Control of Networked Systems with Engineering Applications

Guest Editors: Housheng Su, Michael Z. Q. Chen, Qing Hui, Yuan Fan, and Zhiwei Gao



Control of Networked Systems with Engineering Applications

Control of Networked Systems with Engineering Applications

Guest Editors: Housheng Su, Michael Z. Q. Chen, Qing Hui, Yuan Fan, and Zhiwei Gao

Copyright © 2016 Hindawi Publishing Corporation. All rights reserved.

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

Editorial Board

Mohamed Abd El Aziz, Egypt Farid Abed-Meraim, France Jose Á. Acosta, Spain Paolo Addesso, Italy Claudia Adduce, Italy Ramesh Agarwal, USA Juan C. Agüero, Australia Ricardo Aguilar-López, Mexico Tarek Ahmed-Ali, France Hamid Akbarzadeh, Canada Muhammad N. Akram, Norway Mohammad-Reza Alam, USA Salvatore Alfonzetti, Italy Francisco Alhama, Spain Mohammad D. Aliyu, Canada Juan A. Almendral, Spain Lionel Amodeo, France Sebastian Anita, Romania Renata Archetti, Italy Felice Arena, Italy Sabri Arik, Turkey Alessandro Arsie, USA Edoardo Artioli, Italy Fumihiro Ashida, Japan Hassan Askari, Canada Mohsen Asle Zaeem, USA Romain Aubry, USA Matteo Aureli, USA Francesco Aymerich, Italy Seungik Baek, USA Khaled Bahlali, France Laurent Bako, France Stefan Balint, Romania Alfonso Banos, Spain Roberto Baratti, Italy Martino Bardi, Italy Azeddine Beghdadi, France Abdel-Hakim Bendada, Canada Ivano Benedetti, Italy Elena Benvenuti, Italy Jamal Berakdar, Germany Michele Betti, Italy Jean-Charles Beugnot, France Simone Bianco, Italy Gennaro N. Bifulco, Italy

David Bigaud, France Antonio Bilotta, Italy Jonathan N. Blakely, USA Paul Bogdan, USA Alberto Borboni, Italy Paolo Boscariol, Italy Daniela Boso, Italy Guillermo Botella-Juan, Spain Abdel-Ouahab Boudraa, France Fabio Bovenga, Italy Francesco Braghin, Italy Michael J. Brennan, UK Maurizio Brocchini, Italy Julien Bruchon, France Michele Brun, Italv Javier Buldú, Spain Tito Busani, USA Raquel Caballero-Águila, Spain Pierfrancesco Cacciola, UK Salvatore Caddemi, Italy Jose E. Capilla, Spain F. Javier Cara, Spain Ana Carpio, Spain Carmen Castillo, Spain Inmaculada T. Castro, Spain Gabriele Cazzulani, Italy Luis Cea, Spain Miguel E. Cerrolaza, Spain M. Chadli, France Gregory Chagnon, France Ching-Ter Chang, Taiwan Michael J. Chappell, UK Kacem Chehdi, France Peter N. Cheimets, USA Chunlin Chen, China Xinkai Chen, Japan Francisco Chicano, Spain Hung-Yuan Chung, Taiwan Joaquim Ciurana, Spain John D. Clayton, USA Giuseppina Colicchio, Italy Mario Cools, Belgium Sara Coppola, Italy Jean-Pierre Corriou, France Juan Carlos Cortés López, Spain

Carlo Cosentino, Italy Paolo Crippa, Italy Andrea Crivellini, Italy Erik Cuevas, Mexico Peter Dabnichki, Australia Luca D'Acierno, Italy Weizhong Dai, USA Andrea Dall'Asta, Italy Purushothaman Damodaran, USA Farhang Daneshmand, Canada Fabio De Angelis, Italy Stefano de Miranda, Italy Filippo de Monte, Italy Maria do Rosário de Pinho, Portugal Xavier Delorme, France Frédéric Demoly, France Luca Deseri, USA Angelo Di Egidio, Italy Yannis Dimakopoulos, Greece Zhengtao Ding, UK Mohamed Djemai, France Manuel Doblaré, Spain Alexandre B. Dolgui, France Florent Duchaine, France George S. Dulikravich, USA Bogdan Dumitrescu, Romania Horst Ecker, Austria Ahmed El Hajjaji, France Fouad Erchiqui, Canada Anders Eriksson, Sweden R. Emre Erkmen, Australia Andrea L. Facci, Italy Giovanni Falsone, Italy Hua Fan, China Yann Favennec, France Giuseppe Fedele, Italy Roberto Fedele, Italy Jose R. Fernandez, Spain Jesus M. Fernandez Oro, Spain Eric Feulvarch, France Simme Douwe Flapper, Netherlands Thierry Floquet, France Eric Florentin, France Jose M. Framinan, Spain Francesco Franco, Italy

Elisa Francomano, Italy Mario L. Fravolini, Italy Leonardo Freitas, UK Tomonari Furukawa, USA Jose M. G. Aznar, Spain Mohamed Gadala, Canada Matteo Gaeta, Italy Mauro Gaggero, Italy Zoran Gajic, Iraq Erez Gal, Israel Ugo Galvanetto, Italy Akemi Gálvez, Spain Rita Gamberini, Italy Maria L. Gandarias, Spain Arman Ganji, Canada Xin-Lin Gao, USA Zhong-Ke Gao, China Giovanni Garcea, Italy Fernando García, Spain Laura Gardini, Italy Alessandro Gasparetto, Italy Vincenzo Gattulli, Italy Oleg V. Gendelman, Israel Mergen H. Ghayesh, Australia Agathoklis Giaralis, UK Anna M. Gil-Lafuente, Spain Hector Gómez, Spain David González, Spain Francisco Gordillo, Spain Rama S. R. Gorla, USA Oded Gottlieb, Israel Nicolas Gourdain, France Kannan Govindan, Denmark Antoine Grall, France Jason Gu, Canada Federico Guarracino, Italy José L. Guzmán, Spain Quang Phuc Ha, Australia Masoud Hajarian, Iran Frédéric Hamelin, France Zhen-Lai Han, China Thomas Hanne, Switzerland Xiao-Qiao He, China Sebastian Heidenreich, Germany Luca Heltai, Italy Alfredo G. Hernández-Diaz, Spain M.I. Herreros, Spain Vincent Hilaire, France

Roland Hildebrand, Germany Eckhard Hitzer, Japan Jaromir Horacek, Czech Republic Muneo Hori, Japan András Horváth, Italy Gordon Huang, Canada Nicolas Hudon, Canada Sajid Hussain, Canada Asier Ibeas, Spain Orest V. Iftime, Netherlands Giacomo Innocenti, Italy Emilio Insfran, Spain Nazrul Islam, USA Benoit Iung, France Payman Jalali, Finland Reza Jazar, Australia Khalide Ibilou, France Linni Jian, China Bin Jiang, China Zhongping Jiang, USA Ningde Jin, China Grand R. Joldes, Australia Dylan F. Jones, UK Tamas Kalmar-Nagy, Hungary Tomasz Kapitaniak, Poland Haranath Kar, India Konstantinos Karamanos, Belgium Driss Mehdi, France Jean-Pierre Kenne, Canada Chaudry M. Khalique, South Africa Pasquale Memmolo, Italy Do Wan Kim, Republic of Korea Nam-Il Kim, Republic of Korea Oleg Kirillov, Germany Manfred Krafczyk, Germany Frederic Kratz, France Petr Krysl, USA Jurgen Kurths, Germany Kyandoghere Kyamakya, Austria Davide La Torre, Italy Risto Lahdelma, Finland Hak-Keung Lam, UK Jimmy Lauber, France Antonino Laudani, Italy Aime' Lay-Ekuakille, Italy Nicolas J. Leconte, France Marek Lefik, Poland Yaguo Lei, China Stefano Lenci, Italy Roman Lewandowski, Poland

Panos Liatsis, UAE Anatoly Lisnianski, Israel Peide Liu, China Peter Liu, Taiwan Wanquan Liu, Australia Yan-Jun Liu, China Alessandro Lo Schiavo, Italy Jean J. Loiseau, France Paolo Lonetti, Italy Sandro Longo, Italy Sebastian López, Spain Luis M. López-Ochoa, Spain Vassilios C. Loukopoulos, Greece Valentin Lychagin, Norway Antonio Madeo, Italy José María Maestre, Spain Fazal M. Mahomed, South Africa Noureddine Manamanni, France Didier Maquin, France Paolo Maria Mariano, Italy Damijan Markovic, France Benoit Marx, France Paolo Massioni, France Gérard A. Maugin, France Alessandro Mauro, Italy Michael Mazilu, UK Roderick Melnik, Canada Xiangyu Meng, Canada Jose Merodio, Spain Alessio Merola, Italy Luciano Mescia, Italy Laurent Mevel, France Yuri Vladimirovich Mikhlin, Ukraine Aki Mikkola, Finland Hiroyuki Mino, Japan Pablo Mira, Spain Vito Mocella, Italy Roberto Montanini, Italy Gisele Mophou, France Rafael Morales, Spain Marco Morandini, Italy Philip Moser, Germany Aziz Moukrim, France Emiliano Mucchi, Italy Domenico Mundo, Italy Jose J. Muñoz, Spain

Giuseppe Muscolino, Italy Marco Mussetta, Italy Hakim Naceur, France Hassane Naji, France Dong Ngoduy, UK Tatsushi Nishi, Japan Xesús Nogueira, Spain Ben T. Nohara, Japan Mohammed Nouari, France Mustapha Nourelfath, Canada Sotiris K. Ntouyas, Greece Roger Ohayon, France Mitsuhiro Okayasu, Japan Javier Ortega-Garcia, Spain Alejandro Ortega-Moñux, Spain Naohisa Otsuka, Japan Erika Ottaviano, Italy Arturo Pagano, Italy Alkis S. Paipetis, Greece Alessandro Palmeri, UK Anna Pandolfi, Italy Elena Panteley, France Achille Paolone, Italy Xosé M. Pardo, Spain Manuel Pastor, Spain Pubudu N. Pathirana, Australia Francesco Pellicano, Italy Marcello Pellicciari, Italy Haipeng Peng, China Mingshu Peng, China Zhike Peng, China Marzio Pennisi, Italy Matjaz Perc, Slovenia Francesco Pesavento, Italy Dario Piga, Italy Antonina Pirrotta, Italy Marco Pizzarelli, Italy Vicent Pla, Spain Javier Plaza, Spain Sébastien Poncet, Canada Jean-Christophe Ponsart, France Mauro Pontani, Italy Stanislav Potapenko, Canada Sergio Preidikman, USA Christopher Pretty, New Zealand Carsten Proppe, Germany Luca Pugi, Italy Giuseppe Quaranta, Italy

Dane Quinn, USA Vitomir Racic, Italy Jose Ragot, France K. R. Rajagopal, USA Gianluca Ranzi, Australia Alain Rassineux, France S.S. Ravindran, USA Alessandro Reali, Italy Oscar Reinoso, Spain Nidhal Rezg, France Ricardo Riaza, Spain Gerasimos Rigatos, Greece Eugenio Roanes-Lozano, Spain Bruno G. M. Robert, France José Rodellar, Spain Rosana Rodriguez-Lopez, Spain Ignacio Rojas, Spain Carla Roque, Portugal Debasish Roy, India Gianluigi Rozza, Italy Rubén Ruiz García, Spain Antonio Ruiz-Cortes, Spain Ivan D. Rukhlenko, Australia Mazen Saad, France Kishin Sadarangani, Spain Mehrdad Saif, Canada Miguel A. Salido, Spain Roque J. Saltarén, Spain Francisco J. Salvador, Spain Alessandro Salvini, Italy Maura Sandri, Italy Miguel A. F. Sanjuan, Spain Juan F. San-Juan, Spain Roberta Santoro, Italy Ilmar Ferreira Santos, Denmark José A. Sanz-Herrera, Spain Nickolas S. Sapidis, Greece Evangelos J. Sapountzakis, Greece Andrey V. Savkin, Australia Thomas Schuster, Germany Mohammed Seaid, UK Lotfi Senhadji, France Joan Serra-Sagrista, Spain Gerardo Severino, Italy Ruben Sevilla, UK Leonid Shaikhet, Ukraine Hassan M. Shanechi, USA Bo Shen, Germany

