Modified cuckoo search algorithm with self adaptive parameter method. *Inf. Sci.* **298**, 80–97 (2015)
37. Li, X., Wang, J., Yin, M.H.: Enhancing the performance of cuckoo search algorithm using orthogonal learning method. *Neural Comput. Appl.* **24**(6), 1233–1247 (2014)
38. Agrawal, S., Panda, R., Bhuyan, S., Panigrahi, B.K.: Tsallis entropy based optimal multilevel thresholding using cuckoo search algorithm. *Swarm Evol. Comput.* **11**, 16–30 (2013)
39. Ouaarab, A., Ahiod, B., Yang, X.S.: Discrete cuckoo search algorithm for the travelling salesman problem. *Neural Comput. Appl.* **24**(7–8), 1659–1669 (2014)



An Improved DV-Hop Algorithm with Jaccard Coefficient Based on Optimization of Distance Correction

Wangsheng Fang, Geng Yang^(✉), and Zhongdong Hu

Jiangxi University of Science and Technology, Ganzhou, China
justinyanggeng@126.com

Abstract. The DV-hop positioning algorithm is range-free positioning algorithm in wireless sensor networks. In order to reduce the positioning error caused by the unknown node positioning within the one-hop distance of the anchor node, a hop-number correction factor based on the Jaccard coefficient is proposed (JDV-hop improved algorithm). Firstly, the positioning problem is converted into the problem of calculating the number of nodes in the area of intersection within the communication radius between neighboring nodes. Then, the differential error in the DDV-hop algorithm is used to solve the problem of calculating the cumulative error of the average distance per hop, and then the average number of nodes per hop is increased. Finally, in the problem of optimal selection of anchor nodes, a credibility factor that allows cooperative positioning between nodes is introduced, and the node with high accuracy of positioning results is effectively converted into an anchor node, thereby achieving the effect of reducing energy consumption. The MATLAB simulation results show that under the same conditions, the improved algorithm has higher positioning accuracy than DDV-hop algorithm and DV-hop algorithm.

Keywords: Node positioning · Anchor node · Jaccard similarity coefficient
Differential error · Cooperative localization

1 Introduction

The wireless sensor network consists of resource-constrained sensor nodes. These nodes can communicate with each other and collaborate to collect information in the environment. Through wireless communication, a self-organized intelligent network system is formed to enable real-time monitoring, sensing and acquisition of monitored areas. Information of various environments or monitoring objects, and processing these information, sending the acquired information to the task management node or the user who needs the information [1]. Because the sensor node is the basic unit of the wireless sensor network, node positioning is the primary problem faced after the deployment of the wireless sensor network system is completed, and is the basis of other functions of the wireless sensor network, so the location information of the node is crucial.

The existing positioning technologies of wireless sensor networks can generally be divided into two categories: range-based and range-free positioning technologies. The distance-based positioning method first needs to accurately measure the distance

information (distance or angle) between related nodes, and then uses trilateration, triangulation, or maximum likelihood estimation positioning calculation methods to calculate the node position. Common ranging-based positioning algorithms are RSSI [2], AOA [3], TOA [4], TDOA [5]. Positioning algorithms based on ranging have the characteristics of high positioning accuracy, but they have two main shortness. First, distance information is easily affected by multi-path fading, noise, and environmental changes. Secondly, additional distance measuring devices are usually required. It will consume more energy and increase hardware costs. The positioning technology without distance measurement has the characteristics of low hardware cost and low energy consumption, and is particularly used in environments where the scale of network is large and energy consumption is small. The positioning algorithms without distance measurement mainly include a centroid algorithm, a convex approximation algorithm, a DV-hop algorithm, an APIT algorithm, and an Amorphous positioning algorithm.

The positioning process of the DV-hop positioning algorithm utilizes the information broadcasting process and location information of the beacon node in the network to perform node positioning, which can effectively save costs and save energy. Therefore, it is a widely used wireless sensor network positioning technology. However, the DV-Hop algorithm has the disadvantage of low positioning accuracy. For example, the positioning accuracy of the network is approximately 33% when the average connectivity is 10 and the isotropic ratio of the beacon node is 10% [6, 7]. Therefore, scholars at home and abroad have continuously improved the DV-hop positioning algorithm. Liu et al. proposed an RWDV-Hop algorithm that uses multiple communication radii to broadcast its own grouping information, and obtains more accurate hop counts of unknown nodes and anchor nodes, thereby more accurately calculating the weighted average hop distance [8]. However, the implementation of this algorithm requires anchor nodes to broadcast data information of different communication radii to neighboring nodes during the flooding phase, thus increasing the energy consumption of network nodes. Dang et al. proposed to use the ratio of the RSSI value received by the node to the RSSI value of the ideal one-hop to correct the one-hop distance of the classical DV-Hop positioning algorithm, and to use the RSSI ranging technology for the first hop node to directly calculate the distance [9], but ignored in the positioning process directly used RSSI value is vulnerable to environmental impact caused by a large error, but also because the anchor node needs to directly read the RSSI value from the node register, which significantly increased the pan Flood data volume and communication overhead. Hou and Zhou et al. proposed a new differential error-based algorithm, DDV-Hop, to improve the average hop size of each positioning node during the positioning calculation, and then to the average hop distance received from different anchor nodes. A weighted evaluation is performed to estimate the location of the node to be located [10]. The algorithm effectively reduces the error of classical DV-Hop, but there is a large error in the estimated hop distance when the number of hops between two 986 selected anchor nodes is large. Fan Shipping proposed on the basis of Hou et al. the use of a weighted average of the distance error, corrected the original average distance per hop, and then used segmented exponential and logarithmic decreasing weights to improve the weight of the particle swarm, using improved The particle swarm algorithm solves the unknown node coordinates [11], thereby improving the positioning accuracy. Zhang introduced multiple communication

radius methods to refine the number of hops between nodes, and when calculating the average hop distance of unknown nodes, removes isolated nodes and uses the average hop distance obtained by the anchor nodes to perform weighted normalization processing so that unknown nodes are located. Increased accuracy [12]. However, it is more costly to normalize all weighted parameter values. These improved positioning algorithms can effectively reduce the error of the classic DV-Hop algorithm under certain conditions, but the effect is not significant in saving node energy consumption and reducing hardware costs, and increases the computational complexity of the entire network.

In this paper, aiming at the problem that the existing DV-hop improved algorithm calculates the average distance per hop error and the error of the one hop distance of the node within the communication radius is still large, an improved algorithm based on the Jaccard coefficient jump correction is proposed (JDV-Hop improved algorithm), introducing the correction factor of the Jaccard coefficient to correct the single-hop distance of the communication radius between the neighbor nodes, and adopting the difference error coefficient of the actual position and estimated position of the anchor node in the DDV-Hop algorithm to Correct the average hop distance, and finally a kind of credible degree factor of cooperative positioning is introduced. Nodes with higher positioning accuracy are upgraded to anchor nodes, and more accurate positioning is achieved on the premise of effective energy saving.

2 DV-Hop Algorithm and Analysis

2.1 Traditional DV-Hop Algorithm

The principle of DV-Hop (distance vector-hop) positioning algorithm is similar to the classical distance vector routing algorithm. In the DV-Hop algorithm, the average hop distance is estimated by calculating the minimum hop count between the unknown node and the anchor node, and the hop distance is used instead of the actual distance to calculate the position coordinate of the unknown node.

The DV-Hop algorithm is divided into the following three phases.

1. Calculate the minimum number of hops for the unknown node and each beacon node. The beacon node broadcasts beacon information to the network. The beacon information includes location information of the beacon node and a hop count parameter with an initial value of 0, for example, a format of $\{id, x_i, y_i, h_i\}$. The receiving node records the minimum number of hops to each beacon node, ignoring the larger number of hops from the same beacon node. Then the hop value is incremented by one and forwarded to the neighbor node. With this method, all nodes in the network can record the minimum number of hops to each beacon node.
2. Calculate the actual hop distance between the unknown node and the beacon node. Each beacon node calculates the average distance per hop by using Eq. (1) according to the location information of other beacon nodes and the number of hops recorded in the first phase.

$$HopSize_{ij} = \frac{\sum_{j \neq i} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{j \neq i} h_j} \tag{1}$$

Among them, $(x_i, y_i), (x_j, y_j)$ are the coordinates of the beacon node i, j and h_i is the hop count between the beacon nodes i and j ($i \neq j$). Then, the beacon node broadcasts the calculated average distance per hop to the network using the packet with the lifetime field. The unknown node only records the first received average distance per hop and forwards it to the neighbor node. This strategy ensures that the vast majority of nodes receive the average distance per hop from the nearest beaconing node. After receiving the average distance per hop, the unknown section calculates the hop distance to each beacon node according to the recorded hop count. Let h_i denote the minimum number of hops from an unknown node to the i anchor node. Then, the hop distance d_i is:

$$d_i = h_i \times HopSize_{ij}(i \neq j) \tag{2}$$

3. When the unknown node obtains the distance from 3 or more anchor nodes, the triangulation method, the maximum likelihood estimation method, or the least squares method is used to calculate the self coordinate position.

3 The Jaccard Coefficient and the Jaccard Distance

The Jaccard coefficient, also known as Jaccard’s similarity coefficient, is used to compare the similarities and differences between finite sample sets. The larger the Jaccard coefficient, the higher the sample similarity [13]. The definition is as follows: Given two sets A, B, the Jaccard coefficient is defined as the ratio of the size of the intersection of A and B to the size of the union of A and B.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \tag{3}$$

When the set A and B are all empty, $J(A, B)$ is defined as 1. The dexterity indicator associated with the Jaccard’s coefficient is called the Jaccard distance and is also used to describe the degree of difference between collections. The larger the Jaccard distance, the lower the sample similarity. The formula is defined as follows:

$$d_j(A, B) = 1 - J(A, B) = \frac{A \Delta B}{|A \cup B|} \tag{4}$$

One of the differences $A \Delta B = |A \cup B| - |A \cap B|$, in the field of data mining, it is often necessary to compare the distance between objects with Boolean attributes. The Jaccard distance is a commonly used method. For example, two comparison objects A,

B are given. A, B have n binary attributes, that is, each attribute takes a value of (0, 1). The following four statistics are defined:

- M_{00} : A, B The attribute value is the number of attributes that are 0 at the same time;
- M_{01} : The number of attributes whose A attribute value is 0 and B attribute value is 1;
- M_{10} : The number of attributes whose A attribute value is 1 and B attribute value is 0;
- M_{11} : A, the number of attributes whose B attribute value is 1 at the same time.