Suzanne M. Shontz, USA Babak Shotorban, USA Zhan Shu, UK Dan Simon, Greece Luciano Simoni, Italy Christos H. Skiadas, Greece Michael Small, Australia Alba Sofi, Italy Francesco Soldovieri, Italy Raffaele Solimene, Italy Jussi Sopanen, Finland Ruben Specogna, Italy Sri Sridharan, USA Ivanka Stamova, USA Salvatore Strano, Italy Yakov Strelniker, Israel Sergey A. Suslov, Australia Thomas Svensson, Sweden Andrzej Swierniak, Poland Yang Tang, Germany Sergio Teggi, Italy Alexander Timokha, Norway Gisella Tomasini, Italy Francesco Tornabene, Italy Antonio Tornambe, Italy Fernando Torres, Spain Rosario Toscano, France Sébastien Tremblay, Canada Irina N. Trendafilova, UK George Tsiatas, Greece Antonios Tsourdos, UK Vladimir Turetsky, Israel Mustafa Tutar, Spain Ilhan Tuzcu, USA Efstratios Tzirtzilakis, Greece Filippo Ubertini, Italy Francesco Ubertini, Italy Hassan Ugail, UK Giuseppe Vairo, Italy Kuppalapalle Vajravelu, USA Robertt A. Valente, Portugal Pandian Vasant, Malaysia Miguel E. Vázquez-Méndez, Spain Josep Vehi, Spain Kalyana C. Veluvolu, Republic of Korea Alessandro Veneziani, USA Fons J. Verbeek, Netherlands Franck J. Vernerey, USA

Georgios Veronis, USA Anna Vila, Spain Rafael Villanueva, Spain Uchechukwu E. Vincent, UK Mirko Viroli, Italy Michael Vynnycky, Sweden Shuming Wang, China Yan-Wu Wang, China Yongqi Wang, Germany Roman Wendner, Austria Desheng D. Wu, Canada Yuqiang Wu, China Guangming Xie, China Xuejun Xie, China Gen Qi Xu, China Hang Xu, China Joseph-Julien Yamé, France Xing-Gang Yan, UK Luis J. Yebra, Spain Peng-Yeng Yin, Taiwan Qin Yuming, China Vittorio Zampoli, Italy Ibrahim Zeid, USA Huaguang Zhang, China Qingling Zhang, China Jian Guo Zhou, UK Quanxin Zhu, China Mustapha Zidi, France

Contents

Control of Networked Systems with Engineering Applications

Housheng Su, Michael Z. Q. Chen, Qing Hui, Yuan Fan, and Zhiwei Gao Volume 2016, Article ID 9596312, 2 pages

A Secure Network Coding Based on Broadcast Encryption in SDN

Yue Chen, Hongyong Jia, Kaixiang Huang, Julong Lan, and Xincheng Yan Volume 2016, Article ID 7145138, 8 pages

Multistage Warning Indicators of Concrete Dam under Influences of Random Factors Guang Yang and Meng Yang

Volume 2016, Article ID 6581204, 12 pages

Semisupervised Community Detection by Voltage Drops

Min Ji, Dawei Zhang, Fuding Xie, Ying Zhang, Yong Zhang, and Jun Yang Volume 2016, Article ID 9850927, 10 pages

Optimization of the Distribution and Localization of Wireless Sensor Networks Based on Differential Evolution Approach

Armando Céspedes-Mota, Gerardo Castañón, Alberto F. Martínez-Herrera, and Leopoldo Eduardo Cárdenas-Barrón Volume 2016, Article ID 7918581, 12 pages

Adaptation Algorithm of Geometric Graphs for Robot Motion Planning in Dynamic Environments

Jae-Han Park, Ji-Hun Bae, and Moon-Hong Baeg Volume 2016, Article ID 3973467, 19 pages

Heat Transfer Analysis and Modification of Thermal Probe for Gas-Solid Measurement

Hong Zhang and Xiangying Qi Volume 2016, Article ID 9071531, 7 pages

Global Asymptotic Stability of Switched Neural Networks with Delays

Zhenyu Lu, Kai Li, and Yan Li Volume 2015, Article ID 717513, 11 pages

A Swarm Intelligent Algorithm Based Route Maintaining Protocol for Mobile Sink Wireless Sensor Networks

Xiaoming Wu, Yinglong Wang, and Yifan Hu Volume 2015, Article ID 823909, 10 pages

Graph-Analytical Method of Determining Impedance in Electrical Transformers Zorica Bogicevic, Slobodan Bjelić, Petar Spalević, and Milan Mišić

Volume 2015, Article ID 745629, 11 pages

Robust Leaderless Consensus of Uncertain Multiagent Systems with Fast Switching Topologies Wei Liu, Shaolei Zhou, Shi Yan, and Gaoyang Yin Volume 2015, Article ID 810950, 6 pages

Networked Convergence of Fractional-Order Multiagent Systems with a Leader and Delay Yuntao Shi and Junjun Zhang

Volume 2015, Article ID 314985, 6 pages

Exploring the Impact of Network Structure and Demand Collaboration on the Dynamics of a Supply Chain Network Using a Robust Control Approach Yongchang Wei, Hongwei Wang, and Fangyu Chen

Volume 2015, Article ID 102727, 13 pages

Adaptive Second-Order Synchronization of Two Heterogeneous Nonlinear Coupled Networks Bo Liu, Shuang Li, and Xuhui Tang Volume 2015, Article ID 578539, 7 pages

Algorithm for Identifying Minimum Driver Nodes Based on Structural Controllability Reza Haghighi and HamidReza Namazi Volume 2015, Article ID 192307, 9 pages

Controllability of Multiagent Systems with a Directed Tree Li Wang and Xiao Han Volume 2015, Article ID 106769, 8 pages

Editorial **Control of Networked Systems with Engineering Applications**

Housheng Su,¹ Michael Z. Q. Chen,² Qing Hui,³ Yuan Fan,⁴ and Zhiwei Gao⁵

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

²Department of Mechanical Engineering, The University of Hong Kong, Pokfulam Road, Pokfulam, Hong Kong

³Department of Mechanical Engineering, Texas Tech University, Lubbock, TX, USA

⁴School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore

⁵Biomedical Engineering, Northumbria University, Newcastle upon Tyne, UK

Correspondence should be addressed to Housheng Su; houshengsu@qq.com

Received 19 July 2016; Accepted 19 July 2016

Copyright © 2016 Housheng Su et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recently, control of networked systems problems has attracted much attention among researchers in biology, physics, control engineering, and computer science for decades, partially due to the broad applications of control of networked systems in many engineering areas including cooperative control of mobile robots and estimation of mobile sensor networks. The study of complex network topology and modeling is to understand the influence of network structure on its function and consequently to find effective ways to improve network performance.

This special issue focuses on theoretical and technological achievements in control of networked systems. The contents of the accepted papers are summarized below.

(1) Coordinated control and controllability of networked systems: Y. Shi and J. Zhang ("Networked Convergence of Fractional-Order Multiagent Systems with a Leader and Delay") investigated the convergence of fractional-order discrete-time multiagent systems with a leader and sampling delay based on the Hermit-Biehle theorem. Z. Lu et al. ("Global Asymptotic Stability of Switched Neural Networks with Delays") presented the delay-dependent stability for a class of switched neural networks with timevarying delays via the quadratic convex combination. W. Liu et al. ("Robust Leaderless Consensus of Uncertain Multiagent Systems with Fast Switching Topologies") proposed a robust leaderless consensus algorithm of uncertain multiagent systems with directed fast switching topologies. B. Liu et al. ("Adaptive Second-Order Synchronization of Two Heterogeneous Nonlinear Coupled Networks") investigated the second-order synchronization of two heterogeneous nonlinear coupled networks by introducing adaptive control laws. L. Wang and X. Han ("Controllability of Multiagent Systems with a Directed Tree") studied the controllability of continuous-time networked systems based on the consensus protocol when the graph contains a directed spanning tree.

(2) Analysis and control of network structure: Z. Bogicevic et al. ("Graph-Analytical Method of Determining Impedance in Electrical Transformers") presented a graphanalytical method for determining the electrical impedance of alternate energy sources. R. Haghighi and H. Namazi ("Algorithm for Identifying Minimum Driver Nodes Based on Structural Controllability") investigated the structural controllability problem for networked systems, and a simple graph-based algorithm was presented to obtain the minimum driver nodes. A. Céspedes-Mota et al. ("Optimization of the Distribution and Localization of Wireless Sensor Networks Based on Differential Evolution Approach") studied the feasibility to apply the differential evolution algorithm in the presence of obstacles or strict restrictions. Y. Wei et al. ("Exploring the Impact of Network Structure and Demand Collaboration on the Dynamics of a Supply Chain Network Using a Robust Control Approach") studied the dynamics and robustness of a two-layer supply chain network of multiple retailers and multiple distributors. M. Ji et al. ("Semisupervised Community Detection by Voltage Drops") investigated a semisupervised community detection algorithm for partitioning network into groups.

(3) Applications of control of networked systems: G. Yang and M. Yang ("Multistage Warning Indicators of Concrete Dam under Influences of Random Factors") proposed multistage warning indicators of concrete dam space and considered the influences of complex random factors. J.-H. Park et al. ("Adaptation Algorithm of Geometric Graphs for Robot Motion Planning in Dynamic Environments") presented an adaptive graph algorithm for collision-free motion planning of articulated robots in dynamic environments. Y. Chen et al. ("A Secure Network Coding Based on Broadcast Encryption in SDN") proposed a network coding to improve the capacity and robustness of multicast applications on software defined networks. X. Wu et al. ("A Swarm Intelligent Algorithm Based Route Maintaining Protocol for Mobile Sink Wireless Sensor Networks") investigated a novel route maintaining protocol based on the immune based artificial bee colony for the mobile sink wireless sensor networks. H. Zhang and X. Qi ("Heat Transfer Analysis and Modification of Thermal Probe for Gas-Solid Measurement") studied a gas-solid mass flow-rate measurement system and applied a novel modified thermal probe.

Note that although the selected topics and papers are not a comprehensive representation of the areas covered by the special issue, the published papers in this special issue do provide some recent advances in the field of control of networked systems, which could benefit the current research in some way.

Acknowledgments

We would like to thank all the authors for their contributions and acknowledge all the reviewers for their time and effort in assessing the manuscripts.

> Housheng Su Michael Z. Q. Chen Qing Hui Yuan Fan Zhiwei Gao

Research Article A Secure Network Coding Based on Broadcast Encryption in SDN

Yue Chen,¹ Hongyong Jia,¹ Kaixiang Huang,¹ Julong Lan,² and Xincheng Yan¹

¹State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450001, China
²China National Digital Switching, System Engineering and Technological Research Centre, Zhengzhou 450002, China

Correspondence should be addressed to Kaixiang Huang; kx.huang@outlook.com

Received 30 October 2015; Revised 22 May 2016; Accepted 7 July 2016

Academic Editor: Aime' Lay-Ekuakille

Copyright © 2016 Yue Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

By allowing intermediate nodes to encode the received packets before sending them out, network coding improves the capacity and robustness of multicast applications. But it is vulnerable to the pollution attacks. Some signature schemes were proposed to thwart such attacks, but most of them need to be homomorphic that the keys cannot be generated and managed easily. In this paper, we propose a novel fast and secure switch network coding multicast (SSNC) on the software defined networks (SDN). In our scheme, the complicated secure multicast management was separated from the fast data transmission based on the SDN. Multiple multicasts will be aggregated to one multicast group according to the requirements of services and the network status. Then, the controller will route aggregated multicast group with network coding; only the trusted switch will be allowed to join the network coding by using broadcast encryption. The proposed scheme can use the traditional cryptography without homomorphy, which greatly reduces the complexity of the computation and improves the efficiency of transmission.

1. Introduction

The inflexible transport mode underlining today's network restricts the development of the networks. Its capability and structure are not well suited to the requirement of all sorts of emerging transmission services. The gap between service requirements and basic network capabilities becomes bigger. Increasing efforts have been devoted to finding more reconfigurable network, such as the software defined network (SDN) [1, 2] (e.g., OpenFlow [3, 4]) and the Flexible Architecture of Reconfigurable Infrastructure (FARI [5]).

Multicast [6] is an efficient way of disseminating information to large groups of users and plays a significant role in network services. Unfortunately, the traditional IP multicast protocols are very complex, because the router needs to not only transmit the data but also forward the control messages to get the group membership and the global network information [7]. In the new network paradigm (e.g., SDN), through separating data plane and control plane, the logically centralizing controller can control the behavior of the entire network and the router only to forwarding data. To design a scalable and secure approach for multicast is significant in the future network.

Network coding [8] is a novel transmission mechanism that allows intermediate nodes to encode the received packets, improve the capacity of multicast applications, and enhance network robustness and throughput relative to the traditional "store-and-forward" [9]. However, if there are some malicious nodes in the networks, and they forward fake vectors or invalid combinations of received vectors, the polluted vectors will be quickly spread to the other nodes and even the whole network. Only part of the multiple packets obtained by the receivers are the uncorrupt vectors. Recently, to provide a secure random linear network coding [10], most schemes need to use the homomorphic cryptographic technology to sign or hash the original data. The development of SDN brings a new method in providing the secure services.