Obviously:

$$M_{00} + M_{01} + M_{10} + M_{11} = n \tag{5}$$

Jaccard coefficient:

$$J(A, B) = \frac{M_{11}}{M_{01} + M_{10} + M_{11}} \tag{6}$$

Jaccard distance:

$$d_j(A, B) = 1 - J(A, B) = \frac{M_{01} + M_{10}}{M_{01} + M_{10} + M_{11}} \tag{7}$$

4 Steps of Improved the Algorithm

1. First assume that all nodes have the same communication radius and R . Each beacon node broadcasts only one packet within the range of communication radius R . The basic information includes $\{ID, (x_i, y_i), Hops, M_i\}$, and the initial value of the hop count 0, M_i is the number of nodes to be located within the communication radius of the beacon node. Within the single-hop distance of the anchor node, it is determined that the node also broadcasts a data packet $\{N_i, N_m\}$, N_i represents the number of all nodes within the communication radius range of each node, and N_m represents the area where it intersects with the communication radius range of the anchor node. The number of nodes within the node, from which the Jaccard coefficient for each node within a single-hop distance is $\tau_i = \frac{N_m}{N_i + M_i}$, And set the hop count of each node whose node hop count is one hop to its Jaccard distance $Hops = 1 - \tau_i$, Then forward the packet to it's next hop node.
2. After all the nodes in the entire network receive the anchor node information, the Euclidean distance d_{ij} between each anchor node can be obtained, and the average hop distance and d'_{ij} of each anchor node are calculated. The d'_{ij} is the anchor node. The estimated distance obtained by multiplying the hop count and the average hop distance between i and j , thus obtaining the differential error value $diff_err_i$ for each anchor node, and broadcasting the differential error of all anchor nodes over the entire network. When the hop count information is received, not only the hop distance and differential error of the first anchor node received are saved. Instead, a threshold value M_{thh} is set so that the number of differential information of the

anchor node received by the unknown node reaches M_{thh} , and finally a coefficient weight σ_i of the correction error can be obtained; and the average hop distance of the unknown node is corrected as $Hopsiz_e_A$.

- Through the two flooding information of the anchor node, each unknown node gets the number and hop distance of the anchor nodes connected with itself. The unknown nodes select three anchor nodes that are close to one another and meet the collinearity, and calculate their credibility factor e_i according to Eqs. (5) and (6). When the unknown node's confidence factor conforms to $e_i \leq \mu_i$, it is upgraded to an anchor node and participates in the positioning of other nodes to be determined. The above steps are repeated with a new set of anchor nodes until there is no unknown node in the network that meets the positioning conditions.

5 Simulation Results

In order to verify the feasibility of the improved methods of traditional DV-Hop algorithm and DDV-Hop algorithm, MATLAB R2014a was used to simulate the DV-hop algorithm, DDV-hop algorithm and the proposed JDV-Hop algorithm. The experimental results were compared and analyzed. The simulation environment was set to a square area of $100\text{ m} \times 100\text{ m}$ and 150 randomly distributed nodes (including anchor nodes and unknown nodes), the node communication radius is R .

Unknown node positioning error formula is:

Average positioning error:

$$E = \frac{\sum_{i=1}^m \sqrt{(x_{real} - x_i)^2 + (y_{real} - y_i)^2}}{m} \quad (7)$$

Relative positioning error:

$$e = \frac{E}{R} \times 100\% \quad (8)$$

5.1 Average Positioning Error Under Different Communication Radius

In the default simulation environment, change the node communication radius R , classic DV-Hop algorithm, DDV-Hop algorithm, improved algorithm in [8], and improved algorithm in this paper (JDV-Hop communicates with nodes in simulation experiments). The radius from 25 m to 55 m, four different average relative error changes as shown in Fig. 1. It can be seen that with the increase of the communication radius R , the average positioning error of the four positioning algorithms are showing a downward trend, DDV-Hop algorithm and The changes of the JDV-Hop algorithm proposed in this paper are relatively stable. When $R > 40$, the DDV-Hop algorithm and the literature [8] algorithm and the JDV-Hop algorithm, the average relative error of the three algorithms tends to be stable, in which DDV-hop The average positioning error

of hop is only 18.5 times lower than that of the traditional DV-hop algorithm. The mean positioning error of the improved JDV-Hop algorithm is about 27% lower than that of the traditional DV-Hop algorithm, which is better than the algorithm of literature [8]. The error value is reduced by 10%, which is 8% lower than the average error value of DDV-Hop. Therefore, under the same simulation environment, when the number of anchor nodes is constant, the corresponding network increases with the increase of the communication radius. When the connectivity becomes larger, the improved algorithm of this paper can be implemented Plus precise positioning.

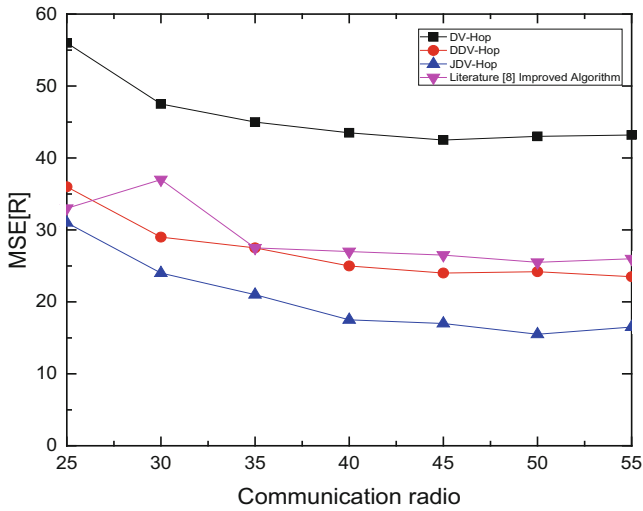


Fig. 1. Normalized MSE versus radio communication for a 30-RN network

5.2 Positioning Errors at Different Anchor Node Densities

In the default simulation environment, the number of anchor nodes is increased from 10 to 50 (i.e., the anchor node ratio is increased from 7% to 33%). Figure 2 represent the simulation diagrams of the relative positioning error values of several algorithms and the anchor node ratio when the communication radius is 30 m, 40 m, and 50 m. In Fig. 2, the improved algorithm of this paper is compared with the classic DV-Hop algorithm and the literature [9]. When the number of anchor nodes is greater than 30, the improved algorithm in this paper is more accurate than the traditional DV-Hop algorithm. It has decreased by nearly 12%, while the DDV-Hop improved algorithm has only decreased by nearly 5%. The method used by the [9] algorithm to measure the distance directly using a ranging technique such as one-hop distance improvement is an algorithm with the lowest average error.

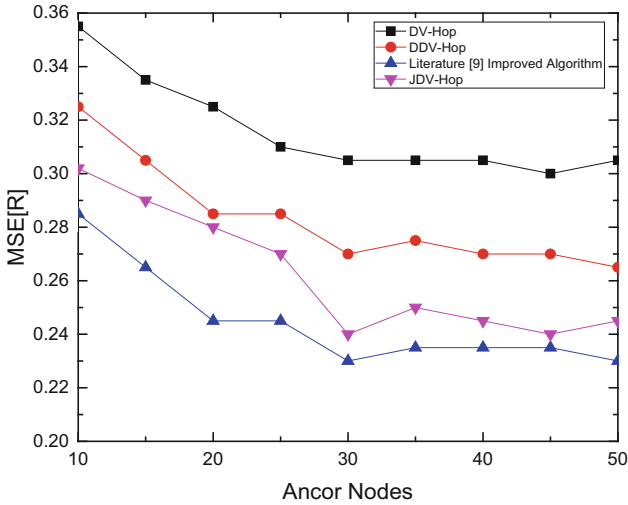


Fig. 2. Normalized MSE versus RNs for a 150-node network ($R = 30$ m)

6 Conclusion

A new DV-hop algorithm named JDV-hop based on the Jacoder coefficient hops correction factor is proposed for the error caused by the unknown node location in the single-hop distance of the node. The positioning problem is transformed into the intersection within the communication radius of the neighbor nodes. The problem of calculating the number of regional nodes. On the basis of not increasing the cost and energy consumption, a correction factor τ_i is used to correct the single-hop distance of the node. The differential correction coefficient of the differential error in the DDV-hop algorithm is introduced to achieve a more accurate average jump calculation. In the third stage of the improved algorithm, this paper proposes a trustworthy factor that can be collocated, upgrades nodes with high accuracy of positioning results to anchor nodes, and reduces energy consumption and node positioning errors. MATLAB simulation results show that the new algorithm JDV-Hop can not only reduce energy consumption compared with traditional DV-Hop positioning algorithms and related literature algorithms, but also can effectively reduce the average positioning error and relative positioning error. In the future, further improvements will be made to the algorithmic problems in the study of a small number of nodes and a low anchor node coverage.

References

1. Gui, L., Val, T., Wei, A., et al.: Improvement of range-free localization technology by a novel DV-hop protocol in wireless sensor networks. *Ad Hoc Netw.* **24**, 55–73 (2015)
2. Kumar, P., Reddy, L., Varma, S.: Distance measurement and error estimation scheme for RSSI based localization in wireless sensor networks. In: *Wireless Communication and Sensor Networks*, pp. 1–4. IEEE (2009)

3. Rong, P., Sichitiu, M.L.: Angle of arrival localization for wireless sensor networks. In: *Sensor and Networks*, vol. 1. pp. 374–382. IEEE (2006)
4. Lewandowski, A., Wietfeld, C.: A comprehensive approach for optimizing TOA-localization in harsh industrial environments. In: *Position Location and Navigation Symposium*, pp. 516–525. IEEE (2010)
5. Kovavisaruch, L., Ho, K. C.: Alternate source and receiver location estimation using TDOA with receiver position uncertainties. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. iv/1065–ib/1068. IEEE (2005)
6. Datta, S., et al.: Distributed localization in static and mobile sensor networks. In: *IEEE International Conference on Wireless and Mobile Computing, In Wireless and Mobile Computing, Networking and Communications*, pp. 69–76. IEEE Computer Society (2006)
7. Niculescu, D., Nath, B.: DV based positioning in ad hoc networks. *Telecommun. Syst.* **22**(1–4), 267–280 (2003)
8. Liu, S., et al.: An improving DV-hop algorithm based on multi communication radius. *Chin. J. Sens. Actuators* **28**(6), 883–887 (2015)
9. Dang, H., Li, Z.: An improved DV-hop location algorithm for wireless sensor network. *Instrum. Tech. Sens.* **1**, 159–163 (2017)
10. Hou, S., Zhou, X., Liu, X.: A novel DV-hop localization algorithm for asymmetry distributed wireless sensor networks. In: *IEEE International Conference on Computer Science and Information Technology*, pp. 243–248. IEEE (2010)
11. Fan, S., Luo, D., Liu, Y.: DV-hop localization algorithm based on hop-size and improvement particle swarm optimization. *J. Transduct. Technol.* **29**(9), 1410–1415 (2016)
12. Jing, L., Zhang, L.: An improved DV-hop algorithm based on optimization of one-hop distance for sensor network localization. *J. Transduct. Technol.* **30**(4), 582–586 (2017)
13. Cao, Y., Yan, K., Xu, J.: A kind of double particle swarm optimization DV-hop localization algorithm based on best anchor nodes set. *J. Transduct. Technol.* **3**, 424–429 (2015)



An Image Encryption Algorithm Based on Hyper-chaotic System and Genetic Algorithm

Xuncaï Zhang^{1(✉)}, Hangyu Zhou¹, Zheng Zhou¹, Lingfei Wang¹,
and Chao Li²

¹ School of Electrics and Information Engineering,
Zhengzhou University of Light Industry, Zhengzhou 450002, China
zhangxuncaï@163.com

² Shan Dong Snton Optical Material Technology Company Limited,
Dongying 157500, China

Abstract. In this paper, a new hybrid encryption scheme based on 4D hyper-chaotic Chen system and genetic algorithm is proposed. In order to improve the security performance of the secret key, the parameters related to the average value of pixels in plaintext images are chosen as the initial values of 4D hyper-chaotic Chen system. After the S-box operation of each row and column, the crossover operation of genetic algorithm is performed on all the rows and columns. Finally, a bitwise XOR operation of the image matrix in one dimension is achieved, and an effective cipher-text image is obtained, which is very secure when defending against differential attacks, statistical analysis attacks and brute force attacks.