In this paper, we propose a scheme called secure switch network coding (SSNC). Multiple multicast groups will be aggregated to one multicast group according to the requirements of services and the network status. Then, the controller will route this multicast group with network coding. Those



FIGURE 1: The pollution attacks.

switches in the same path will get the same attribute. When the data packet enters the switch in the SDN and meets the transaction's resource requirements, the switch can decrypt the received data packets and then combine the packets according to the encoding matrix. So the SSNC advances in the following aspects: (a) the controller is responsible for managing the multicast group that it can prevent attackers sending data and illegal users receiving the data; (b) the controller will authenticate and authorize the switch by only allowing trusted switches meet with the services requirements to join the path of the multicast tree; (c) the switches only forward the data according to the flow entry; and (d) we meet this challenge by using broadcast encryption and AES without homomorphism.

2. Related Work

In a typical linear network coding scenario [11], the sender first breaks the message into sequence vectors in an *n*dimensional linear space \mathbb{F}_q^n . The sender transmits these message vectors to its neighboring nodes. The intermediate nodes will randomly and linearly combine the arriving vectors according to the local encoding matrix and then forward the new vectors to their adjacent nodes. Receivers can recover the original messages from the sufficient number of arriving packets by the encoding matrix.

When malicious nodes forward fake vectors or invalid combinations of received vectors, the receivers will have no way to decode vectors [12], and network resources are wasted. So it is crucial to prevent pollution attacks in practical applications of network coding. From Figure 1, pollution attacks cannot be mitigated by standard signatures or MACs. Because the receivers do not have the original message vectors, they cannot verify the signature of the original vectors. And there is no use to sign the entire message before transmission.

To defend against pollution attack in NC, an increasing number of researchers have proposed several novel hashing or signature schemes. Zhao et al. [13] introduced a signature scheme that breaks a file into a number of blocks viewed as vectors spanning a subspace V. Each node verifies the integrity of a received vector w by checking the membership of w in V based on the signature on V. The intermediate nodes only need to verify the signature without doing anything else. But in this scheme, the signature is too long, the public key is only used for a single file, and the sender needs to know the entire file before generating the authentication information. Therefore, Boneh et al. [14] proposed homomorphic signature schemes with better performance that both public key size and per-packer overhead are constant. This scheme signs individual vector instead of the entire subspace, but it suffers from highly expensive computational overhead due to the operations of bilinear pairing.

Yu et al. [15] proposed a probabilistic key predistribution and message authentication codes to defend against pollution attacks. The sources need to add multiple MACs to the data before forwarding them. Besides multiple nodes can verify the different parts of the message via shared secret keys. Agrawal et al. [16, 17] designed a homomorphic MAC system which allows checking the integrity of network coded data. But this system is collusion resistant up to a predetermined collusion bound c and is vulnerable to the tag pollution attacks. Based on the idea of [18], Li et al. [19] proposed a RIPPLE, a symmetric key based in-network solution for network coding authentication, which is on the basis of the homomorphic Mac and TESLA [20]. In RIPPLE, for calculating the MAC labels, the sources must know the longest path from each node to the source. Wu et al. [21] designed a key predistribution-based tag encoding scheme KEPTE. The source will generate multiple tags for each packet. Besides, the intermediate nodes will generate a new tag according to the received tags and then verify the correctness of the packet. But there must be a key distribution center in this scheme.

In the SDN architecture, the complicated secure multicast management could be separated from the fast data transmission [22, 23]. We need to verify that the switch has some authenticated attributes. So the session keys should only be bold by the switch with those attributes. The broadcast encryption can be used to distribute the session key to those switches. When a switch receives a broadcast message, it can verify that the message really comes from the controller. Because the session key of the packet and the attributes of switch are authenticated and issued by the PKG, thus the only secure switch can combine received packets according to the encoding matrix and then encrypt and forward the new packet by the flow entry. So the packet can be forwarded in the secure switches in the SDN.

3. Broadcast Encryption

, is

The main construction of broadcast encryption for *n* users is as follows:

Bro.Setup(*n*): In this broadcast encryption scheme, a trustee is needed to generate some public information used in the following construction. The trustee chooses proper bilinear group \mathbb{G} of prime order *p*. Choose random values: $g \leftarrow \mathbb{G}$; $\alpha \leftarrow \mathbb{Z}_p$, and set

$$g_i = g^{(\alpha')} \in \mathbb{G}$$
for $i = 1, 2, ..., n, n + 2, ..., 2n, v = g^{\gamma} \in \mathbb{G}$,
(1)

where a random $\gamma \in \mathbb{Z}_p$. The public key is

$$PK = (g, g_1, \dots, g_n, g_{n+2}, \dots, g_{2n}, \nu) \in \mathbb{G}^{2n+1}.$$
 (2)

The private key for user $i \in \{1, ..., n\}$ is set as $d_i = g_i^{\gamma} \in \mathbb{G}$. Note that $d_i = v^{(\alpha^i)}$.

Output the public key PK and the *n* private keys d_1, \ldots, d_n .

Bro.Encrypt(*S*, PK): Choose random values, $t \leftarrow \mathbb{Z}_p$, and set

 $K = e(g_{n+1}, g)^t \in \mathbb{G}$; the value $e(g_{n+1}, g)$ can be computed as $e(g_n, g_1)$.

Then set

$$\mathrm{Hdr} = \left(g^{t}, \left(v \cdot \prod_{j \in S} g_{n+1-j}\right)^{t}\right) \in \mathbb{G}^{2}, \tag{3}$$

and output the pair (Hdr, *K*).

Bro.Derypt(S, i, d_i , Hdr, PK): Let Hdr = (C_0 , C_1). Then, output

$$K = \frac{e(g_i, C_1)}{e(d_i, \prod_{j \in S, j \neq i} g_{n+1-j+i}, C_0)}.$$
 (4)

A private key is only one group element in \mathbb{G} , and the ciphertext Hdr is only two group elements. The ciphertexts and private keys are of constant size, but the public key grows linearly in the number of users.

4. SSNC Authentication Method

We mainly use the network coding to implement the multicast application. In the SDN, a legal sender can generate and send out original packets without being aware of the encoding/decoding affairs, whereas a legal recipient can receive these packets from the network. For multicast transmission, the controller is responsible for deciding whether to use the network coding. When multiple multicast trees are aggregated to use the network coding for transmission, the controller is then responsible for routing the paths, updating the flow table of the switches, and calculating the local encoding matrix for each switch involved in the transmission. Finally, the controller generates the flow-table entries of the network coding and sends these to each switch via the interface.

Figure 2 shows that the control layer mainly consists of the control server. In the controller, the general module is responsible for basic operation and management, such as knowing the global network topology, generating and updating the flow entry, and monitoring the state awareness. To create a strategy, the general module also exchanges basic parameters with the other modules. The state-aware module monitors and analyzes the network state to promptly discover the abnormal switch nodes and current traffic distribution. The network-topology module obtains the whole network structure to provide a global view. The flow-entry module generates and updates the flow-table entries of the switches. The multicast module manages the group members, the routing of multicast trees, the decision of whether to use network coding for multicast, the selection of networkcoding path, and the generation of a coding matrix based on

the parameter information provided by the general module. The security module mainly generates and manages the certificates and completes the authorization and authentication of the member switches by cooperating with the other modules.

The data layer is composed of switches and users, which are only involved in data transmission. In the switch, the encoding module is responsible for encoding the data according to the local encoding matrix assigned by the controller. The decoding module only runs when the switch is connected to the receiver. It decodes according to the decoding matrix and then sends the data to the receiver. The security module stores private keys and certificates and simultaneously verifies the credibility of the data as well as its own credibility by connecting with the controller.

The network coding needs to establish the forward paths between all the senders and receivers. Besides, for traditional networks, it is difficult to know all the nodes on the path. So most researches sign the source data or the spanning vectors space and ensure the security through the source authentication in each node. If the switches on the path are all real and trusted during the whole data transmission, other switches can be effectively prevented from replaying and tampering attacks. Authentication, authorization, and integrity checks must be provided in network coding transmission. Only authenticated and authorized switches are used.

4.1. System Setup. The trusted controller runs the **Bro.Setup**(n) algorithm in SDN. Input the number of switches n in the SDN. Then every switch gets the system setup outputs, public PK, and its private key d_i .

4.2. Switch Session Keys Distribution. Multiple multicast groups will be aggregated to one group according to the quality of service level and the network status. Then the controller routes this group by network coding, to get the subset $S \subseteq \{1, ..., m\}$, and the switch $S_i = i$ belongs to this path. Input the public key PK and a subset S; the **Bro.Encrypt**(S, PK) algorithm outputs a pair (Hdr, K), where Hdr is the broadcast ciphertext, and the key $K \in \mathcal{K}$ is a symmetric multicast session key. Because a malicious node can fake a pair (Hdr, K), the controller chooses a collisionresistant hash function **H**, and sign the hash value of Hdr: $\sigma =$ enc(h(Hdr), Private.rsa) by the RSA algorithm. So the message (S, Hdr, σ) is broadcast to the set S:

Controller
$$\longrightarrow$$

 $S_1 : \{S, Hdr, \sigma\},$
 $S_2 : \{S, Hdr, \sigma\},$
 \vdots
(5)

4.3. Multicast Data Forwarding. When the switch gets the broadcast message (S, Hdr, σ) , it is necessary to verify the identity of the controller. So the switch decrypts the σ to get the value $h(\text{Hdr}) = \text{dec}(\sigma, \text{Public.rsa})$ and computes and checks the h(Hdr) by the function **H** as the input Hdr.



FIGURE 2: The scheme of SSNC.

After the message passes the verification, input a subset $S \subseteq \{1, ..., n\}$, the switch id $i \in \{1, ..., n\}$, and the private key d_i for the switch i, a header Hdr, and the public key PK, if $i \in S$. Then, the **Bro.Derypt**(S, i, d_i , Hdr, PK) algorithm outputs the multicast session key $K \in \mathcal{K}$. The key K can then be used to encrypt/decrypt the multicast data. To improve the performance of the multicast transmission, the encryption function can be embedded in hardware.

In SSNC, after the secure multicast forwarding tree is established, each switch, sender, and receiver in the multicast paths will get the session key *K*. For the sender, the original data is represented by a vector $V = (v_1, v_2, ..., v_l)$. The data vector should be encrypted by the key *K*, denoted as $\{v_i\}K$. The different data $\{v_i\}K$ is sent to the different next switch *S*:

Sender
$$\longrightarrow$$

 $S_1 : \{v_1\} K,$
 $S_2 : \{v_2\} K,$
 \vdots
(6)

The switches located in the same path as the sender have the same requirement of the quality of service and security level. Only when they have the same session key K will the switches decrypt all receiving packets by using the session key K and then linearly combine the received data vectors to a new data vector y based on the predistributed encoding matrix $M_i = \{\alpha_1, \alpha_2, ..., \alpha_l\}^T$. After that, the switch encrypts and forwards the vector y to the next switches:

$$y = \sum_{i=1}^{l} \alpha_{i} v_{i}$$

$$S \longrightarrow$$

$$S_{\text{NEXT1}} : \{y\} K,$$

$$S_{\text{NEXT2}} : \{y\} K,$$

$$\vdots$$

$$(7)$$

For the receiver located in the same path as the sender, once receiving a packet, it will decrypt data using the session key *K* and encode the packets by network coding to get the original data.

4.4. *Rekeying.* When the switch is broken down or cannot meet the quality of service level or the security requirements, this switch should be removed from the subset *S*. The controller may need to route new path for the aggregated multicast to replace the problem. To enhance the security of the multicast, the controller also needs to rekey per certain period. Therefore, the controller is required to generate new keys for the subset *S*. When new keys are generated, these keys will be broadcasted to all switches. But only the switch belonging to the subset *S* can decrypt the new keys.

4.5. Maintenance of Multicast Path. When the multicast groups are determined, the controller is responsible for aggregating multicast groups and deciding the usage of network coding. And if it decides to use the network coding for transmission, it will route this group and provide secure and reliable switchers to construct the transmission path of multicast. It is very important to establish the trust relationship between the switch and the controller [24]. The controller can keep white lists for trusted and authenticated devices and build the multicast paths from this list. The list is dynamically changed based on anomaly/failure detection algorithms [25]. If the trustworthiness of the switch is questioned or has abnormal behavior, the switch will be reported by other switches or controllers. So this switch will be automatically quarantined by other switches and controllers and thus removed from the list.

5. Performance and Security Analysis

5.1. Security Analysis. The set of the whole network is denoted by \mathcal{N} , the number of all switches is *n*, the set of the switches in one secure multicast path is denoted by \mathcal{S} ($\mathcal{S} \subset \mathcal{N}$), the number of switches is *s*, the number of the multicasts in the network is *l*, and the attacker is denoted by \mathcal{A} .

5.1.1. Eavesdropping Attack. An eavesdropping attacker can wiretap one or more links in the network. It also refers to the information not leaked to the unauthorized users. In the network coding, the original data will be divided into many data vectors before being sent out. And the data vectors will be combined into new data vectors. The original data can be decoded only if \mathcal{A} got enough data vectors and the corresponding coding matrix. To make the eavesdropper not get the correct data, traditional network coding needs encrypting coefficients or partial data. But the encryption scheme usually needs to be homomorphic. In the SSNC, the coefficients of coding are predistributed to the trusted switch by the controller instead of being transited with the data. And the multicast packets are transmitted with the ciphertext encrypted by AES. Each of the trusted switches should be authenticated by the controller, but \mathcal{A} may fake the controller to send the session K. Thereby, in the SSNC, we provide the source authentication when broadcasting the session K by the RSA.

5.1.2. Pollution Attack. This attack is usually stated by unauthorized nodes which inject polluted packets into the information flow. An attacker \mathscr{A} fakes a message $\{v^*\}K^*$. Only if the switch of \mathscr{S} has the fake session key K^* will it receive the data v^* . But \mathscr{A} has to crack secret keys of the AES. Without cracking the AES, \mathscr{A} has to fake the trusted switch to join the secure multicast group or fake the controller to distribute the session key K^* to the switches of \mathscr{C} . We can use the RSA technology to resolve the second problem in Section 4.2. To resolve the first problem, the switch should be authenticated and authorized by controller at the first time, when it connects to the controller. And with the system running, the controller will authenticate and authorize them at regular intervals.

The security of the whole system depends on that of the RSA, AES, and broadcast encryption used in our scheme. From the above analyses, it can be concluded that our system is capable of resisting pollution attack and the eavesdrop attack.

5.2. Performance Analysis. The communication overhead of the network coding refers to the bandwidth cost for distributing the authentication information to each switch. Without the data vectors, the overhead mainly considers two aspects, the keys and signatures. The computational overhead mainly refers to encoding/decoding the vector and verifying the signatures. The storage overhead refers to store keys, certificates, and other safety parameters.

In the traditional secure network coding, homomorphic hash, signature, and MAC are frequently used methods to defend against the pollution attack. However, they are always expensive in computation. The advantages of the network coding are only reflected in the networks consistent with faithful switches [26]. So in the SSNC, to take the advantages of the SDN, the control layer completes the complex and expensive computations. The whole secure network coding is divided into two stages. The first stage is to set up the system. The controller will select the trusted switch meeting the requirement of QoS for the set of S from the network N. By using the broadcast encryption, only the secure switch can get the session K. The controller also needs route the multicast and generates and distributes the keys. In the second stage, the data layer only needs to encode and forward the data by the session K.

Each intermediate switch only needs to store its session keys and the controller public key. While the controller provides the source authentication service, the switch stores the controller public key, whose size is denoted as $|K_R|$. The session key is shared by one group for one key. We denote the size of session key as $|K_A|$. For the switch, the total storage overhead is $|K_R| + |K_A| * c$, where *c* is the number of the switches belonging to the secure group. For the controller, it stores a pair of keys of RSA and all the sessions keys, so that the total is $|K_R| * c + |K_A| * l$.

In the SSNC scheme, the encoding matrix and the session keys will be distributed to each switch before the stage of transferring the data. So the packets do not need to carry any more information. We denote the size of the data vector as |v|.

The performance of the SSNC will depend on the setup stage, as in the forwarding stage, we can encrypt the message by the hardware. So it is very important for routing the multicast by the network coding. When the group changes very frequently, the SSNC will pay more cost in the setup stage, and the performance maybe became poor. In the worst case, it should update the keys to the whole network. To reduce the change rate of the multicast paths, we aggregate the same class multicasts to one aggregated path.

To demonstrate the effectiveness of SSNC, we compared the following three schemes (Table 1).

In the scheme presented by Ho et al. in [11], the source executes signature algorithm, while the intermediate nodes execute combinations and verification algorithms. From their

TABLE 1: The comparison of computation, storage, and communication overhead.

Scheme	Computation	Storage	Communication
This paper	$(n_{\text{iuput}} + 1)t_{\text{AES}}$ (lowest)	$\begin{split} K_{R} + K_{A} * \\ c (\text{controller}) \\ K_{R} + K_{A} * \\ l (\text{switch}) \\ (\text{lowest}) \end{split}$	NULL (lowest)
[11]	$n_{\text{iuput}} * t_{\text{PRF}} + t_{\text{PRG}}$ (highest)	$ K_{\text{HMAC}} *n*l$ (highest)	tag + y + P (middle)
[7]	$n_{ m iuput} * t_e$ (middle)	$ K_{\text{space}} * l$ (middle)	$ K_{\text{space}} + y _{\text{DSA}}$ (highest)
[8]	$n_{ ext{iuput}} * t_e$ (middle)	K _{NCS} * <i>l</i> (middle)	$ y _{\rm NCS} + K_{\rm NSC} + P$ (middle)

experiment, it can be found that the time overhead of signature algorithm is much longer than the process of combination and verification. The intermediate nodes need to use Random Generator algorithm and Pseudo Random Function algorithm to combine the received data. The time overhead of the two algorithms is denoted as t_{PRG} and t_{PRF} . In the SSNC, the node will firstly decrypt the vectors and then combines the vectors. But in the scheme in [11], the switch firstly combines the vectors and then verifies the vectors. In this case, it is hard to tell which vector is forged. Besides, it needs to generate n keys for one multicast. Each switch distributes different keys. So it requires a complicated key management mechanism. The size of key is denoted as $|K_{HMAC}|$. Each node needs to store two sets of keys to ensure that it can act as both of the data source and receiver node. In contrast with scheme in [11], SSNC key management is more simple and effective. In the phase of data transmission, besides the original vector, it also needs to attach tags *t*, the extension of the vector data, and the data source id. The size of the extended vector and tag are, respectively, denoted as |y| and |tag|, which is set in advance. The id of data source is a constant and denoted as P. This scheme needs to distribute keys before transmission.

In the scheme in [7], the source will find an orthogonal vector and signatures. The intermediate node will verify the signatures of which the security is based on the Diffie-Hellman problem. This computation overhead is denoted as $|t_e|$. The complexity of this algorithm is so high. This scheme uses the public key encryption scheme. The source preserves a private key and distributes the public key. Although the number of needed keys is not so much, the size of the keys is related to the size of files. We denote the size of public and secret key as $|K_{\text{space}}|$. The size of public key is 6(g + h)/gh times that of the file, where the file with size *M* is divided into *g* blocks and *h* is the size of each block. During the transmission, the main communication overhead is public key and algorithm, whose size is denoted as $|y|_{\text{DSA}}$.

In [8], the scheme combines linear subspace with homomorphic signature and uses constant size of public key and signature. It uses constant size of public key and signature.



FIGURE 3: The number of the control messages.

The internal nodes verify the signature according to public key, id, and the dimensionality of signed vector. The complexity of the calculation is similar to scheme in [7] but storage overhead is much smaller. The size of the keys is constant and denoted as $|K_{\text{NCS}}|$. Besides, the size of signature in this scheme is constant, which usually takes up $g \log_2 p$ bits or more, denoted as $|y|_{\text{NCS}}$.

Comparing with three classical solutions, the key generation is more simple and the key management is more effective in our scheme.

5.3. Network Experiment Performance. We evaluated the performance of our scheme through an experimental network. We tested SSNC on Mininet, one of the popular network simulation platforms. Each switch is connected with a receiver. Besides, one controller is connected with all switches. The controller will run the algorithm of network coding [27] according to the topology, broadcast the session keys, and update the flow entry of the switch. In the traditional secure network coding, the controller only needs to broadcast the keys. Therefore in the scheme we presented, the number of control messages increases with the number of switches in the network coding path. We emulated 5 aggregated multicast sessions. The number of the control messages is shown in Figure 3.

Before sending the multicast data packets, the switch decryption time will be dominated by the |S| - 2 group operations, which need to compute $\prod_{j \in S, j \neq i} g_{n+1-j+i}$. This takes plenty of time. But when the switch forwards the data packets, it only needs to run the AES algorithm. The hardware will accelerate the completion of the AES algorithm. In this experiment, we used Dell desktop computers with 2.93 GHz Intel core i3 530 CPU and 4 GB main memory. The sender will send the messages to three receivers in different network topology. We timed the following operations:

- Broadcasting the session keys: the controller calculates the forward tree and the session *K*; the switch decrypts the broadcast message to get the key.
- (2) Forwarding the data packets: switch decrypts and encrypts the packets using the AES.

TABLE 2: Time of the two stages.

Network topology	Setup	Forward
Switch = 20 , receiver = 3	258.1 (ms)	4.6 (ms)
Switch = 25 , receiver = 3	578.2 (ms)	4.1 (ms)
Switch = 30 , receiver = 3	797.4 (ms)	4.1 (ms)

The results of multicast are shown in Table 2. We can see that the first packet of the multicast session will take more time to arrive to the receiver, but the total time of the session will take less time.

6. Conclusions

In this paper, a novel fast and secure switch network coding multicast on the software defined networks is proposed. It separates the secure management from the transmission. During the management stage, the controller routes the aggregated multicast and broadcasts the session keys. Only the secure switches meeting the requirement of quality of service can join in the multicast forwarding tree. Our scheme ensures that the switch verifies the data from other trusted switches, which greatly improves the ability to prevent the pollution and eavesdropping attacks. During data transmission, the node only focuses on how to forward the data to keep the high performance of the network coding. In order to provide more flexible and efficient multicast services, further efforts are needed to improve the system dynamics, especially on how to deal with the frequent changes of members.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work is sponsored by the National Basic Research Program of China (973 Program) under Grant no. 2012CB315901 and National Natural Science Foundation of China (no. 61373006).

References

- B. A. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: past, present, and future of programmable networks," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 3, pp. 1617–1634, 2014.
- [2] N. McKeown, T. Anderson, H. Balakrishnan et al., "OpenFlow: enabling innovation in campus networks," ACM SIGCOMM Computer Communication Review, vol. 38, no. 2, pp. 69–74, 2008.
- [3] R. Cohen, L. Lewin-Eytan, J. S. Naor, and D. Raz, "On the effect of forwarding table size on SDN network utilization," in *Proceedings of the 33rd IEEE Conference on Computer Communications (IEEE INFOCOM '14)*, pp. 1734–1742, Toronto, Canada, May 2014.

- [4] A. Lara, A. Kolasani, and B. Ramamurthy, "Network innovation using open flow: a survey," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 493–512, 2014.
- [5] J. Lan, G. Xiong, Y. Hu et al., "Research on the architecture of reconfigurable fundamental information communication network," *Telecommunications Science*, vol. 31, no. 4, 2015.
- [6] E. Deering S, Multicast Routing in Internetworks and Extended LANs, ACM, 1988.
- [7] S. Keshav and S. Paul, "Centralized multicast," in Proceedings of the 20th IEEE International Conference on Network Protocols (ICNP '12), p. 59, IEEE Computer Society, 2012.
- [8] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, 2000.
- [9] S.-Y. R. Li, N. Cai, and R. W. Yeung, "On theory of linear network coding," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '05)*, pp. 273–277, Adelaide, Australia, September 2005.
- [10] K. Han, T. Ho, R. Koetter, M. Médard, and F. Zhao, "On network coding for security," in *Proceedings of the Military Communications Conference (MILCOM '07)*, pp. 1–6, IEEE, Orlando, Fla, USA, October 2007.
- [11] T. Ho, M. Médard, J. Shi et al., "On randomized network coding," in *Proceedings of the Allerton Conference on Communication*, 2003.
- [12] S. Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Transactions on Information Theory*, vol. 49, no. 2, pp. 371– 381, 2003.
- [13] F. Zhao, T. Kalker, M. Médard, and K. J. Han, "Signatures for content distribution with network coding," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '07)*, pp. 556–560, Nice, France, June 2007.
- [14] D. Boneh, D. Freeman, J. Katz et al., "Signing a linear subspace: signature schemes for network coding," in *Proceedings of the* 12th International Conference on Practice and Theory in Public Key Cryptography (PKC '09), pp. 68–87, Springer, 2009.
- [15] Z. Yu, Y. Wei, B. Ramkumar, and Y. Guan, "An efficient scheme for securing XOR network coding against pollution attacks," in *Proceedings of the IEEE INFOCOM*, pp. 406–414, IEEE, Rio de Janeiro, Brazil, April 2009.
- [16] S. Agrawal and D. Boneh, "Homomorphic MACs: MAC-based integrity for network coding," in *Applied Cryptography and Network Security*, Lecture Notes in Computer Science, pp. 292– 305, Springer, Berlin, Germany, 2009.
- [17] S. Agrawal, D. Boneh, X. Boyen, and D. M. Freeman, "Preventing pollution attacks in multi-source network coding," in *Public Key Cryptography—PKC 2010*, vol. 6056 of *Lecture Notes in Computer Science*, pp. 161–176, Springer, Berlin, Germany, 2010.
- [18] J. Dong, R. Curtmola, and C. Nita-Rotaru, "Practical defenses against pollution attacks in intra-flow network coding for wireless mesh networks," in *Proceedings of the 2nd ACM Conference on Wireless Network Security (WiSec '09)*, pp. 111–122, ACM, March 2009.
- [19] Y. Li, H. Yao, M. Chen, S. Jaggi, and A. Rosen, "RIPPLE authentication for network coding," in *Proceedings of the IEEE INFOCOM*, pp. 1–9, San Diego, Calif, USA, March 2010.
- [20] H. Krawczyk and R. Canetti, "The TESLA broadcast authentication protocol," *Rsa Cryptobytes*, vol. 20, no. 2, article 2002, 2002.