Keywords: S-box · Hyper-chaotic system · Genetic algorithm
Image encryption

1 Introduction

In recent years, with the rapid development of information industry in today's society, the security of information data has received more and more attention. In many industrial, commercial and scientific research, data encryption algorithms have made rapid progress. More and more texts, images or videos containing private or confidential information are generated, transmitted and stored on devices such as networks and mobile phones. Therefore, how to ensure the safety and integrity of these confidential information has become a very important topic. A lot of encryption algorithms have been derived in the field of encryption. Among them, DES, AES, RES algorithm is applied to text structure data encryption [1, 2]. Because of the strong association between the adjacent pixels of the plaintext image, these encryption methods have the defects that fail to consider the features of the image itself, so it is not suitable to be applied to the image encryption [3]. In order to overcome these shortcomings, the encryption algorithm based on chaotic system has been put forward and has attracted more and more attention from scholars. A large number of experiments show that the

encryption scheme based on chaotic system is more suitable for image encryption. The chaotic system has ergodicity, and it has high sensitivity, pseudorandom and mixed effects on the initial values and parameters of the system. Therefore, many image encryption schemes based on chaotic systems have been put forward by scholars.

The digital image encryption system based on chaotic system belongs to symmetric secret key cryptosystem. Encryption and decryption have the same secret key. After entering the encrypted image and secret key, the encrypted algorithm outputs the cipher-text image through the encryption algorithm, and the secret key is transmitted to the decryption aspect through the secret channel, and the encrypted image is transmitted to the decryption through the public channel. Decryption system input secret key and cipher-text image, after decryption algorithm, the output of the decrypted image is obtained. According to Shannon information theory, many encryption algorithms based on chaotic systems have been proposed [4–10], they are all composed of two steps of permutation and diffusion [11, 12]. The permutation here refers to changing the pixel position of the image, but the value of the pixel remains unchanged. Diffusion means not changing the position of pixels, but change the gray value of the pixel points, so that the gray information of any pixel point affects the gray value of the other pixels as much as possible. In the permutation stage, there are usually two methods. One is permutation in pixel-level, and the other one is permutation in bit-planes-level. Pixel-level refers to the transformation of the pixel location of the image, but the value of each pixel remains unchanged [13–15]. Bit-plane not only changes the location of pixels, but also changes the values of pixels [16–21]. Therefore, the bit-plane permutation method has more advantages, because it contains two kinds of permutation effects. In the diffusion phase, the value of each pixel must be transformed. At the same time, cipher-text images should also be sensitive to plaintext images. That is to say, if the value of any pixel in a plaintext image changes slightly, then the entire cipher-text image should also be completely changed. Therefore, in most image encryption algorithms, permutation and diffusion phases are applied at the same time.

The chaotic system applied in the encryption algorithm can be divided into one-dimensional (1D) chaotic system and multi-dimensional (MD) chaotic system. The structure of 1D chaotic system is relatively simple and easy to implement. However, its chaotic behavior is small and the secret key space is small. Therefore, the encryption method using 1-D chaotic system is relatively weak. The MD chaotic system is more complex and has multiple parameters. Therefore, the encryption method based on MD chaotic system has larger secret key space and stronger security. Since Leiser et al. proposed the possibility of applying DNA computing to image encryption, and DNA computing has the advantages of large parallel computing, huge storage and ultra-low energy consumption, some image encryption schemes based on DNA computing have been proposed. A 4D hyper-chaotic system is used to generate pseudorandom sequences for diffusion operations, and a circular permutation operation for plaintext images is carried out. These algorithms convert plaintext images and chaotic sequences into DNA matrices, and encrypt images based on the principle of base complementary pairing, encoding and decoding principle, subtraction and XOR principle. Usually, these DNA operations on a common computer are very complicated and very time-consuming. Other encryption methods are based on grayscale images.

2 Preliminary Work

2.1 The Logistic Mapping

Logistic mapping is a simple chaotic mapping from the perspective of mathematical form. As early as 1950s, a number of ecologists used it to describe population changes. This system has extremely complex dynamical behavior and wide range of applications in the field of secure communications. The mathematical expressions are as follows:

$$x_{n+1} = \mu x_n(1 - x_n) \quad (1)$$

where $\mu \in (3.5699, 4]$.

2.2 4D Hyper-chaotic Mapping

Chaos is a complex form of motion peculiar to nonlinear dynamical systems. It is a common phenomenon in nature. Since the discovery of the first classic three dimensional autonomous chaotic system by Lorenz, it has attracted the attention of many chaotic researchers. On the basis of this, many new three-dimensional chaotic systems have been discovered, like Chen system and Lu system. These three dimensional autonomous chaotic systems have only one positive Lyapunov exponent, which are easily realized in physics, but easy to be cracked in engineering applications. The hyper-chaotic system has 2 or more than 2 positive Lyapunov exponents, and its moving orbit is separated in many directions, and has more complex dynamic behavior than the low dimensional chaotic system.

In this paper, 4D hyper-chaotic Chen system is used. It has richer and more complex dynamic behavior than most known systems, so it is more suitable for image encryption. The system can be described by the following 4D ordinary differential equation.

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = -xz + dx + cy - q \\ \dot{z} = xy - bz \\ \dot{u} = x + k \end{cases} \quad (2)$$

Where $a = 36$, $b = 3$, $c = 28$, $d = -16$, $-0.7 \leq k \leq 0.7$, formula (2) is in hyper-chaos. When $k = 0.2$, the 4 Lyapunov exponents of the formula (2) are: $\lambda_1 = 1.552$, $\lambda_2 = 0.023$, $\lambda_3 = 0$, $\lambda_4 = -12.573$. This means that the system is actually a hyper-chaotic system.

2.3 S-box Operation

In order to disrupt the position of row and column pixel values, S-box is used to scramble. Using pseudo random sequence $J = (j_1, j_2, \dots, j_n)$ generated by chaotic mapping, we sort it in ascending order, and a new sequence $K = (k_1, k_2, \dots, k_n)$ is obtained. Next, check the pseudo random sequence J and find out the location values of

each element of the sequence K in sequence J . The sequence L needed for S-box operation can be obtained. The relationship between them is as follows.

$$k_i = j_{l_i} \tag{3}$$

The L sequence is used to permute the row and column pixels.

$$p'_i = p_{l_i} \tag{4}$$

2.4 Genetic Mutation

Genetic mutation refers to the exchange and scrambling of the bits of the elements, thereby changing the value of elements. The obtained pseudo random sequence is performed by modulus operations. The value of each element will be a nonnegative integer less than or equal to 7. Therefore, the possibility of the 8 values corresponds to the 8 mutation rules proposed, which is show in Fig. 1. The 8 bits of each pixel are rearranged and scrambles according to the mutation rules.

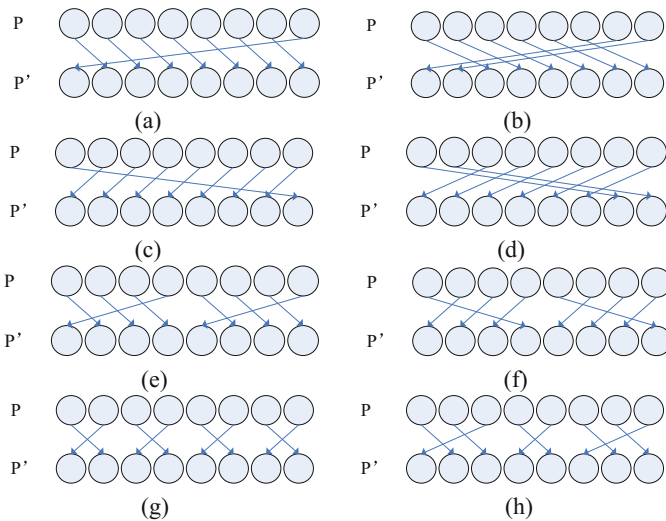


Fig. 1. Mutation rules

2.5 Crossover Operation

Crossover operator plays a central role in genetic algorithm. It is the main way to produce new individuals. The basic idea of crossover operator is to generate new individuals by exchanging part of the genes between two individuals.

In this way, we have introduced a crossover key word to determine the genes that individuals should be inherited by new individuals. For Table 1, two sequences A and

Table 1. Crossover operation

A	101110
B	011000
Crossover key	011101
A'	001100
B'	111010

B are operated. When the bit of “crossover key word” is 0, the new individual A' inherits the gene corresponding to the old individual A. When the “crossover key word” is 1, the new individual A' inherits the gene of the old individual B. In this way, we get a new individual A'. In the same way, a new individual B' is generated.

3 The Proposed Encryption System

The flow chart of our encryption algorithm is shown in Fig. 2, where the detailed encryption steps are given.

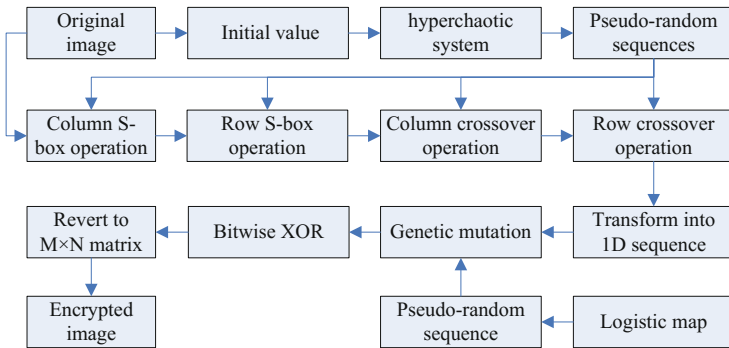


Fig. 2. The flow chart of encryption

Step 1: Given an image P of size M * N, the average value of P(1, 1), P(2, 1)... P(M, 1) is calculated, and it is recorded as a₁, the average value of P(1, 2), P(2, 2) ... P(M, 2) is calculated, and it is recorded as a₂, the average value of P(1, 2), P(2, 2)... P(M, 2) is calculated, and it is recorded as a₃. The same procedure may be easily adapted to obtain the remaining values. Finally n a_i is obtained. Then calculate the value of q_i according to Formula (5).

$$q_i = [(a_i - \lfloor a_i \rfloor) * 10^{10}] \bmod (M * N - 1) + 1 \tag{5}$$

Step 2: 4 integers are produced according to the following formula (6):

$$\begin{cases} q'_1 = \left(\frac{4}{N} \sum_{i=1}^{N/4} q_i \right) \text{mod} 20 \\ q'_2 = \left(\frac{4}{N} \sum_{i=N/4+1}^{N/2} q_i \right) \text{mod} 20 \\ q'_3 = \left(\frac{4}{N} \sum_{i=N/2+1}^{3N/4} q_i \right) \text{mod} 20 \\ q'_4 = \left(\frac{4}{N} \sum_{i=3N/4+1}^N q_i \right) \text{mod} 20 \end{cases} \quad (6)$$

q_i is determined by the average value of the pixel value of each column of the plaintext image and the intermediate amount are related to the plaintext image.

Step 3: Employing q'_1, q'_2, q'_3, q'_4 as the initial value, pseudo-random sequence X, Y, Z, U is generated by hyper-chaotic Chen system. The calculation procedure is detailed in formula (2). In this calculation process, the first $M \times N$ values of the sequences of calculation results are retained.

Step 4: Permutation operation is carried out in the i -th column under the effect of S-box, in which pseudorandom sequence X is employed ($i \in [1, N]$). The matrix permuted is obtained through this procedure. Then the intermediate image P1 is obtained.