- [21] X. Wu, Y. Xu, C. Yuen, and L. Xiang, "A tag encoding scheme against pollution attack to linear network coding," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 1, pp. 33– 42, 2014.
- [22] U. C. Kozat, G. Liang, and K. Kokten, "On diagnosis of forwarding plane via static forwarding rules in Software Defined Networks," in *Proceedings of the 33rd IEEE Conference on Computer Communications (IEEE INFOCOM '14)*, pp. 1716–1724, IEEE, Toronto, Canada, May 2014.
- [23] R. Cohen, L. Lewin-Eytan, J. Naor S, and D. Raz, "On the effect of forwarding table size on SDN network utilization," in *Proceedings of the IEEE Conference on Computer Communications* (INFOCOM '14), pp. 1734–1742, IEEE, Toronto, Canada, April-May 2014.
- [24] D. Kreutz, F. M. V. Ramos, and P. Verissimo, "Towards secure and dependable software-defined networks," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking (HotSDN '13)*, pp. 55–60, ACM, Hong Kong, August 2013.
- [25] Z. Yan and C. Prehofer, "Autonomic trust management for a component-based software system," *IEEE Transactions on Dependable & Secure Computing*, vol. 8, no. 6, pp. 810–823, 2011.
- [26] S. Yao, J. Chen, R. Du, L. Deng, and C. Wang, "A survey of security network coding toward various attacks," in *Proceedings* of the 13th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom '14), pp. 252–259, IEEE Computer Society, Beijing, China, September 2014.
- [27] S. Jaggi, P. Sanders, P. A. Chou et al., "Polynomial time algorithms for multicast network code construction," *IEEE Transactions on Information Theory*, vol. 51, no. 6, pp. 1973–1982, 2005.

Research Article

Multistage Warning Indicators of Concrete Dam under Influences of Random Factors

Guang Yang^{1,2} and Meng Yang^{1,3}

¹State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering, Hohai University, Nanjing 210098, China
²National Engineering Research Center of Water Resources Efficient Utilization and Engineering Safety, Hohai University, Nanjing 210098, China

³College of Water-Conservancy and Hydropower, Hohai University, Nanjing 210098, China

Correspondence should be addressed to Meng Yang; ymym_059@126.com

Received 25 November 2015; Revised 25 February 2016; Accepted 21 March 2016

Academic Editor: Paolo Crippa

Copyright © 2016 G. Yang and M. Yang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Warning indicators are required for the real-time monitoring of the service conditions of dams to ensure safe and normal operations. Warnings are traditionally targeted at some "single point deformation" by deformation measuring points of concrete dam, and scientific warning theory on "overall deformation" measured is nonexistent. Furthermore, the influences of random factors are not considered. In this paper, the overall deformation of the dam was seen as a deformation system of single interactional observation points with different contribution degrees. The spatial deformation entropy, which describes the overall deformation, was established and the fuzziness indicator that measures the influence of complex random factors on monitoring values according to cloud theory was constructed. On this basis, multistage warning indicators of "spatial deformation" that consider fuzziness and randomness were determined. Analysis showed that the change law of information entropy of the dam' overall deformation is identical to the real change law of the dam; thus, it reflects the real deformation state of the dam. Moreover, the identified warning indicators improved the warning ability of concrete dams.

1. Introduction

Deformation is one of the major monitored items in dam safety. Concrete dams are exposed to influences of various nondeterministic settings such as the load effect of water level, uplift pressure, and wind waves caused by hydrologic and hydraulic uncertainties, as well as geological and material uncertainties such as shearing and compressive strength. Thus, a concrete dam is a complicated system of nondeterministic settings that are affected by various complex random factors [1, 2]. Considering a dam's long-term service, conducting timely and effective warning against emergencies through real-time monitoring is key to its safe operation [3].

The monitoring of dam safety is an important research subject in advanced mechanics and mathematics theories. In 1950, Tonini categorized the factors influencing the displacement of dam into water pressure, temperature, and effectiveness for a given period [4]. These factors were expressed in the polynomials of specific functions before a statistic model with regression analysis was established. Then, the deterministic model and mixed model were consulted for deformation of concrete dam and introduced finite element to monitor and evaluate the safety of dams [5]. Furthermore, many scholars brought new achievements in diagnosing dam safety from many aspects. In 2009, Gu and Wang established the catastrophe model of time-dependent component on the basis of catastrophe theory and proposed the method to determine the threshold value of the structure displacement of the dam [6]. On the basis of the POT model in extreme value theory, in 2012, Su et al. estimated the warning indicators by setting the threshold value and combining the probability of dam deformation with transfinite data sequence as the subject of modeling analysis [7]. Warning indicators are sure to have some fuzziness and randomness because of the influences of various complex random factors. On the basis of the fuzzy finite element, Chen (2006) realized the nondeterministic optimal control on roller compacted concrete dam [8]. Although the above-mentioned theories and methods complement and improve traditional methods in solving difficult problems in dam safety, such approaches only address the warning of "point" deformation and not the scientific "spatial and overall" deformation. Further studies on the deformation of concrete dams must consider the influences of complex random factors. Thus, the expression of a dam's overall deformation should be constructed and scientific and accurate warning indicators that consider randomness and fuzziness should be determined.

This paper began with an analysis of an indicator of fuzziness that affects the value of monitoring the dam by studying the influences of fuzziness and randomness from long-term service of the dam based on cloud theory [9-11]. Thereafter, the relationship between a dam's overall deformation and deformation of single observation points was analyzed through information entropy and synergetics. In the analysis, overall deformation refers to a system wherein single observation points with different contribution degrees (weights) influence one another. A criterion indicator measuring the overall deformation conditions was constructed. On this basis, multistage warning indicators for the overall deformation of concrete dam considering fuzziness and randomness were determined. Therefore, nondeterministic optimal control was achieved [12]. This paper concludes with a project case that verified the feasibility of the proposed theory.

2. Nondeterministic Optimal Control

Hydraulic engineering considers that some fuzziness and randomness in dams are inevitable because of the influences of various random factors such as the nondeterminacy of mechanical parameters, imposed load, and boundary conditions. Statistically, the smaller probability of dam displacement indicates that the dam is in a more dangerous state. $\mu + 3\sigma$ and $\mu - 3\sigma$ can be used as a warning indicator in one confidence coefficient of the dam if the displacement obeys the normal distribution of mean value and variance and is in the range of $\mu \pm 3\sigma$ in deterministic optimal control. In reality, given that dams are affected by random factors, warning indicators will have a range of variations. For example, in Figure 1, the warning value of upstream displacement has a maximum control value and minimum control value.

As the foundation of cloud theory, the cloud model is precisely both controlled and uncontrolled in microscope scale. *U* is the time series of monitoring the dam deformation, and *C* is the qualitative judgment on dam safety. If the quantitative value $x \in U$, u(x) fall in [0, 1] and follow the probability distribution law:

$$\mu: U \longrightarrow [0,1], \quad \forall x \in U, \ x \longrightarrow \mu(x).$$
 (1)

The distribution of x in U is called cloud and (x, u) is the cloud droplet.

In Figure 2, for observation point *i*, if the range wherein the cloud droplet falls is given, the upper bound and lower

bound of the cloud droplet in the cloud model y_i^u and y_i^l can be expressed as follows:

$$y_{i}^{u}(x_{i}, Ex_{i}, En_{i}, He_{i}) = e^{-(x_{i} - Ex_{i})^{2}/2(En_{i} + 3He_{i})^{2}},$$

$$y_{i}^{l}(x_{i}, Ex_{i}, En_{i}, He_{i}) = e^{-(x_{i} - Ex_{i})^{2}/2(En_{i} - 3He_{i})^{2}}.$$
(2)

 x_{ij} is the value *j* of observation point *i*, and the fuzziness Δ_{ij} can be calculated by

$$\Delta_{ij} = y^{\mu}_{ij} - y^{l}_{ij}.$$
 (3)

In this equation, y_{ij}^{u} is the upper limit value of x_{ij} in the range and y_{ij}^{l} is the lower limit value of x_{ij} in the range.

Given the influence of random factors, when $x_{ij} \ge 0$, x_{ij} changes in the range of $[x_{ij} - x_{ij}\Delta_{ij}, x_{ij} + x_{ij}\Delta_{ij}]$; when $x_{ij} < 0$, it is in the range of $[x_{ij} + x_{ij}\Delta_{ij}, x_{ij} - x_{ij}\Delta_{ij}]$. When drawing up the warning indicator for downstream deformation, the maximum and minimum control values of the indicator can be determined when the significance level is α ; thus, indicating that the nondeterministic optimal control has been achieved.

3. Methods of Characterizing Contributions of Single Observation Point

The overall deformation condition of the concrete dam is usually exposed to the influences of water pressure and temperature and is related to many factors including the physical and mechanical properties of dam materials, body structure, geology, and hydrology and it could be referred to in Figure 3. The principles of synergetics posit that a concrete dam is a synthesis of feature points with different contributions (weights) that influence one another. The contribution of a single observation point needs to be studied to construct a reasonable expression of overall deformation.

3.1. Construction of the Indicator Set of a Single Observation Point Weight. Entropy [13–15], a basic concept in thermodynamics, refers to a state function in a system. The concept of information entropy is a measurement of the system's disorder and nondeterminacy [15]. The measured value j on observation point i is x_{ij} and its corresponding entropy is S_{ij} . According to entropy theory, when an observation point is in a more dangerous state, the system is in greater disorder and its entropy value is smaller. Thus, the following is obtained:

$$S_{ij} = -\left[\mu_{ij}\ln\mu_{ij} + \left(1 - \mu_{ij}\right)\ln\left(1 - \mu_{ij}\right)\right], \qquad (4)$$
$$\mu_{ij} = \begin{cases} \int_{-\infty}^{x_{ij}} f(\varsigma) \, d\varsigma, & x_{ij} \ge 0\\ \\ \int_{x_{ij}}^{+\infty} f(\varsigma) \, d\varsigma, & x_{ij} < 0. \end{cases}$$
(5)

Formula (4) defines the information entropy of the measurement value. No matter what distribution $\{x_{ij}\}$ obeys, if the probability density function of the measured value is



FIGURE 1: Diagram of deterministic optimal control and nondeterministic optimal control.

Degree of certainty



FIGURE 2: Diagram of the range where the cloud droplet falls.

known, the corresponding information entropy sequence can be calculated. Deformation of dam can be divided into three parts: water pressure component, temperature component, and aging component. Aging component comprehensively reflects the creep and plastic deformation of dam concrete and rock foundation and compression deformation of geological structure of rock foundation. At the same time, it also includes the irreversible displacement caused by the dam crack and the autogenous volume deformation. It changes dramatically in the early stage and gradually tends to be stable in the later stage. The project selected in this paper is a dam which has worked for many years, and its aging components tend to be stable, and the deformation value is stable in annual period rule, which obeys normal distribution.

The indicator measuring how much information is contained in x_{ij} is the inverse entropy D_{ij} : $D_{ij} = 1 - S_{ij}$. When one measured value reflects more information, the entropy value S_{ij} will be smaller and its inverse entropy D_{ij} will be greater. Thus, D_{ij} can be used to measure how much information is reflected by a single measured value. Suppose the weight distribution of all observation points is $\{\omega_i \mid i = 1, 2, ..., n\}$, where *n* is the number of points observed and ω_i will meet the following requirement: $\omega_i \ge 0$ and $\sum \omega_i = 1$. The entropy S_{ij} of the object matrix of the deformation measured value $\{x_{ij} \mid i = 1 \sim n, j = 1 \sim m\}$ can be computed through (4) and (5). The inverse entropy matrix is expressed as follows:

$$D_{ij} = \begin{bmatrix} D_{11} & D_{12} & \cdots & D_{1m} \\ D_{21} & D_{22} & \cdots & D_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ D_{n1} & D_{n2} & \cdots & D_{nm} \end{bmatrix}.$$
 (6)

In this matrix, D_{ij} is the inverse entropy of x_{ij} . The weight of the feature points was traced by the projection pursuit method.

3.2. Process of Calculating the Weight of a Single Observation Point. By using the projection pursuit method [16, 17], the high-dimensional data can be projected to low dimension space, and projection that reflects the structure or features of high-dimensional data is pursued to analyze high-dimensional data. This method is advantageous because it is highly objective, robust, resistant to interference, and accurate. The steps are as follows.

Step 1. The extreme value of the inverse entropy matrix was normalized through the following equation:

$$D_{ij}^{*} = \frac{D_{ij} - [D_{j}]_{\min}}{[D_{j}]_{\max} - [D_{j}]_{\min}},$$
(7)

where $[D_j]_{\text{max}}$ and $[D_j]_{\text{min}}$ are the maximum and minimum values of line *j* in the matrix, respectively.



FIGURE 3: Diagram of overall deformation system of concrete dam.

Step 2. The normalized value D_{ij}^* was projected to unit direction *P*: $P = (p_1, p_2, ..., p_j)$ and $p_1^2 + p_2^2 + \cdots + p_j^2 = 1$. The indicator function of the projection *G*(*i*) was constructed:

$$G(i) = \sum_{j=1}^{m} p_j D_{ij}^*, \quad (i = 1, 2, ..., n).$$
(8)

Step 3. The objective function of the projection was constructed. The best direction for the projection was estimated by solving the maximization problem of the objective function in the constraint condition:

Objective function: Max :
$$H(p) = S_G \cdot Q_G$$
,
Constraint condition: $\sum_{j=1}^{m} p_j^2 = 1$. (9)

In this equation, S_G is the divergent degree of the projection. Q_G is the local density of 1D data points along P and is expressed as follows:

$$S_{G} = \left[\frac{\sum_{i=1}^{n} \left(G\left(i\right) - \overline{g}\left(i\right)\right)^{2}}{n-1}\right]^{0.5}$$

$$Q_{G} = \sum_{i=1}^{n} \sum_{j=1}^{n} \left(R - r_{ij}\right) \cdot f\left(R - r_{ij}\right),$$
(10)

where $\overline{g}(i)$ is the mean value of this sequence, *R* is the window radius of the local density and $R = 0.1S_G$ in this paper, r_{ij} is the distance between two projection values, and f(t) is the unit step function. f(t) is equal to 1 as *t* is greater than 0. Otherwise, f(t) is equal to 0.

Step 4. The projection value of one sample point was computed by substituting the best direction P^* into (8); ω_i can be calculated by normalizing the projection value:

$$\omega_i = \frac{G^*(i)}{\sum_{j=1}^n G^*(j)}, \quad i = 1, 2, \dots, n.$$
(11)

4. Study on Equivalent Model of Dam's Overall Deformation

The overall deformation of a dam can be considered a deformation system of feature points with different contributions that influence one another, as well as observation points of the deformation as feature points. The deformation condition was analyzed systemically, and the overall deformation was expressed by the evolution of equation of all feature points. The deformation condition was described qualitatively by the tectonic type of information entropy. The absolute value of the information entropy measures the danger level of the deformation value. Smaller absolute value means higher danger level. The positive and negative values indicate the direction of the deformation. A positive value corresponds to downstream deformation, whereas a negative value corresponds to upstream deformation. The influences of random factors were considered and the fuzzy information entropy was constructed.

4.1. Constructing Fuzzy Information Entropy of Single Measured Value. The downstream deformation is positive, whereas the upstream deformation is negative.

When the observation point moves downstream, make $\mu_{ij} = \int_{-\infty}^{x_{ij}} f(\varsigma) d\varsigma$ and according to the definition of information entropy, the information entropy of x_{ij} is expressed as (4).

Considering the influence of Δ_{ij} , μ_{ij} will float in $[\mu_{ij}^0, \mu_{ij}^1]$; the following is then obtained:

$$\mu_{ij}^{0} = \int_{-\infty}^{x_{ij}-x_{ij}\Delta_{ij}} f(\varsigma) d\varsigma,$$

$$\mu_{ij}^{1} = \int_{-\infty}^{x_{ij}+x_{ij}\Delta_{ij}} f(\varsigma) d\varsigma.$$
(12)

When the observation point moves upstream, make $\mu_{ij} = \int_{x_{ij}}^{+\infty} f(\varsigma) d\varsigma$; the information entropy of x_{ij} is defined as follows:

$$S_{ij} = \mu_{ij} \ln \mu_{ij} + (1 - \mu_{ij}) \ln (1 - \mu_{ij}).$$
(13)



FIGURE 4: Diagram of S_{ij} changes with the change of μ_{ij} if $E_x \ge 0$.

Considering the influence of Δ_{ij} , μ_{ij} floats in $[\mu_{ij}^0, \mu_{ij}^1]$. The following is then obtained:

$$\mu_{ij}^{0} = \int_{x_{ij}-x_{ij}\Delta_{ij}}^{+\infty} f(\varsigma) \, d\varsigma$$

$$\mu_{ij}^{1} = \int_{x_{ij}+x_{ij}\Delta_{ij}}^{+\infty} f(\varsigma) \, d\varsigma.$$
(14)

Considering the influences of random factors, the information entropy of $x_{ij} - S_{ij}$ floats in $[S_{ij}^0, S_{ij}^1]$, which was defined as the fuzzy information entropy of x_{ij} .

Take the downstream as an example. Influenced by determined and random factors, the fuzzy entropy of S_{ij} changes into the range of S_{ij} in the range of $[\mu_{ij}^0, \mu_{ij}^1]$.

4.2. Methods for Determining the Range of the Information Entropy of Single Measured Value. For Figure 4, the expectation of one deformation monitoring sequence sample at one observation point ($E_x \ge 0$, S_{ij}) changes with the change of μ_{ij} , as shown in Figure 4, where $\mu_0 = \int_{-\infty}^0 f(\varsigma) d\varsigma$.

When the dam deforms downstream, the change law of S_{ij} is as follows: S_{ij} will increase with increasing μ_{ij} when μ_{ij} is in the range of $(\mu_0, 0.5)$; S_{ij} will decrease with decreasing μ_{ij} when μ_{ij} is in the range of (0.5, 1); when $\mu_{ij} = 0.5$, S_{ij} will reach the maximum value. If 0.5 is in the range of $[\mu_{ij}^0, \mu_{ij}^1]$, the maximum of S_{ij} is S_{ij}^1 when $\mu_{ij} = 0.5$ and its minimum value is S_{ij}^0 at the endpoint. If 0.5 is not in the range of $[\mu_{ij}^0, \mu_{ij}^1]$, S_{ij} will have its maximum value S_{ij}^1 and minimum value S_{ij}^0 at endpoints. When the dam deforms upstream, S_{ij} will rise with the rise of μ_{ij} and it will have its maximum value S_{ij}^1 and minimum value S_{ij}^0 at endpoints.

The expectation of one deformation monitoring sequence sample at one observation point ($E_x < 0, S_{ij}$) changes with



FIGURE 5: Diagram of S_{ij} changes with the change of μ_{ij} if $E_x < 0$.

the change of μ_{ij} as shown in Figure 5, where $\mu_0 = \int_{-\infty}^0 f(\varsigma) d\varsigma$ and Figure 5 is presented for S_{ij} changes with the change of μ_{ij} .

When the dam deforms downstream, the change law of S_{ij} is as follows: S_{ij} will increase with decreasing μ_{ij} and will have its maximum value S_{ij}^1 and minimum value S_{ij}^0 at endpoints. When the dam deforms upstream, the figure shows that when μ_{ij} is in the range of $(1-\mu_0, 0.5)$, S_{ij} will increase with the drop of μ_{ij} ; when μ_{ij} is in the range of (0.5, 1), S_{ij} will decrease with the increase of μ_{ij} ; when $\mu_{ij} = 0.5$, S_{ij} will reach the minimum value. If 0.5 is contained in the range of $[\mu_{ij}^0, \mu_{ij}^1]$, S_{ij} will have its minimum value S_{ij}^0 at $\mu_{ij} = 0.5$ and have its maximum value S_{ij}^1 at endpoint; if 0.5 is not contained in the range, S_{ij} will have its maximum value S_{ij}^0 at endpoints.

4.3. Construction of the Fuzzy Information Entropy of Overall Deformation. On the basis of the above results, the expression of information entropy of overall deformation can be deduced. The contribution of the order degree of observation point *i* is $\omega_i \mu_{ij}$; make $\mu_{ij}^1 = \mu_{ij}$ and $\mu_{ij}^2 = 1 - \mu_{ij}$. According to the broad definition of information entropy, when the dam move deforms downstream, the expression of information entropy of overall deformation is expressed as follows:

$$S_{j} = -\sum_{i=1}^{m} \sum_{k=1}^{2} \omega_{i} \mu_{ij}^{k} \ln \left(\omega_{i} \mu_{ij}^{k} \right)$$
$$= -\sum_{i=1}^{n} \sum_{k=1}^{2} \omega_{i} \mu_{ij}^{k} \left(\ln \omega_{i} + \ln \mu_{ij}^{k} \right)$$
$$= -\sum_{i=1}^{n} \sum_{k=1}^{2} \omega_{i} \mu_{ij}^{k} \ln \omega_{i} - \sum_{i=1}^{n} \sum_{k=1}^{2} \omega_{i} \mu_{ij}^{k} \ln \mu_{ij}^{k}$$



FIGURE 6: Computational process of fuzzy information entropy of overall deformation.

(15)

$$= -\sum_{i=1}^{n} \omega_{i} \ln \omega_{i} \sum_{k=1}^{2} \mu_{ij}^{k} - \sum_{i=1}^{n} \omega_{i} \sum_{k=1}^{2} \mu_{ij}^{k} \ln \mu_{ij}^{k}$$
$$= -\sum_{i=1}^{n} \omega_{i} \ln \omega_{i} + \sum_{i=1}^{n} \omega_{i} S_{ij}.$$

When the dam move deforms upstream, the expression of the information entropy of overall deformation is

$$S_{j} = \sum_{i=1}^{m} \sum_{k=1}^{2} \omega_{i} \mu_{ij}^{k} \ln \left(\omega_{i} \mu_{ij}^{k} \right) = \sum_{i=1}^{n} \omega_{i} \ln \omega_{i} + \sum_{i=1}^{n} \omega_{i} S_{ij}.$$
 (16)

Therefore, the expression of information entropy of overall deformation is defined as follows:

$$S_{j} = \begin{cases} -\sum_{i=1}^{n} \omega_{i} \ln \omega_{i} + \sum_{i=1}^{n} \omega_{i} S_{ij}, & S_{ij} \ge 0\\ \sum_{i=1}^{n} \omega_{i} \ln \omega_{i} + \sum_{i=1}^{n} \omega_{i} S_{ij}, & S_{ij} < 0. \end{cases}$$
(17)

The absolute value of the information entropy of the overall deformation measures the danger level of the deformation; that is, a smaller absolute value means a higher danger level; positive and negative values stand for the direction of the deformation: a positive value means downstream deformation, whereas a negative value means upstream deformation.

Considering the influences of random factors, the fuzzy information entropy of x_{ij} is $[S_{ij}^0, S_{ij}^1]$; the fuzzy information entropy of overall deformation can be illustrated through (17). Computational process of fuzzy information entropy of overall deformation is shown in Figure 6.

5. Proposed Multistage Fuzzy Information Entropy of Overall Deformation Warning Indicators

Horizontal displacement of dam crest changes in an annual cycle: "upstream and downstream switch." Therefore, this displacement should be in a certain scale and be controlled under some monitoring indicators for the safe dam operation.

In the case of downstream displacement, the primary fuzzy warning indicator δ'_1 is defined as $\delta'_1 = (\delta^0_1, \delta^1_1)$. δ^0_1 is the lower limit of this indicator, and δ^1_1 is the upper limit; the secondary indicator δ'_2 is $\delta'_2 = (\delta^0_2, \delta^1_2)$, where δ^0_2 is the lower limit of this indicator and δ^1_2 is the upper limit. When $\delta^1_1 > \delta^0_2$, a cross phenomenon appears in the primary indicator and secondary indicator, when both of them should be categorized according to the membership of displacement measured. δ^* was introduced because the membership of displacement at this point is the same. Figure 7 shows diagram of multistage fuzzy information entropy warning indicators.

The primary fuzzy warning indicator is $\delta'_1 = (\delta^0_1, \delta^*)$, and the secondary is $\delta'_2 = (\delta^*, \delta^1_2)$.

If the deformation value is in (δ_1^0, δ_1^1) or (δ_1^0, δ^*) , the dam is in the state of primary warning; if the value is in (δ_2^0, δ_2^1) or (δ_2^0, δ^*) , the dam is in the state of secondary warning.

The time sequence of deformation at each observation point was analyzed by using the above theoretical method. The lower and upper limits of the fuzzy information entropy of overall deformation affected by Δ_{ij} will be $\{S_j^0\}$ and $\{S_j^1\}$. Considering the dam's long-term service, when the dam moves downstream, the lower limit $\{S_{mi}^0\}$ and upper limit



FIGURE 7: Diagram of multistage fuzzy information entropy warning indicators.