Step 5: Permutation operation is carried out in the i -th row under the effect of S-box, in which pseudorandom sequence Y is employed ($i \in [1, M]$). The matrix permuted is obtained through this procedure. Then the intermediate image P2 is obtained.

Step 6: Uniform crossover operation is employed between the i -th column and the $(i + 1)$ -th column, in which the pseudo-random sequence Z is applied ($i \in [1, N - 1]$). Then the intermediate image P3 is obtained.

Step 7: Uniform crossover operation is employed between the i -th row and the $(i + 1)$ -th row, in which the pseudo-random sequence U is applied ($i \in [1, M - 1]$). Then the intermediate image P4 is obtained.

Step 8: Obtain one pseudo-random sequence T through the logistic map. The calculation process is as formula (1).

Step 9: Transform the matrix P4 into one dimensional matrix P5, whose length is $M \times N$. Genetic mutation is performed on the obtained sequences, and the mutation is achieved by scrambling the bit values of each pixel.

Step 10: Bitwise XOR operation is employed between the i -th element and the $(i + 1)$ -th element of E ($i \in (1, MN - 1)$). The sequence obtained from XOR operation is transformed into $M \times N$ matrix, and the final encrypted image P6 is obtained.

4 Experimental Results

In this part, simulation results are given to testify the proposed algorithm. The experimental environment is as follows: CPU: Intel(R) Pentium(R) G3220, 3.00 GHz; Memory: 4.00 GB; Operating system: Windows 7; Coding tool: Matlab 2017. "Lena" (256×256) are used as plain images. Each of the corresponding encrypted and

decrypted images are shown as Fig. 3. It shows that the encrypted image appears like noise so that no one can get any information from the image.

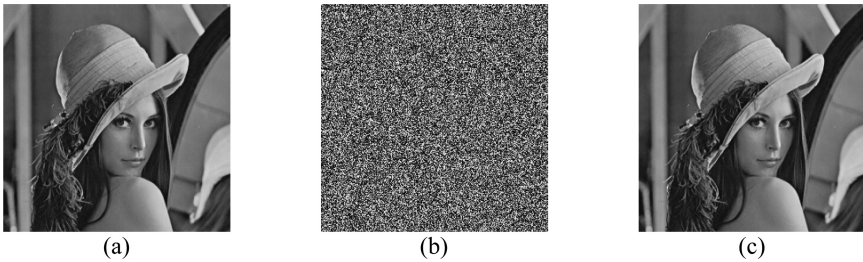


Fig. 3. (a) Plaintext image, (b) encrypted image, (c) decrypted image

In order to prevent statistical analysis attack, the relationship between pixels in the image needs to be thoroughly disrupted, which is the reasons for disturbing the arrangement of the images. The statistical properties of plaintext should be totally destroyed by the encryption algorithm. Whether the algorithm can defense the statistical analysis attack has the following three indicators: the histograms, correlations of two adjacent pixels, and the information entropy. Because the image of the plaintext usually needs to carry the information that it needs to store and transmit, the distribution of its pixels will have a strong regularity, and the correlation between adjacent pixels will be very high. So the information entropy of the plaintext image is usually low.

4.1 Histograms

The histogram can directly reflect the distribution of pixels. Generally speaking, we hope that the distribution is more uniform. Figure 4 shows the histogram of the original image and the encrypted image. As can be seen from the histogram analysis, the distribution of pixels in the encrypted image has been completely disrupted. The correlation between each pixel has been successfully destroyed and the distribution of pixels becomes very uniform.

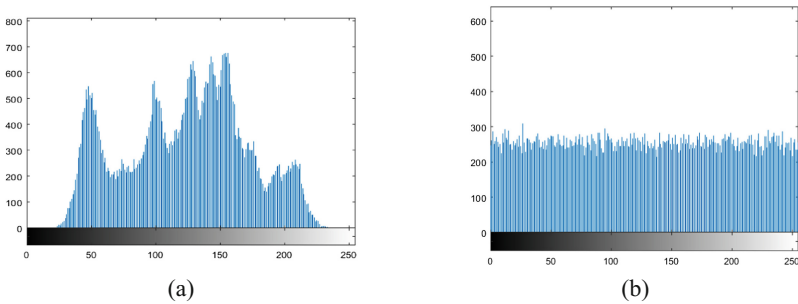
4.2 Correlations of Adjacent Pixels

The correlation between pixels of planar images should be reduced as far as possible, including vertical, horizontal and diagonal correlations of two adjacent pixels. Table 2 shows the correlation of adjacent pixels of the plaintext and cipher-text images in all directions and comparison with references.

We randomly selected 2500 pairs of pixels in all directions, and calculated the correlation between pixels as follows:

Table 2. Correlation coefficients

	Plaintext image	Encrypted image
Horizontal correlation	0.9673	0.0460
Vertical correlation	0.9398	-0.0159
Diagonal correlation	0.9085	-0.0373

**Fig. 4.** (a) Histogram of plaintext image, (b) histogram of encrypted image

$$\begin{cases} E(x) = \frac{1}{N} \sum_{i=1}^N x_i \\ D(x) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))^2 \\ Cov(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))(y_i - E(x)) \\ r_{xy} = \frac{Cov(x, y)}{\sqrt{D(x)} * \sqrt{D(y)}} \end{cases} \quad (7)$$

where x and y represent the gray values of the two adjacent pixels in the image “Lena”.

From Table 2, it can be concluded that the encryption scheme break the correlation between adjacent pixels well. Therefore, the encryption effect is very good and can effectively resist the attacks.

4.3 Information Entropy

Information entropy is an important index to measure the effect of encryption. A good algorithm should make the cipher-text information entropy close to the ideal value. So information entropy analysis is one of the most important indexes of randomness. Information entropy can be calculated by Formula (8).

$$H(X) = - \sum_{x \in Z} P(x_i) \log_2 P(x_i) \quad (8)$$

where $p(x_i)$ denotes the probability of symbol x_i . The standard of the gray image “Lena” we used is 256×256 . For a completely random image matrix, the ideal value of its information entropy should be 8. If the cipher image generated by an encryption algorithm is close to 8, it shows that the encryption effect is effective.

Table 3 gives the information entropy of cipher-text image obtained by the encryption algorithm proposed. It also contains a comparison with the reference [22].

Table 3. Information entropy

	Original image	Encrypted image	Lena in Ref. [22]
Information entropy	7.4532	7.9890	7.9874

It can be seen from the Table 3 that the information entropy of the cipher-text image obtained by the proposed encryption scheme is close to 8 and has better effect than the reference. So the proposed encryption scheme has good defense ability against statistical analysis attacks.

4.4 Key Space Analysis

In all attacks against encrypted information, brute force is the most common and simple way of cracking. Attackers attempt to crack cipher-text images by trying each key one by one. Therefore, as long as the key space of the encryption algorithm is large enough, it can effectively resist the attack of brute force. In the proposed algorithm, the secret key includes the initial value of the 4D hyper-chaotic system and the initial value of the Logistic chaotic map. The secret key space of the initial value of 4D hyper-chaotic Chen system is $2.56 * 10^{57}$. The secret key space of the initial value of the Logistic chaotic map is 8^{14} . Therefore the total secret key space of the proposed encryption algorithm is $1.126 * 10^{70}$. This secret key space is large enough to withstand the attack of brute force, so it is secure under this kind of cracking.

4.5 Differential Attack Analysis

When dealing with differential attacks, the index to evaluate whether the encryption algorithm is secure is UPCR and UACI. The most important points in resisting differential attacks is to compare the differences between two cipher-text images. The definition of UPCR is the number of pixels change rate, and the definition of UACI is the unified average changing intensity. We use the grayscale image of $M * N$. Their calculation process is as follows:

$$\left\{ \begin{array}{l} NPCR = \frac{\sum_{i=1}^M \sum_{j=1}^N C(i,j)}{M*N} * 100\% \\ C(i,j) = \begin{cases} 0 & , P_1(i,j) = P_2(i,j) \\ 1 & , P_1(i,j) \neq P_2(i,j) \end{cases} \\ UACI = \frac{\sum_{i=1}^M \sum_{j=1}^N |P_1(i,j) - P_2(i,j)|}{255*M*N} * 100\% \end{array} \right. \quad (9)$$

Where M and N is the number of rows and columns of the grayscale image used; E_1 and E_2 is the two encrypted image that is only one-bit different from the corresponding pixel location. The ideal value for UPCR is 100% when against differential attack.

Table 4 shows the experimental results and comparison with the Ref. [22]. According to the experimental results, the NPCR value of the proposed encryption algorithm is 99.62%, which is very close to the ideal value of resisting differential attack. The ideal value for UACI is 33% when against differential attack. The result of the encryption algorithm shows that the value of UACI is 28.57%, which is also very close to the ideal value. And their figures are better than those in the Ref. [22]. Therefore, the proposed encryption scheme has a good effect in resisting attacks.

Table 4. Differential attack analysis

	Experimental result	Lena in Ref. [22]
NPCR	99.62%	99.60%
UACI	28.57%	28.13%

5 Conclusions

In this paper, a new hybrid encryption scheme is proposed based on hyper chaotic sequence and genetic algorithm. According to the experiment and simulation results, we can see that the proposed encryption scheme has good defense ability against various attack attacks, and has excellent encryption speed. Therefore, the proposed encryption scheme can be used for image encryption and protect information security.

Acknowledgements. The work for this paper was supported by the National Natural Science Foundation of China (Grant nos. 61602424, 61472371, 61572446, and 61472372), Plan for Scientific Innovation Talent of Henan Province (Grant no. 174100510009), Program for Science and Technology Innovation Talents in Universities of Henan Province (Grant no. 15HAS-TIT019), and Key Scientific Research Projects of Henan High Educational Institution (18A510020).

References

1. Li, S., Chen, G., Zheng, X.: Chaos-based encryption for digital images and videos. In: Furht, B., Kirovski, D. (eds.) *Multimedia Security Handbook*. CRC Press, Boca Raton (2004)
2. Chen, J., Zhu, Z., Fu, C., Zhang, L., Zhang, Y.: An efficient image encryption scheme using lookup table-based confusion and diffusion. *Nonlinear Dyn.* **81**(3), 1151–1166 (2015)
3. Chen, J., Zhu, Z., Fu, C., Yu, H., Zhang, L.: A fast chaos-based image encryption scheme with a dynamic state variables selection mechanism. *Commun. Nonlinear Sci. Numer. Simul.* **20**(3), 846–860 (2015)
4. Shannon, C.E.: Communication theory of secrecy systems. *Bell Syst. Tech. J.* **28**(4), 656–715 (1949)
5. Schneier, B.: *Applied Cryptography: Protocols, Algorithms and Source Code in C*, 2nd edn. Wiley, Hoboken (1996)
6. Chang, C.C., Tai, W.L., Lin, C.C.: A reversible data hiding scheme based on side match vector quantization. *IEEE Trans. Circ. Syst. Video Technol.* **16**(10), 1301–1308 (2006)

7. Dumitrescu, S., Wu, X.: A new framework of lsb steganalysis of digital media. *IEEE Trans. Sig. Process.* **53**(10), 3936–3947 (2005)
8. Seyedzadeh, S., Norouzi, B., Mosavi, M., Mirzakuchaki, S.: A novel color image encryption algorithm based on spatial permutation and quantum chaotic map. *Nonlinear Dyn.* **81**(1–2), 511–529 (2015)
9. Som, S., Dutta, S., Singha, R., Kotal, A., Palit, S.: Confusion and diffusion of color images with multiple chaotic maps and chaos-based pseudorandom binary. *Nonlinear Dy.* **80**, 615–627 (2015)
10. Zhang, Y., Wang, X.: A symmetric image encryption algorithm based on mixed linear-nonlinear coupled map lattice. *Inf. Sci.* **273**, 329–351 (2014)
11. Fouda, J.A.E., Effa, J.Y., Sabat, S., Maaruf, A.: A fast chaotic block cipher for image encryption. *Commun. Nonlinear Sci. Numer. Simul.* **19**(3), 578–588 (2014)
12. Aziz, M., Tayarani-N, M., Afsar, M.: A cycling chaos-based cryptic-free algorithm for image steganography. *Nonlinear Dyn.* **80**, 1271–1290 (2015)
13. Wang, X., Zhang, H.: A color image encryption with heterogeneous bit-permutation and correlated chaos. *Opt. Commun.* **342**, 51–60 (2015)
14. Wang, X., Liu, L., Zhang, Y.: A novel chaotic block image encryption algorithm based on dynamic random growth technique. *Opt. Lasers Eng.* **66**, 10–18 (2015)
15. Zhou, N., Zhang, A., Wu, J., Pei, D., Yang, Y.: Novel hybrid image compression–encryption algorithm based on compressive sensing. *Optik-Int. J. Light Electron Opt.* **125**(18), 5075–5080 (2014)
16. Zhu, H., Zhao, C., Zhang, X.: A novel image encryption-compression scheme using hyper-chaos and Chinese remainder theorem. *Sig. Process. Image Commun.* **28**(6), 670–680 (2013)
17. Ghebleh, M., Kanso, A., Noura, H.: An image encryption scheme based on irregularly decimated chaotic maps. *Sig. Process. Image Commun.* **29**(5), 618–627 (2014)
18. Luo, Y., Du, M., Liu, J.: A symmetrical image encryption scheme in wavelet and time domain. *Commun. Nonlinear Sci. Numer. Simul.* **20**(2), 447–460 (2015)
19. Zhao, J., Wang, S., Chang, Y., Li, X.: A novel image encryption scheme based on an improper fractional-order chaotic system. *Nonlinear Dyn.* **80**, 1721–1729 (2015)
20. Tong, X., Wang, Z., Zhang, M., Liu, Y., Xu, H., Ma, J.: An image encryption algorithm based on the perturbed highdimensional chaotic map. *Nonlinear Dyn.* **80**, 1493–1508 (2015)
21. Leier, A., Richter, C., Banzhaf, W.: Cryptography with DNA binary strands. *Biosystems* **57**(1), 13–22 (2000)
22. Akhavan, A., Samsudin, A., Akhshani, A.: Cryptanalysis of an image encryption algorithm based on DNA encoding. *Opt. Laser Technol.* **95**, 94–99 (2017)



A Performance Comparison of Crossover Variations in Differential Evolution for Training Multi-layer Perceptron Neural Networks

Tae Jong Choi¹, Yun-Gyung Cheong¹, and Chang Wook Ahn²(✉)

¹ Sungkyunkwan University, 2066, Seobu-ro,
Jangan-gu, Suwon-si, Gyeonggi-do, Republic of Korea
{gry17, aimecca}@skku.edu

² Gwangju Institute of Science and Technology (GIST),
123, Cheomdangwagi-ro, Buk-gu, Gwangju, Republic of Korea
cwan@gist.ac.kr

Abstract. Artificial neural networks (ANNs) are a kind of well-known machine learning techniques, and it is required to adjust the weights of their neurons to learn a given task, which usually done by using a gradient-based optimization algorithm. However, gradient-based optimization algorithms likely get stuck in a local optimum, and therefore, researchers have attempted to apply population-based metaheuristics. In this paper, we study the performance comparison of various crossover operators in differential evolution (DE) for training ANNs. We investigated the classification performance of three crossover operators, the binomial crossover, the exponential crossover, and the multiple exponential recombination (MER), with medical datasets. The experimental results show that the binomial crossover and the MER have better performance compared with the exponential crossover, and the exponential crossover varies significantly in performance depending on the architecture. Also, we found that dependent variables in training ANNs may not be located proximately each other, which results in makes the advantage of the exponential crossover and the MER effectless.

Keywords: Artificial neural networks
Differential evolution algorithm · Crossover operator
Feed-forward neural network · Neural network training

1 Introduction

Artificial neural networks (ANNs) are a well-known machine learning technique that has a powerful performance on function approximation, classification, regression, and reinforcement learning. There are numerous varieties of ANNs depending on their structure and how each neuron connects the other, and one

of the most popular types of ANNs is multi-layer perceptron (MLP) neural networks. MLPs have three conceptual layers, input, hidden, and output, and each layer contains a set of neurons that consists of inputs, a bias, weights for both the inputs and the bias, a nonlinear activation function, and an output.

To learn a given task, an ANN is required to adjust the weights of its neurons, and usually, a gradient-based optimization algorithm is applied to that. However, gradient-based optimization algorithms have a disadvantage that they likely get stuck in a local optimum during the learning process [1]. Therefore, researchers have attempted to apply population-based metaheuristics such as Artificial Bee Colony (ABC), Differential Evolution (DE), and Particle Swarm Optimization (PSO) to the learning process of adjusting the weights of ANNs [1].

A recent study claimed that in deep architectures of recently proposed ANNs gradient-based optimization algorithms might not get stuck in a local optimum because there is a significantly small probability of existing local optima in huge dimensional problems [2]. However, we still need to consider training ANNs with population-based metaheuristics because not all the real-world problems require a large-scale structure. In some practical problems, middle or small sized ANNs are better than a large-scale structure to avoid a well-known critical issue, called the overfitting problem. Moreover, if the claim is valid, then the probability that gradient-based optimization algorithms on middle or small sized ANNs converge to a local optimum is significantly high, so swarm and evolutionary computation techniques can be an alternative to overcome this drawback.

In this paper, we study the performance comparison of various crossover operators in differential evolution (DE) for training ANNs. DE [3] is one of the most popular EAs that shows powerful optimization performance on multidimensional real-valued problems, and its effectiveness has been demonstrated on many real-world problems [4–6]. Although some studies successfully applied DE for training ANNs for classification and regression problems [7–11], there is still room for improvement. For example, it is needed to consider which the mutation strategy and the crossover operator is suitable for training ANNs, instead of simply using one of the state-of-the-art DE variants that prove its effectiveness on other optimization problems. In addition, most of state-of-the-art DE variants are based on the binomial crossover, but this operator handles each variable independently, so there is a chance that highly correlated variables might be separated [12]. In addition, the weights of a neuron may highly correlate with each other, so the binomial crossover may not be useful for training ANNs, unlike other optimization problems. In this paper, therefore, we investigate the optimization performance of various crossover operators in DE for training ANNs, and this can be considered as a fundamental study for the future work.

2 Related Work

2.1 DE Algorithm

In this section, we briefly explain the conventional DE, called DE/rand/1/bin. The conventional DE has the NP number of individuals as a population,

and each individual is a D -dimensional vector, denoted by $\mathbf{X}_{i,G} = \{x_{i,G}^1, x_{i,G}^2, \dots, x_{i,G}^D\}$ where G denotes the generation. At first, the lower and upper bounds are initialized as $\mathbf{X}_{min} = \{x_{min}^1, x_{min}^2, \dots, x_{min}^D\}$ and $\mathbf{X}_{max} = \{x_{max}^1, x_{max}^2, \dots, x_{max}^D\}$, respectively. Then all the individuals are initialized as follows.

$$x_{i,0}^j = x_{min}^j + rand_{i,j} \cdot (x_{max}^j - x_{min}^j) \tag{1}$$

where $rand_{i,j}$ denotes a uniformly distributed random number.

The conventional DE has three repeated operators, called mutation, crossover, and selection. In the mutation operator, at first, three donor individuals are randomly selected from the current population, denoted by $\mathbf{X}_{r_1,G}$, $\mathbf{X}_{r_2,G}$, and $\mathbf{X}_{r_3,G}$ and $r_1 \neq r_2 \neq r_3 \neq i$. Then a mutant vector ($\mathbf{V}_{i,G}$) for each target individual is calculated as follows.

$$\mathbf{V}_{i,G} = \mathbf{X}_{r_1,G} + F \cdot (\mathbf{X}_{r_2,G} - \mathbf{X}_{r_3,G}) \tag{2}$$

where F denotes the scaling factor, which is one of the control parameters of DE.

In the crossover operator, at first, a random integer $j_{rand} \in [1, D]$ is generated, used for that at least one component of a trial vector is copied from a mutant vector. Then a trial vector ($\mathbf{U}_{i,G}$) for each target individual is calculated as follows.

$$u_{i,G}^j = \begin{cases} v_{i,G}^j & \text{if } rand_{i,j} \leq CR \text{ or } j = j_{rand} \\ x_{i,G}^j & \text{otherwise} \end{cases} \tag{3}$$

where CR denotes the crossover rate, which is one of the control parameters of DE.

The selection operator compares the trial vectors and their associated target individuals. Then if the fitness value of a trial vector is better than or equal to that of its associated target individual, the trial vector is selected as a member of the next generation. Otherwise, the trial vector is discarded, and the target individual remains for the next generation. A member of the next generation is calculated as follows.

$$\mathbf{X}_{i,G+1} = \begin{cases} \mathbf{U}_{i,G} & \text{if } f(\mathbf{U}_{i,G}) \leq f(\mathbf{X}_{i,G}) \\ \mathbf{X}_{i,G} & \text{otherwise.} \end{cases} \tag{4}$$

The conventional DE repeats these the mutation, the crossover, and the selection operators until one of the termination criteria is met. Since the conventional DE proposed, researchers have devised enhancement techniques such as adaptive parameter control [13–22], automatic strategy control [23,24], and integrating other machine learning techniques [25–27].

2.2 Crossover Operators in DE Algorithm

In this section, we elaborate three crossover operators, binomial, exponential, and multiple exponential. Regarding the binomial crossover, we already explained its

algorithmic procedure on Sect. 2.1, so we only discuss its properties in here. The binomial crossover is generally considered more robust than the exponential crossover due to two reasons. First, it is easy to control the number of mutated components in a trial vector because it has a linear relationship between the control parameter CR and the mutation probability [28] while the exponential crossover has a nonlinear relationship. Second, the binomial crossover can produce all the 2^D possible outcomes while the exponential crossover can only produce part of it [12].

The algorithmic procedure of the exponential crossover is as follows. At first, a random integer $n \in [1, D]$ is generated, used for as a starting point for exchanging components. Then the number of exchanged components L is calculated as follows.

$L = 0$; DO $\{L = L + 1;\}$ WHILE $((rand_{i,j} \leq CR)$ AND $(L \leq D))$;
 After that, a trial vector is calculated as follows.

$$u_{i,g}^j = \begin{cases} v_{i,g}^j & \text{if } j = \langle n \rangle_D, \langle n + 1 \rangle_D, \dots, \langle n + L - 1 \rangle_D \\ x_{i,g}^j & \text{otherwise} \end{cases} \quad (5)$$

where $\langle \cdot \rangle_D$ denotes the modulo function with the modulus D . The exponential crossover is considered as a consecutive crossover, and therefore, it has an advantage over the binomial crossover for finding an optimal solution on nonseparable problems. That is, adjacent components easily tend to be disrupted in the binomial crossover whereas they can be preserved in a significantly high probability in the exponential crossover.