 $\{S_{mj}^{1}\}\$ were selected and when the dam moves upstream, the lower limit $\{R_{mj}^{0}\}\$ and upper limit $\{R_{mj}^{1}\}\$ were selected. $\{S_{mj}^{0}\}, \{S_{mj}^{1}\}, \{R_{mj}^{0}\}\$, and $\{R_{mj}^{1}\}\$ are random variables, and four subsample spaces with the sample size of *N* can be obtained by the following:

$$S^{0} = \left\{ S^{0}_{m1}, S^{0}_{m2}, \dots, S^{0}_{mm} \right\}$$

$$S^{1} = \left\{ S^{1}_{m1}, S^{1}_{m2}, \dots, S^{1}_{mm} \right\}$$

$$R^{0} = \left\{ R^{0}_{m1}, R^{0}_{m2}, \dots, R^{0}_{mm} \right\}$$

$$R^{1} = \left\{ R^{1}_{m1}, R^{1}_{m2}, \dots, R^{1}_{mm} \right\}.$$
(18)

Shapiro-Wilk test and Kolmogorov-Smirnov test can both test whether the samples obey normal distribution or not. But the Kolmogorov-Smirnov test is applicable to fewer samples. It can not only test if the samples are subject to normal distribution, but also test if samples are subject to other distributions. The basic idea of the K-S test is to compare the cumulative frequency of the observed value ($F_n(x)$) with the assumed theoretical probability distribution ($F_x(x)$) to construct statistics.

According to the method of empirical distribution function, segmented cumulative frequency is obtained by using the following formula:

$$F_{n}(x) = \begin{cases} 0, & x < x_{i} \\ \frac{i}{n}, & x_{i} \le x < x_{i+1} \\ 1, & x \ge x. \end{cases}$$
(19)

In the formula, $x_1, x_2, ..., x_n$ is sample data after arrangement. The sample size is *n*.

In the full range of random variable *X*, the maximum difference between $F_n(x)$ and $F_x(x)$ is

$$D_{n} = \max \left| F_{x}(x) - F_{n}(x) \right| < D_{n}^{\beta}.$$
 (20)

In the formula, D_n is a random variable whose distribution depends on *n*. D_n^β is critical value for a significant level β . It is considered that the distribution to be used at a significant level β cannot be resisted; otherwise, it should be rejected.

The distribution form was tested through the K-S method to determine the probability density function. Fuzzy warning indicator was then determined with different significant level. In dam safety evaluation, significant level α is the probability of the dam failure. Supposing S_m is the extreme of the information entropy of the upstream overall deformation, if $S > S_m$, the probability of the dam failure is $P(S > S_m) = \alpha = \int_{S_m}^{\infty} f(x) dx$ and the reliability index of dam failure is $1 - \alpha$. According to the dam importance, different failure probability is set and the multistage warning indicators were identified.

6. Example Analysis

6.1. Project Profile. One flat-slab deck dam built with reinforced concrete is an important part of one river basin cascade exploitation. The elevation of this dam crest is 137.70 m, and the height of biggest part is about 43 m; the crest runs 225.0 m in length and is made of 27 flat-slab buttresses with a span of 7.5 m. The space between the left side of 2# buttress and the right side of 9# buttress is the joint part; the overflow buttress is located from the 9# buttress to the 20# buttress; the rest is the water-retaining buttress. The workshop buttress is located from the 5# buttress to the 8# buttress. In this dam, the level of dead water is 122.0 m, the normal high water level is 131.0 m (in practice, it is 129.0 m), the design flood level is 136.7 m, and the check flood level is 137.5 m. To monitor the displacement of this dam, a direct plumb line and an inverted plumb line were arranged in four buttresses: 4#, 9#, 21#, and 24#. There is a crushed zone under the dam foundation where occurrence is N20°~25°W, SW \angle 70°~80°, the maximum width is about 3 m, and the narrowest place is about 1 m. There is an elevation clip joint mud at level 91 m.

The deformation field characterized by the observation point at 21# should be typical and is a key point because it is in the riverbed. Thus, the observation point G21 at the height of 134 m along the direct plumb line at 21# was analyzed, as well as point 27 at the height of 118 m along this line and point 28 at the height of 107 m. All these values measured were transformed into absolute displacement. The arrangement of each point is shown in Figure 8. The daily monitoring data series is from January 1, 2003, to December 31, 2013.

In this paper, two-stage warning indicators were set according to the practical running of this project and danger: $\alpha = 5\%$ is the primary warning that is mainly used to discriminate and handle early dangerous case and the reliability index of dam failure is 95%, whereas $\alpha = 1\%$ is the secondary warning that is mainly used to determine grave danger and prevent urgent danger and the reliability index of dam failure is 99%.



FIGURE 8: The arrangement of observation point.



FIGURE 9: The process line of upstream water level.



FIGURE 10: The process line of temperature.

6.2. Calculating the Contribution of Deformation at the Observation Point to the Overall Deformation. Figures 9-10 show the upstream stage hydrograph and temperature stage hydrograph, respectively. The water stage remained unchanged, whereas the temperature changed in the annual cycle. Figure 11 shows the relationship between information entropy of overall deformation and temperature in 2007. A negative correlation exists between the overall deformation of 21# buttress and temperature; that is, an increase in temperature corresponds to the decrease in upstream or downstream displacement and a decrease in temperature means an increase in the upstream or downstream displacement. Figure 12 shows the correlation between the displacement at observation point G21 and the temperature. The overall deformation law is roughly identical with that at observation point G21.

Figure 12 reveals that the temperature obviously influenced overall deformation; that is, the temperature can change the contribution of single observation point to the overall deformation. Temperature change was divided into the stage of temperature rise and stage of temperature drop.



FIGURE 11: The relationship between information entropy of overall deformation and temperature in 2011.



FIGURE 12: The relationship between horizontal displacement and temperature of observation point G21 in 2011.

The weight of deformation at observation point in two stages was calculated. The results are shown in the Table 1.

6.3. Results of Information Entropy of Overall Deformation. Range of cloud fall of the displacement at G21, 27, and 28 is shown in Figure 13. The boundary values of most dangerous information entropy from 2003 to 2013 are shown in Tables 2–5. The absolute value means the degree of danger; downstream is set as negative value. In the significance level $\alpha = 0.05$, 7 kinds of common distribution (lognormal distribution, normal distribution, uniform distribution, triangular distribution) exponential distribution, γ distribution, and β distribution) were used to carry on hypothesis test for { S_{mj}^{0} }, ${S_{mj}^{1}}$, { R_{mj}^{0} }, and { R_{mj}^{1} } with K-S method. The maximum D_{n} of each sequence was obtained and compared with the critical value $D_{n}^{0.05}$ of the significant level 0.05 to determine the type

Observation point	Altitude	The period of temperature rise	The period of temperature drop
G21	134 m	0.357	0.392
27	118 m	0.331	0.316
28	107 m	0.312	0.292

TABLE 1: The weight table of observation point.

TABLE 2: The lower limit of the most dangerous information entropy of downstream overall deformation.

Year	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
Information entropy	0.4784	0.4837	0.4767	0.4784	0.4969	0.4887	0.4969	0.5102	0.5082	0.4799	0.4739

TABLE 3: The upper limit of the most dangerous information entropy of downstream overall deformation.

Year	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
Information entropy	0.4791	0.4845	0.4770	0.4786	0.4971	0.4899	0.4974	0.5186	0.5161	0.4817	0.4769

TABLE 4: The lower limit of the most dangerous information entropy of upstream overall deformation.

Year	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
Information entropy	-0.5433	-0.5491	-0.4773	-0.5083	-0.5067	-0.4784	-0.5080	-0.4839	-0.4836	-0.4825	-0.5020

TABLE 5: The lower limit of the most dangerous information entropy of upstream overall deformation.

Year	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
Information entropy	-0.5339	-0.5440	-0.4770	-0.5079	-0.5055	-0.4770	-0.5040	-0.4822	-0.4806	-0.4754	-0.4766



FIGURE 13: Range of cloud fall of the displacement at G21, 27, and 28.

of the best distribution. K-S test results are shown in Table 6. Multistage fuzzy warning values are presented in Table 7.

K-S test shows that $\{S_{mj}^0\}$, $\{S_{mj}^1\}$, $\{R_{mj}^0\}$, and $\{R_{mj}^1\}$ satisfy normal distribution.

The probability density function of the sequence is

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{\sigma}\right).$$
 (21)

Parameter values of (21) are presented in Table 7.

In downstream deformation, if $\alpha = 5\%$, the primary warning indicator is (0.4763, 0.4775); if $\alpha = 1\%$, the secondary warning indicator is (0.4757, 0.4763). δ^* is used as

the boundary value when two indicators overlap. In the case of upstream deformation, if $\alpha = 5\%$, the primary warning indicator is (-0.4779, -0.4768); if $\alpha = 1\%$, the secondary warning indicator is (-0.4768, -0.4763) (Table 8).

6.4. The Structure Calculation of Monitoring Index of Dam Horizontal Displacement. According to the actual situation, three-dimensional finite element model of the dam is established. According to the structure and basic geological conditions of 21# dam section, the scope of finite element calculation model can be got as follows: taking 1.5 times as high dam in the upstream direction, taking 1.5 times as high dam in the upstream direction, and taking 1 time as high dam below the dam foundation. The model is constituted of 11445 nodes and 8571 units. The unit type is 6 sides 8 nodes isoparametric element. The deformation observation data analysis shows the dam under the condition of low temperature and high water level: there is larger displacement in the downstream when in high temperature, and in low water level, there is larger displacement in the upstream (Table 9, Figure 14). In view of the actual situation of the dam observation and data analysis results, the load condition selection is as follows (Table 10, Figures 15 and 16):

Working condition 1: normal water level 129.0 m and maximum temperature drop.

Working condition 2: dead water level 122.0 m and maximum temperature rise.

Probability distribution	S^0_{mj}	S_{mj}^1	R_{mj}^0	R_{mj}^1
Lognormal distribution	0.26	0.11	0.23	0.28
Normal distribution	0.11	0.08	0.26	0.22
Uniform distribution	0.74	1.55	0.88	0.68
Triangular distribution	0.53	0.54	0.59	0.61
Exponential distribution	0.41	0.42	0.55	0.35
γ distribution	0.37	0.36	0.31	0.33
β distribution	0.68	0.73	0.86	0.63
$D_n^{0.05}$	0.34	0.34	0.29	0.29
The most reasonable probability distribution	Normal distribution	Normal distribution	Normal distribution	Normal distribution

TABLE 6: K-S test results.

TABLE 7: Parameter values of the probability density function.

Data agrica	Parameter values					
Data series	μ	σ^2				
$\left\{S_{mj}^{0}\right\}$	0.488355	0.0129				
$\left\{S_{mj}^{1}\right\}$	0.490627	0.0151				
$\left\{R_{m_i}^0\right\}$	-0.50210	0.0249				
$\left\{R_{m_i}^1\right\}$	-0.49674	0.0245				



FIGURE 14: Finite element model of the dam.

The primary warning indicators of concrete dam were obtained by calculation methods for structures. If the information entropy of the overall deformation reached 0.4770, the dam moved downstream and in the state of primary warning. If the information entropy of the overall deformation reached -0.4773, the dam moved upstream and in the state of primary warning (Table 11). They all fell into intervals calculated by fuzzy methodology. The analysis shows that the method brought up in this paper is reasonable and scientific. Also, the analysis shows the physical meaning of the fuzzy warning index. Under the action of the unfavorable load combination and the influence of the complex random factors, the maximum entropy and minimum information entropy of the overall deformation lie in this interval. Considering the influences of random factors, multistage fuzzy warning values were receptive and safe.

In this paper, the overall deformation of 21# buttress was analyzed through the theoretical method proposed. Influences of random factors on the warning value were considered and multistage fuzzy warning values were determined. If the information entropy of overall deformation is in (0.4763, 0.4775), the dam moved downstream and in



FIGURE 15: The results of finite element calculation of working condition 1.



FIGURE 16: The results of finite element calculation of working condition 2.

the state of primary warning. If the information entropy of overall deformation is in (0.4757, 0.4763), the dam moved downstream and in the state of secondary warning. If the information entropy of overall deformation is in (-0.4779, -0.4768), the dam moved upstream and in the state of primary warning. If the information entropy of the overall deformation is in (-0.4768, -0.4768), the dam moved upstream and in the state of primary warning. If the information entropy of the overall deformation is in (-0.4768, -0.4763), the dam moved upstream and in the state of secondary warning.