Recently, a new crossover operator is proposed, called multiple exponential recombination (MER) [12]. MER is considered as a semi-consecutive crossover operator that divides a trial vector into a few segments, and each segment is copied from the components of either a mutant vector or its associated target individual. Therefore, as its name suggests, MER is similar to the exponential crossover being repeated several times. The algorithmic procedure of MER is as follows. At first, a random integer $n \in [1, D]$ is generated, used for as a starting point for exchanging components. Then four new control parameters, $E_m = T \cdot CR$, $E_s = T \cdot (1 - CR)$, $CR_m = E_m / (E_m + 1)$, and $CR_s = E_s / (E_s + 1)$, are initialized. In here, E_m and E_s denote the approximated average size of each segment from a mutant vector and its associated target individual, respectively, and T controls the size of exchanged segments, which is fixed at ten [12] in all of the experiments in this paper. Although it seems that additional computation cost is required for adjusting the four new control parameters, these parameters depend on CR and T that is fixed at 10, and therefore, it is similar to the traditional operators. After initializing all the control parameters, MER calculates a trial vector, presented in Algorithm 1.

The MER possesses the advantages of both the binomial and the exponential crossover operators. First, it is a linear relationship between the control parameter CR and the mutation probability, so it is easy to control the number of mutated components in a trial vector. Second, it can produce all the 2^D possible outcomes. Finally, it is empirically proved that MER can handle dependent variables well [12].

Algorithm 1. Algorithmic procedure of MER

```

k = 1, Mutation_Enable = 1
while k ≤ D do
  if Mutation_Enable = 1 then
    while k ≤ D and randi,j ≤ CRm do
      j = (n)D
      ui,Gj = vi,Gj
      n = n + 1 and k = k + 1
    end while
    Mutation_Enable = 0
  else
    while k ≤ D and randi,j ≤ CRs do
      j = (n)D
      ui,Gj = xi,Gj
      n = n + 1 and k = k + 1
    end while
    Mutation_Enable = 1
  end if
end while

```

3 DE Training Algorithm for Artificial Neural Networks

As we mentioned earlier, DE is an evolutionary algorithm that is to find an optimal solution for multidimensional real-valued functions. Therefore, DE can be applied for training ANNs instead of a gradient-based optimization algorithm. In order to apply DE for training ANNs, we need to define an objective function as well as the structure of chromosomes, and there is a pioneer study [7] that suggests how to define these factors. Therefore, In this section, we briefly introduce DE training algorithm [7].

The output vector of an MLP neural network is the function of an input vector \mathbf{x} and a weight vector \mathbf{W} , i.e., $\mathbf{y} = f(\mathbf{x}, \mathbf{W})$. In supervised learning, the input vector \mathbf{x} and the output vector \mathbf{y} are given, and the weight vector \mathbf{W} is needed to be adjusted. In order to measure the quality of a neural network, a network error function E can be defined as follows.

$$E(\mathbf{y}, f(\mathbf{x}, \mathbf{W})) : (\mathbf{y}^{D_o}, \mathbf{x}^{D_i}, \mathbf{W}^{D_w}, f) \rightarrow R \tag{6}$$

The goal of DE training algorithm is to minimize the network error function $E(\mathbf{y}, f(\mathbf{x}, \mathbf{W}))$ by optimizing the D_w -dimensional vector $\mathbf{W} = (w_1, w_2, \dots, w_{D_w})$. Similar to the conventional DE, DE training algorithm has the NP number of individuals as a population where each individual is represented as a D_w -dimensional vector.

The evolutionary operators of DE training algorithm, mutation, crossover, and selection, work in the same manner as the conventional DE.

4 Influence of the Crossover Variants on the Classification Performance of DE Training Algorithm

4.1 Breast Cancer Datasets

In this paper, we investigate the influence of the crossover variants on the classification performance of DE training algorithm on a real-world problem. We

used a composition of medical datasets, called breast cancer, which is one of the most common diseases along with lung and bronchus cancers. This kind of medical applications can be a significant help in the decision-making process of practitioners. Therefore, there are many related datasets are available on the web, which helps machine learning and evolutionary computing researchers to verify the performance of their new algorithms.

The datasets consist of Breast Cancer Wisconsin Diagnostic (WDBC), Breast Cancer Wisconsin Prognostic (WPBC), and Breast-Cancer [29]. The WDBC and the WPBC datasets are divided into a total of 32 input features and two classes. And, the Breast-Cancer dataset is divided into a total of 9 input features and two classes. In the WDBC dataset, there are 569 instances, of which 50% is used as training data, 25% is used as validation data, and the remaining 25% is used as test data in this paper. In the WPBC dataset, there are 198 instances, of which 50% is used as training data, 25% is used as validation data, and the remaining 25% is used as test data in this paper. And finally, in Breast-Cancer dataset, there are 286 instances, of which 50% is used as training data, 25% is used as validation data, and the remaining 25% is used as test data in this paper.

4.2 Settings for the MLP Neural Network

In this paper, we applied six MLP architectures to evaluate their classification performance. This is because, the best MLP architectures are unknown to solve the datasets, and therefore, we applied multiple candidates to reduce the selection bias. The six MLP architectures are as follows.

1. [15:10:10:2]: three hidden layers where the first layer has 15 neurons and the rest hidden layers has ten neurons, and the last layer is the output layer.
2. [15:10:5:2]: three hidden layers where the first layer has 15 neurons, the second has ten neurons, and the third has five neurons.
3. [20:10:2]: two hidden layers where the first layer has 20 neurons, the second has ten neurons.
4. [20:5:2]: two hidden layers where the first layer has 20 neurons, the second has five neurons.
5. [15:10:2]: two hidden layers where the first layer has 15 neurons, the second has ten neurons.
6. [15:5:2]: two hidden layers where the first layer has 15 neurons, the second has five neurons.

Regarding the activation functions on each neuron, we applied the sigmoid function, which is not recommended in many machine learning researchers due to the vanishing gradient problem. However, training ANNs with population-based metaheuristics has no issue with the problem, and even, it can apply different and non-differentiable activation functions on each neuron. In addition, we used Mean Square Error (MSE) function as the objective function, and 64 as a batch size.

4.3 Settings for the DE Training Algorithm

The optimization performance of DE algorithm significantly depends on which mutation strategy, crossover operator, and the values of control parameters are used. In this paper, we combined two well-known mutation strategies with three crossover operators as explained in Sect. 2.2 to evaluate their classification performance. At first, we used DE/rand/1 and DE/current-to-best/2 mutation strategies, and the former strategy has a robust explorative property while the other has a high exploitive property. Therefore, the DE/rand/1 strategy has slow convergence speed but less chance of getting stuck in a local optimum, the other has vice versa characteristics.

The followings are the list of compared algorithms.

1. DE/rand/1/bin with $F = 0.5$ and $CR = 0.9$.
2. DE/rand/1/exp with $F = 0.5$ and $CR = 0.9$.
3. DE/rand/1/mer with $F = 0.5$ and $CR = 0.9$.
4. DE/current-to-best/2/bin with $F = 0.5$ and $CR = 0.3$.
5. DE/current-to-best/2/exp with $F = 0.5$ and $CR = 0.3$.
6. DE/current-to-best/2/mer with $F = 0.5$ and $CR = 0.3$.

In the article [9], the authors presented several important suggestions for using DE training algorithm. At first, in order to prevent the overfitting problem, the authors presented a modified version of selection operator (ES-1), which is as follows.

$$\mathbf{X}_{i,G+1} = \begin{cases} \mathbf{U}_{i,G} & \text{if } f(\mathbf{U}_{i,G}, TR) \leq f(\mathbf{X}_{i,G}, TR) \text{ and } f(\mathbf{U}_{i,G}, V) \leq f(\mathbf{X}_{i,G}, V) \\ \mathbf{X}_{i,G} & \text{otherwise.} \end{cases} \quad (7)$$

where TR and V denote the data from training and validation dataset, respectively. Second, the authors verified an effective way for the initialization and the boundaries of the weights of each neuron by experimentally, where initializes each weight by $[-1, 1]$, and sets the lower and the upper bounds of each weight for during the training session as $[-1000, 1000]$. Finally, the authors used the maximum number of function evaluations and the population size as $10000 \cdot D$ and $5 \cdot D$, respectively. In this paper, we used these suggestions for the performance evaluations.

4.4 Experimental Results

Table 1 shows the result of the Friedman with Dunn's posthoc test. In this table, DE/rand/1/mer ranked best, but all of the null hypotheses are not rejected according to the adjusted p -value with the Dunn's test. The result indicates that DE/rand/1/mer is not able to find statistically better solutions than the other two algorithms. However, it should be noted that the Dunn's posthoc test is a quite conservative test.

The average best mean and its standard deviation is provided in Table 2. In this table, the classification performance of each algorithm is presented according

Table 1. Friedman with Dunn’s posthoc test for DE/rand/1 variants

Algorithm	Average ranking	z-value	p-value	Significant
DE/rand/1/bin	1.83			
DE/rand/1/exp	2.22	-1.17E+00	2.43E-01	No
DE/rand/1/mer	1.94	-3.33E-01	7.39E-01	No

Table 2. Mean and standard deviation for each DE/rand/1 variants on each architecture and dataset averaged over 10 runs

Dataset	Architecture	DE/rand/1/exp MEAN (STD DEV)		DE/rand/1/mer MEAN (STD DEV)		DE/rand/1/bin MEAN (STD DEV)
WDBC	15:10:10:2	2.00E-01 (4.35E-03)	+	1.30E-01 (1.61E-02)	=	1.28E-01 (1.74E-02)
	15:10:5:2	2.09E-01 (8.47E-03)	+	1.28E-01 (1.35E-02)	=	1.29E-01 (1.71E-02)
	20:10:2	1.50E-01 (6.21E-03)	+	1.21E-01 (1.11E-02)	=	1.26E-01 (1.48E-02)
	20:5:2	1.84E-01 (5.22E-03)	+	1.13E-01 (1.50E-02)	=	1.20E-01 (1.52E-02)
	15:10:2	1.69E-01 (1.20E-02)	+	1.24E-01 (1.41E-02)	=	1.23E-01 (1.75E-02)
	15:5:2	1.80E-01 (7.05E-03)	+	1.17E-01 (1.66E-02)	=	1.20E-01 (1.71E-02)
WPBC	15:10:10:2	1.74E-01 (2.16E-03)	=	1.69E-01 (6.70E-03)	=	1.69E-01 (9.69E-03)
	15:10:5:2	1.74E-01 (1.73E-03)	=	1.65E-01 (6.73E-03)	=	1.71E-01 (1.08E-02)
	20:10:2	1.73E-01 (2.75E-03)	=	1.73E-01 (7.57E-03)	=	1.70E-01 (1.11E-02)
	20:5:2	1.76E-01 (5.25E-03)	=	1.67E-01 (1.37E-02)	=	1.71E-01 (1.25E-02)
	15:10:2	1.73E-01 (3.91E-03)	=	1.68E-01 (7.88E-03)	=	1.72E-01 (1.31E-02)
	15:5:2	1.73E-01 (2.88E-03)	=	1.73E-01 (1.19E-02)	=	1.72E-01 (1.29E-02)
Breast-Cancer	15:10:10:2	5.97E-02 (4.66E-04)	=	6.31E-02 (4.27E-03)	=	6.33E-02 (2.30E-03)
	15:10:5:2	5.95E-02 (5.15E-04)	=	6.20E-02 (3.31E-03)	=	6.23E-02 (2.23E-03)
	20:10:2	6.03E-02 (7.78E-04)	=	6.53E-02 (6.24E-03)	=	6.41E-02 (4.59E-03)
	20:5:2	6.07E-02 (6.80E-04)	=	6.73E-02 (3.66E-03)	=	6.40E-02 (3.86E-03)
	15:10:2	6.05E-02 (8.22E-04)	=	6.61E-02 (7.54E-03)	=	6.32E-02 (2.98E-03)
	15:5:2	6.09E-02 (6.11E-04)	-	6.50E-02 (5.07E-03)	=	6.47E-02 (4.08E-03)
Total +/-/-				6/11/1		0/18/0

Statistical significance was evaluated according to the Wilcoxon signed-rank test at a 0.05 level of significance.
 + indicates DE/rand/1/bin is significantly better. = indicates difference is not significant.
 And, - indicates DE/rand/1/bin is significantly worse.

to the datasets and the architectures, along with the results of the Wilcoxon signed-ranked test. At first, comparing DE/rand/1/bin with DE/rand/1/exp, the former has statistically better solutions on six experiments, which are all of the WDBC tests, while the latter has statistically better on one experiment, which are one of the Breast-Cancer tests. In terms of comparing DE/rand/1/bin with DE/rand/1/mer, both algorithms found statistically similar solutions on all the problems. Therefore, these algorithms performed similarly, and these results are supported by the Friedman with Dunn’s posthoc test, which is shown in Table 1.

In addition, we compared the classification performance according to the architectures. Regarding the WDBC dataset, DE/rand/1/bin and DE/rand/1/mer performed well with 20:5:2 structure, while DE/rand/1/exp performed well with 20:10:2 structure. Regarding the WPBC dataset, all the algorithms performed poorly with the structures, which is because the number of instances of the WPBC dataset is considerably less than the other datasets. Finally, regarding the Breast-Cancer dataset, all the algorithms performed well with 15:10:5:2 structure.

The interesting thing to observe here is that DE/rand/1/exp varies significantly in performance depending on the architecture. In other words, it can be observed that the performance of DE/rand/1/exp in the WDBC dataset is

significantly different between 15:10:5:2 structure and 20:10:2 structure, while DE/rand/1/bin and DE/rand/1/mer show similar performance overall. Therefore, if we use DE/rand/1/exp to train an MLP neural network, we need to decide carefully which architecture to use.

We can observe similar results about DE/current-to-best/2 variants in Tables 3 and 4. Table 3 shows the Friedman with Dunn’s posthoc test, and the average best mean and its standard deviation for each algorithm is provided in Table 4. These results are similar to the previous DE/rand/1 variants experiments. Due to space limitation, we omit the detail explanation of Tables 3 and 4.

Table 3. Friedman with Dunn’s posthoc test for DE/current-to-best/2 variants

Algorithm	Average ranking	z-value	p-value	Significant
DE/c-to-b/2/bin	1.89			
DE/c-to-b/2/exp	2.33	-1.33E+00	1.82E-01	No
DE/c-to-b/2/mer	1.78	3.33E-01	7.39E-01	No

Table 4. Mean and standard deviation for each DE/current-to-best/2 variants on each architecture and dataset averaged over 10 runs

Dataset	Architecture	DE/c-to-b/2/exp MEAN (STD DEV)		DE/c-to-b/2/mer MEAN (STD DEV)		DE/c-to-b/2/bin MEAN (STD DEV)
WDBC	15:10:10:2	2.19E-01 (7.09E-03)	+	1.43E-01 (2.11E-02)	=	1.51E-01 (1.59E-02)
	15:10:5:2	2.21E-01 (5.97E-03)	+	1.48E-01 (1.27E-02)	=	1.58E-01 (1.43E-02)
	20:10:2	1.76E-01 (9.70E-03)	+	1.19E-01 (1.21E-02)	=	1.21E-01 (1.42E-02)
	20:5:2	1.97E-01 (9.78E-03)	+	1.35E-01 (1.34E-02)	=	1.26E-01 (9.88E-03)
	15:10:2	1.91E-01 (1.07E-02)	+	1.24E-01 (9.76E-03)	=	1.31E-01 (8.83E-03)
	15:5:2	2.01E-01 (1.20E-02)	+	1.25E-01 (1.68E-02)	=	1.33E-01 (1.04E-02)
WPBC	15:10:10:2	1.76E-01 (1.62E-03)	+	1.72E-01 (2.92E-03)	=	1.67E-01 (6.52E-03)
	15:10:5:2	1.75E-01 (1.49E-03)	+	1.69E-01 (3.98E-03)	=	1.69E-01 (6.82E-03)
	20:10:2	1.76E-01 (1.96E-03)	+	1.67E-01 (7.09E-03)	=	1.68E-01 (5.53E-03)
	20:5:2	1.75E-01 (2.38E-03)	+	1.66E-01 (6.78E-03)	=	1.68E-01 (4.09E-03)
	15:10:2	1.73E-01 (3.86E-03)	=	1.64E-01 (7.58E-03)	=	1.67E-01 (4.66E-03)
	15:5:2	1.75E-01 (1.34E-03)	+	1.68E-01 (1.20E-02)	=	1.63E-01 (6.03E-03)
Breast-Cancer	15:10:10:2	5.94E-02 (1.69E-04)	-	6.19E-02 (1.72E-03)	=	6.13E-02 (1.55E-03)
	15:10:5:2	5.93E-02 (1.48E-04)	-	6.22E-02 (1.39E-03)	=	6.16E-02 (1.61E-03)
	20:10:2	6.01E-02 (1.08E-03)	=	6.28E-02 (1.88E-03)	=	6.21E-02 (1.87E-03)
	20:5:2	5.96E-02 (3.16E-04)	=	6.31E-02 (1.79E-03)	=	6.21E-02 (3.06E-03)
	15:10:2	5.96E-02 (5.62E-04)	=	6.20E-02 (1.58E-03)	=	6.26E-02 (2.22E-03)
	15:5:2	5.99E-02 (1.04E-03)	=	6.26E-02 (1.72E-03)	=	6.38E-02 (4.81E-03)
Total +/-/-				11/5/2		0/18/0

Statistical significance was evaluated according to the Wilcoxon signed-rank test at a 0.05 level of significance.
 + indicates DE/rand/1/bin is significantly better, = indicates difference is not significant.
 And, - indicates DE/rand/1/bin is significantly worse.

As a result, it was confirmed that (1) the binomial crossover and the MER have better performance compared with the exponential crossover, and (2) the binomial crossover and the MER have the more robust performance with different architectures, regardless of the mutation operation. One interesting point is that there is no statistical difference in performance between the binomial crossover and the MER. This result implies that linkages between dependent variables may be not highly related with the proximity between them so the exponential crossover and the MER may not have a benefit over the binomial crossover.

5 Conclusion

In this paper, we empirically analyzed the influence of the crossover variants on the classification performance of DE training algorithm. This kind of studies for investigating the performance of crossover operator is required because the problem space of training MLP neural networks differs from that of optimization problems, and the crossover operator significantly affects the optimization performance of DE algorithm. In addition, the problem of training MLP neural networks has large-scale and inseparable characteristics, and therefore, it can be a contribution to find or develop a suitable crossover operator with respect to the characteristics.

In this paper, we investigated the classification performance of three crossover operators, the binomial crossover, the exponential crossover, and the MER, with medical datasets, the breast cancer dataset. In order to reduce the selection bias, we tested six different MLP architectures and two mutation strategies, DE/rand/1 and DE/current-to-best/2. The former strategy has a robust explorative property, which has less chance of getting stuck in a local optimum, and the latter has a high exploitative property, which has fast convergence speed.

The experimental results show that the binomial crossover and the MER have better performance compared with the exponential crossover regardless of the mutation operation. In addition, we found that the exponential crossover varies significantly in performance depending on the architecture. Therefore, if we use DE/rand/1/exp or DE/current-to-best/2/exp to train an MLP neural network, we need to decide carefully which architecture to use. In addition, one interesting point is that there is no statistical difference in performance between the binomial crossover and the MER. This result implies that linkages between dependent variables may be not highly related with the proximity between them so the exponential crossover and the MER may not have a benefit over the binomial crossover. The studying for more rigorous analysis on the linkage information and suitable crossover operator for training DE algorithm that identifies does not proximately dependant variables are left for future work.

Acknowledgement. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2017R1C1B2012752), and Basic Science Research Program through the NRF funded by the Ministry of Education (No. 2017R1D1A1B03032785).

References

1. Ojha, V.K., Abraham, A., Snášel, V.: Metaheuristic design of feedforward neural networks: a review of two decades of research. *Eng. Appl. Artif. Intell.* **60**, 97–116 (2017)
2. Dauphin, Y.N., Pascanu, R., Gulcehre, C., Cho, K., Ganguli, S., Bengio, Y.: Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In: *Advances in Neural Information Processing Systems*, pp. 2933–2941 (2014)

3. Storn, R., Price, K.: Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **11**, 341–359 (1997)
4. Das, S., Suganthan, P.N.: Differential evolution: a survey of the state-of-the-art. *IEEE Trans. Evol. Comput.* **15**, 4–31 (2011)
5. Das, S., Mullick, S.S., Suganthan, P.N.: Recent advances in differential evolution - an updated survey. *Swarm Evol. Comput.* **27**, 1–30 (2016)
6. Choi, T.J., Ahn, C.W.: Artificial life based on boids model and evolutionary chaotic neural networks for creating artworks. *Swarm Evol. Comput.* (2017)
7. Ilonen, J., Kamarainen, J.K., Lampinen, J.: Differential evolution training algorithm for feed-forward neural networks. *Neural Process. Lett.* **17**, 93–105 (2003)
8. Slowik, A.: Application of an adaptive differential evolution algorithm with multiple trial vectors to artificial neural network training. *IEEE Trans. Ind. Electron.* **58**, 3160–3167 (2011)
9. Piotrowski, A.P.: Differential evolution algorithms applied to neural network training suffer from stagnation. *Appl. Soft Comput.* **21**, 382–406 (2014)
10. Choi, T.J., Ahn, C.W.: An improved differential evolution algorithm and its application to large-scale artificial neural networks. *J. Phys.: Conf. Ser.* **806**, 012010 (2017). IOP Publishing
11. Choi, T.J., Ahn, C.W.: Adaptive Cauchy differential evolution with strategy adaptation and its application to training large-scale artificial neural networks. In: He, C., Mo, H., Pan, L., Zhao, Y. (eds.) *BIC-TA 2017*. CCIS, vol. 791, pp. 502–510. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-7179-9_39
12. Qiu, X., Tan, K.C., Xu, J.X.: Multiple exponential recombination for differential evolution. *IEEE Trans. Cybern.* **47**, 995–1006 (2017)
13. Brest, J., Greiner, S., Boskovic, B., Mernik, M., Zumer, V.: Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. *IEEE Trans. Evol. Comput.* **10**, 646–657 (2006)
14. Zhang, J., Sanderson, A.C.: JADE: adaptive differential evolution with optional external archive. *IEEE Trans. Evol. Comput.* **13**, 945–958 (2009)
15. Choi, T.J., Ahn, C.W., An, J.: An adaptive Cauchy differential evolution algorithm for global numerical optimization. *Sci. World J.* **2013** (2013)
16. Choi, T.J., Ahn, C.W.: An adaptive differential evolution algorithm with automatic population resizing for global numerical optimization. In: Pan, L., Păun, G., Pérez-Jiménez, M.J., Song, T. (eds.) *BIC-TA 2014*. CCIS, vol. 472, pp. 68–72. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-45049-9_11
17. Choi, T.J., Ahn, C.W.: An adaptive population resizing scheme for differential evolution in numerical optimization. *J. Comput. Theor. Nanosci.* **12**, 1336–1350 (2015)
18. Choi, T.J., Ahn, C.W.: An adaptive Cauchy differential evolution algorithm with population size reduction and modified multiple mutation strategies. In: Handa, H., Ishibuchi, H., Ong, Y.-S., Tan, K.-C. (eds.) *Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems*. PALO, vol. 2, pp. 13–26. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-13356-0_2
19. Choi, T.J., Ahn, C.W.: Adaptive α -stable differential evolution in numerical optimization. *Nat. Comput.* **16**, 637–657 (2017)
20. Al-Dabbagh, R.D., Neri, F., Idris, N., Baba, M.S.: Algorithmic design issues in adaptive differential evolution schemes: review and taxonomy. *Swarm Evol. Comput.* (2018)
21. Piotrowski, A.P.: Review of differential evolution population size. *Swarm Evol. Comput.* **32**, 1–24 (2017)

22. Choi, T.J., Lee, Y.: Asynchronous differential evolution with selfadaptive parameter control for global numerical optimization. In: MATEC Web of Conferences, vol. 189, p. 03020. EDP Sciences (2018)
23. Qin, A.K., Huang, V.L., Suganthan, P.N.: Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Trans. Evol. Comput.* **13**, 398–417 (2009)
24. Choi, T.J., Ahn, C.W.: An adaptive cauchy differential evolution algorithm with bias strategy adaptation mechanism for global numerical optimization. *JCP* **9**, 2139–2145 (2014)
25. Zhabitskaya, E., Zhabitsky, M.: Asynchronous differential evolution. In: Adam, G., Buša, J., Hnatič, M. (eds.) MMCP 2011. LNCS, vol. 7125, pp. 328–333. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28212-6_41
26. Ali, M., Pant, M.: Improving the performance of differential evolution algorithm using Cauchy mutation. *Soft Comput.* **15**, 991–1007 (2011)
27. Choi, T.J., Ahn, C.W.: Accelerating differential evolution using multiple exponential Cauchy mutation. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion, pp. 207–208. ACM (2018)
28. Zaharie, D.: Influence of crossover on the behavior of differential evolution algorithms. *Appl. Soft Comput.* **9**, 1126–1138 (2009)
29. Lichman, M.: UCI machine learning repository (2013)

Author Index

- Ahmad, Wasim II-360
Ahn, Chang Wook II-388, II-397, II-477
Altangerel, Khuder I-107
- Bai, Shuai II-178
Bi, Bo I-454
Bian, Xinchao I-355
- Cai, Ci-Yun I-295
Cai, Xinye I-380
Cai, Yaqian I-496
Chen, Jiaqi II-415
Chen, Jie I-12
Chen, Jinsong II-316, II-435
Chen, Junfeng I-236
Chen, Li II-296
Chen, Xiaoji I-246
Chen, Xin I-307
Chen, Ying-Wu I-444
Chen, Zhichuan II-296
Chen, Zhihua II-63
Cheng, Jian II-178
Cheng, Shi I-236
Cheong, Yun-Gyung II-477
Choi, Tae Jong II-477
Cong, Xuwen II-201
Cui, Guangzhao II-226
Cui, Jianzhong I-151, II-55
Cui, Yang II-351
- Deng, Yali I-389
Ding, Rui I-24
Dong, Hongbin I-24
Dong, Yafei I-473
Du, Yi-Chen I-273, I-295
- Fan, Yuanyuan I-60
Fan, Zhun I-355, I-380
Fang, Wangsheng II-457
Fang, Xianwen II-72
Fang, Yi I-355
Feng, Xianbin I-24
- Ganbaatar, Ganbat I-107
Gang, Yusen II-178
- Gao, Chong II-72
Gao, Hai-rong II-308
Gao, Wenbin I-161
Geng, Shuang II-360
Gong, Dunwei I-401, II-188, II-415
Guan, Jing I-82
Guo, Miao I-12
Guo, Ping I-198, I-223
Guo, Yi-nan II-178
- Han, Gaoyong I-263
He, Chun-lin II-308
He, Jun I-24
Hu, Jianjun I-496
Hu, Long II-1
Hu, Mi I-380
Hu, Zhongdong II-457
Huang, An II-63
Huang, Chun II-13, II-129, II-162, II-287
Huang, Weidong II-351
Huang, Yifeng II-377
Huang, Yuansheng I-1, I-36, I-48, I-484, I-496
Hussain, Safdar II-360
- Ishdorj, Tseren-Onolt I-107
- Jafar, Rana Muhammad Sohail II-360
Jia, Shiyu II-1
Jiang, Keqin I-94
Jie, Wang II-264
Juanjuan, He I-173
- Kai, Zhang I-173, I-186
Kim, Jun Suk II-397
- Lei, Heng I-82
Li, Chao II-466
Li, Chen II-23
Li, Jing II-31
Li, Juan II-446
Li, Li I-94
Li, Lijie I-24
Li, Meng II-13

- Li, Nan I-285
 Li, Rui I-401, II-415, II-426
 Li, Wang II-252
 Li, Wenji I-355
 Li, Xingmei II-415
 Li, Yan II-118
 Li, Yanyue II-23
 Li, Yuan-Xiang II-446
 Li, Yuanyuan II-104
 Li, Zehua II-138
 Li, Zheng I-60
 Lin, Qiuzhen II-188
 Lin, Zhiyi I-389
 Liu, Hongwei II-118
 Liu, Huan II-316, II-328
 Liu, Hui I-1
 Liu, Jia II-239, II-435
 Liu, Jie I-70
 Liu, Lei II-239, II-328
 Liu, Qianying II-316, II-435
 Liu, Shijian I-48, I-484
 Liu, Shuaichen II-118
 Liu, Xiangrong I-133, I-307
 Lu, Hui I-236
 Lu, Xue-Qin I-273
 Lv, Aolong II-129
- Ma, Jingjing I-161
 Ma, Xin I-444
 Mei, Lin I-70
 Mo, Wanying I-142
- Ni, Yudong II-104
 Ning, Ding II-252
 Niu, Ben II-201, II-296, II-328
 Niu, Ying II-213
- Park, Dongju II-388
 Peng, Chao II-188
 Peng, Gang I-423, II-95
 Pengfei, Yu I-173
- Qi, Huaqing II-1
 Qiang, Xiaoli II-63
 Qiu, Chenye II-42
 Qu, Rong II-201, II-296
 Quan, Changsheng I-223
- Shen, Lei I-36
 Shen, Yindong II-104
 Shi, Chuan I-246
- Shi, Hongyi II-152
 Shi, Mengshu I-484
 Shi, Yuhui I-236
 Song, Feng II-338
 Song, Wu I-236
 Song, Yan-Jie I-444
 Sun, Junwei I-263, I-285, II-13, II-129,
 II-275, II-287
- Tan, Yihua II-405
 Tang, Ke I-60
 Tang, Zhen II-55
- Ventura, Michele Della I-434
- Wan, Xing II-338
 Wang, Bin II-72, II-83
 Wang, Chunlu II-152
 Wang, Dongwei II-178
 Wang, Gaiying I-133
 Wang, Haoran I-142
 Wang, Hong II-328, II-360
 Wang, Hongwei I-48
 Wang, Jun II-316, II-435
 Wang, Lingfei II-213, II-226, II-466
 Wang, Luhui I-473
 Wang, Min II-95
 Wang, Mingliang I-454
 Wang, Shengsheng I-70
 Wang, Siming I-423
 Wang, Wenbo I-213
 Wang, Xueli I-213
 Wang, Yanfeng I-263, I-285, II-13, II-129,
 II-162, II-213, II-275, II-287
- Wang, Yipeng I-12
 Wang, Yuan I-411
 Wang, Yulong I-315, I-331, I-368, II-377
 Wang, Zhaojun I-355
 Wang, Zhenyu I-389
 Wang, Zhiyu I-133
 Wei, Hu I-186
 Wei, Teng II-252
 Wei, Xiaopeng II-83
 Wei, Yani I-473
 Wu, Bin I-246
 Wu, Xiuli II-31
 Wu, Yu I-389
 Wu, Zhi II-377
- Xiang, Junqi I-198
 Xiao, Lu II-239, II-316

- Xiaoming, Liu I-173
 Xiaoxiao, Ren II-264
 Xie, Haibo II-138
 Xie, Yuehong I-389
 Xin, Bin I-12
 Xing, Lining I-411
 Xing, Li-Ning I-444
 Xing, Shanshan II-83
 Xu, Guangzhi I-401, II-426
 Xu, Siyong I-246
 Xu, Siyuan I-484
 Xu, Zhi-Ge I-343
- Yan, Pei II-405
 Yan, Peng II-264
 Yan, Xiaoshan I-133
 Yan, Xuesong I-60
 Yang, Chen II-239
 Yang, Geng II-457
 Yang, Jing I-151, II-55
 Yang, Lei I-496
 Yang, Ming I-82
 Yang, Xu-Hua I-273
 Yang, Yu II-1
 Yang, Zhenqin I-151
 Yang, Zuhuang II-95
 Ye, Lian I-223
 Yin, Zhixiang I-151, II-55, II-72
 Ying, Weiqin I-389
 Yixuan, Qiao II-264
 Yu, Mingzhu II-201, II-296
 Yu, Xiaodong I-24
 Yuan, Guodong II-287
 Yuan, Yutong I-355
 Yue, Zhao II-252
 Yunfeng, Shao I-186
- Zhang, Haoxin I-331
 Zhang, Heping II-1
 Zhang, Jiuchao I-368
 Zhang, Lijun I-484
 Zhang, Min-Xia I-295, I-343
 Zhang, Qiang II-55, II-72, II-83
 Zhang, Ruozhu I-120
 Zhang, Wenjun I-508, II-118, II-138
 Zhang, Xing I-213
 Zhang, Xuncai II-213, II-226, II-466
 Zhang, Xu-tao II-308
 Zhang, Yingying I-473
 Zhang, Yong II-308
 Zhang, Zhong-Shan I-444
 Zhang, Zhou II-1
 Zhao, Xinchao I-401, II-152, II-188, II-415, II-426
 Zhao, Xingtong II-275
 Zhaozong, Zhang I-186
 Zhen, Yiting II-1
 Zheng, Yu-Jun I-273, I-295, I-343
 Zheng, Zhonglong II-83
 Zhou, Aimin I-246
 Zhou, Changjun II-72, II-83
 Zhou, Chenyang I-423
 Zhou, Hangyu II-466
 Zhou, Kang II-1
 Zhou, Moqin I-213
 Zhou, Qinglei II-162
 Zhou, Shibo I-508
 Zhou, Xiao-Han I-343
 Zhou, Yalan II-23
 Zhou, Zheng II-213, II-226, II-466
 Zong, Hua I-315
 Zou, Jie II-446
 Zuo, Lulu II-239
 Zuo, Xingquan II-42, II-152, II-338