7. Conclusion

This paper presented multistage warning indicators of concrete dam space and considered the influences of complex

The direction of deformation		Confidence le	evel
The direction of deformation	The primary warnir	The secondary warning indicator	
Downstream	(0.4763, 0.47	775)	(0.4757, 0.4763)
Upstream	(-0.4779, -0.4	4768)	(-0.4768, -0.4763)
	TABLE 9: Material par	ameter of the dam.	
Structure	Density (kg/m)	Poisson's ratio	Elastic modulus (GPa)
Concrete face slab	2400	0.167	24
Buttress	2400	0.167	24
Partition wall	2400	0.167	24
Reinforced concrete block	2400	0.160	22
Foundation rock mass	2700	0.175	12
Diorite-dyke	2000	0.3	1.15
Crushed zone	2000	0.3	0.29
Horizontal joints	2000	0.3	0.7

TABLE 8: Multistage fuzzy warning values of concrete dam.

TABLE 10: The results of finite element calculation of displacement of observation point (mm).

Observation point	Working condition 1	Working condition 2
G21	0.785	-0.599
27	0.654	-0.386
28	0.356	0.114

TABLE 11: The primary warning indicators of concrete dam.

The direction of deformation	The primary warning indicator
Downstream	0.4770
Upstream	-0.4773

random factors. The results of the specific studies are as follows:

- Influences of fuzziness and randomness of random factors on the long-term service of dam were discussed; a fuzziness indicator that measures the influence of random factors on monitoring value was constructed through cloud model.
- (2) Equivalent model of overall deformation was proposed. In the model, the overall deformation of dam was regarded as a deformation system where each observation point had different contributions (weights) and affected one another. Based on entropy theory, a space information entropy that can measure the overall deformation condition was established.
- (3) Multistage warning indicators against overall deformation of concrete dam under the influences of fuzziness and randomness were determined and nondeterministic optimal control of the indicator was achieved to improve the competence of warning against the deformation of concrete dam.

Competing Interests

No conflict of interests exits in the submission of this paper.

Acknowledgments

This paper was financially supported by National Natural Science Foundation of China (Grants nos. 41323001, 51139001, 51579086, 51379068, 51279052, 51579083, and 51209077), Jiangsu Natural Science Foundation (Grants nos. BK20140039 and BK2012036), Research Fund for the Doctoral Program of Higher Education of China (Grant no. 20130094110010), the Ministry of Water Resources Public Welfare Industry Research Special Fund Project (Grants nos. 201201038 and 201301061), Jiangsu Province "333 High-Level Personnel Training Project" (Grants nos. BRA2011179 and BRA2011145), Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions (Grant no. YS11001), Jiangsu Province "333 High-Level Personnel Training Project" (Grant no. 2017-B08037), Jiangsu Province "Six Talent Peaks" Project (Grant no. JY-008), and Fundamental Research Funds for the Central Universities (Grants nos. 2016B04114 and 2015B25414).

References

- F. Benedetto, G. Giunta, and L. Mastroeni, "A maximum entropy method to assess the predictability of financial and commodity prices," *Digital Signal Processing*, vol. 46, pp. 19–31, 2015.
- [2] S. Muraro, A. Madaschi, and A. Gajo, "On the reliability of 3D numerical analyses on passive piles used for slope stabilisation in frictional soils," *Geotechnique*, vol. 64, no. 6, pp. 486–492, 2014.
- [3] W. J. Yoon and K. S. Park, "A study on the market instability index and risk warning levels in early warning system for economic crisis," *Digital Signal Processing*, vol. 29, pp. 35–44, 2014.
- [4] D. Tonini, "Observed behavior of several leakier arch dams," *Journal of the Power Division*, vol. 82, no. 12, pp. 135–139, 1956.

- [5] R. Salgado and D. Kim, "Reliability analysis of load and resistance factor design of slopes," *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 140, no. 1, pp. 57–73, 2014.
- [6] Y. C. Gu and S. J. Wang, "Study on the early-warning threshold of structural instability unexpected accidents of reservoir dam," *Journal of Hydraulic Engineering*, vol. 40, no. 12, pp. 1467–1472, 2009.
- [7] H. Z. Su, F. Wang, and H. P. Liu, "Early-warning index for dam service behavior based on POT model," *Journal of Hydraulic Engineering*, vol. 43, no. 8, pp. 974–986, 2012.
- [8] L. Chen, Mechanics performance with parameters changing in space & monitoring model of roller compacted concrete dam [Ph.D. thesis], Hohai University, 2006.
- [9] A. B. Li, L. F. Zhang, Q. L. Wang, B. Li, Z. Li, and Y. Wang, "Information theory in nonlinear error growth dynamics and its application to predictability: taking the Lorenz system as an example," *Science China Earth Sciences*, vol. 56, no. 8, pp. 1413– 1421, 2013.
- [10] O. Victor, "The theory of clouds," *Library Journal*, vol. 132, no. 13, p. 62, 2007.
- [11] M. Yang, H. Z. Su, and X. Q. Yan, "Computation and analysis of high rocky slope safety in a water conservancy project," *Discrete Dynamics in Nature and Society*, vol. 2015, Article ID 197579, 11 pages, 2015.
- [12] C. Ricotta and G. C. Avena, "Evaluating the degree of fuzziness of thematic maps with a generalized entropy function: a methodological outlook," *International Journal of Remote Sensing*, vol. 23, no. 20, pp. 4519–4523, 2002.
- [13] V. I. Khvorostyanov and J. A. Curry, "Parameterization of homogeneous ice nucleation for cloud and climate models based on classical nucleation theory," *Atmospheric Chemistry and Physics*, vol. 12, no. 19, pp. 9275–9302, 2012.
- [14] H. Z. Su, Z. P. Wen, and Z. R. Wu, "Study on an intelligent inference engine in early-warning system of damhealth," *Water Resources Management*, vol. 25, no. 6, pp. 1545–1563, 2011.
- [15] M. Kamiński, "Probabilistic entropy in homogenization of the periodic fiber-reinforced composites with random elastic parameters," *International Journal for Numerical Methods in Engineering*, vol. 90, no. 8, pp. 939–954, 2012.
- [16] E. Czogała and J. Łęski, "Application of entropy and energy measures of fuzziness to processing of ECG signal," *Fuzzy Sets* and Systems, vol. 97, no. 1, pp. 9–18, 1998.
- [17] K. Wu and J. L. Jin, "Projection pursuit model for evaluation of region water resource security based on changeable weight and information," *Resources and Environment in the Yangtze Basin*, vol. 20, no. 9, pp. 1085–1091, 2011.

Research Article Semisupervised Community Detection by Voltage Drops

Min Ji,¹ Dawei Zhang,¹ Fuding Xie,² Ying Zhang,³ Yong Zhang,¹ and Jun Yang²

¹School of Computer Science and Technology, Liaoning Normal University, Dalian, Liaoning 116081, China
 ²School of Urban and Environmental Science, Liaoning Normal University, Dalian, Liaoning 116029, China
 ³College of Business Administration, Dalian University of Finance and Economics, Dalian, Liaoning 116622, China

Correspondence should be addressed to Fuding Xie; xiefd@lnnu.edu.cn

Received 3 November 2015; Revised 28 December 2015; Accepted 30 March 2016

Academic Editor: Pubudu N. Pathirana

Copyright © 2016 Min Ji et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Many applications show that semisupervised community detection is one of the important topics and has attracted considerable attention in the study of complex network. In this paper, based on notion of voltage drops and discrete potential theory, a simple and fast semisupervised community detection algorithm is proposed. The label propagation through discrete potential transmission is accomplished by using voltage drops. The complexity of the proposal is O(|V| + |E|) for the sparse network with |V| vertices and |E| edges. The obtained voltage value of a vertex can be reflected clearly in the relationship between the vertex and community. The experimental results on four real networks and three benchmarks indicate that the proposed algorithm is effective and flexible. Furthermore, this algorithm is easily applied to graph-based machine learning methods.

1. Introduction

From the point of view of mathematics, many real-world systems in nature and society can be effectively modeled as complex networks or graphs. Specifically, the entities of the system are represented by the vertices and the interactions between the entities are represented by the edges. Examples include social relationships, spreading of viruses and diseases, the World Wide Web, author cooperation networks, citation networks, and biochemical networks. It has been shown that many real-world networks have a structure of modules or communities, where the nodes within a community are higher connected to each other than the nodes among communities. The community structures play an important role in the functional properties of complex network, and finding such a structure could be of significant practical importance.

Identifying community structure in special networks has a considerable merit of practice because it gives us insights to the structure-functionality relationship. In the past decades, plenty of techniques have been proposed to detect the community structure hidden in networks. The more typical algorithms for community detection can be found in [1]. Very recently, Chen et al. [2] defined the antimodularity as a quantitative measure of anticommunity partitioning on a network and showed the reliability of antimodularity as a measurement of the quality of an anticommunity partitioning. A vertices similarity probability model to find community structure without the prior knowledge of the type of complex network structure was presented [3]. By studying the community structure in Chinese character network, Zhang et al. [4] found that community structure was always considered as one of the most significant features in complex networks, and it played an important role in the topology and function of the networks. Palla et al. [5] revealed that complex network models exhibited an overlapping community structure, also called fuzzy community. These complicated structures actually make it harder to appropriately construct algorithms to uncover them. Along this way, researchers have made great contributions to the community detection [6–10].

The methods mentioned above belong to unsupervised community detection methods since the topological information of the network is used only and its background knowledge is ignored. In fact, some prior information is of great value in identifying the community structure. Based on the discussion of an equivalence of the objective functions of the symmetric nonnegative matrix factorization and the maximum optimization of modularity density, Ma et al. [11] introduced a semisupervised clustering algorithm for community structure detection. In [12], Silva and Zhao presented a technique for semisupervised classification tasks, by using the modularity measure of complex networks, originally designed for unsupervised learning tasks. Zhang [13, 14] developed a method that implicitly encoded the pairwise constraints by modifying the adjacency matrix of the network, which could also be regarded as the denoising process of the consensus matrix of the community structures. A novel semisupervised community detection algorithm was proposed based on the discrete potential theory [15]. It effectively incorporated individual labels, the labels of corresponding communities, to guide the community detection process for achieving better accuracy. Although these existing semisupervised community detection methods can improve the community identification accuracy, some of them have limitations in high time complexity. Therefore, it is worthwhile to introduce the novel algorithm to identify community structures in complex network rapidly.

The application of discrete potential theory to detect community in network can be traced back to Wu and Huberman's work [16]. They presented a method that allowed for the discovery of communities within graphs of arbitrary size in times that scale linearly with their size. Their method was based on notions of voltage drops across networks that were both intuitive and easy to solve regardless of the complexity of the graph involved. Zhang et al. [17] applied it to directed networks; they presented a new mechanism for the local organization of directed networks and designed the corresponding link prediction algorithm. Wang and Zhang [18] came up with a semisupervised clustering method based on generalized point charge models for text data classification. Liu et al. [15] recently proposed a linear time algorithm to find the community in network based on discrete potential theory. As data sets get larger and larger, it is still necessary to develop the efficient semisupervised learning methods.

Motivated by Wu and Huberman's work [16] and Liu's method [15], in this paper, we present a simple and fast semisupervised algorithm for detecting the community in complex network by discrete potential theory and voltage drops. The complexity of the proposed algorithm is O(|V| +|E|) for the sparse network with |V| vertices and |E| edges. Similar to the membership degree in fuzzy c-means algorithm, the voltage value of each vertex in network implies the relationship between this vertex and community. The main contributions of the proposal are as follows: (1) the proposed algorithm is a simple and fast semisupervised approach to discover community structures in complex network. (2) The proposal gets rid of the limitation of positive definite matrix which is needed to solve a linear system by conjugate gradient decent algorithm. To some extent, this approach remedies the deficiency of Liu et al's work [15]. (3) The unsupervised Wu-Huberman algorithm is extended to semisupervised learning case. The experimental results demonstrate the effectiveness of the proposed algorithm.

2. The Graph and Discrete Potential Method

The graph can be mathematically represented as G = (V, E), where $V = \{v_1, v_2, \dots, v_n\}$ is the set of vertices and $E \subset V \times V$ denotes the set of edges. Generally, the graph can be expressed by its adjacent matrix A, whose elements A_{ij} are equal to 1 if v_i points to v_j and 0 otherwise. We denote d_i as the degree of vertex v_i . The degree matrix D is a diagonal matrix containing the vertex degree d_i (i = 1, 2, ..., n) of a graph on the diagonal. Then the Laplacian matrix L can be defined as

$$L = D - A. \tag{1}$$

Denote x_i^s as the potential of vertex v_i in the electrostatic field generated by vertices with label *s*. Assign the potentials of all labeled vertices with labels other than *s* to zero and the *s*labeled vertices to have a unit potential. The process of potential transmission for each electrostatic field is a circuit theory problem and can be modeled by combinatorial Dirichlet [15]. By using the Laplacian matrix *L*, a combinatorial formulation of the Dirichlet integral is in the form [15, 19]

$$D[x] = \frac{1}{2}x^{T}Lx,$$
(2)

where x is the potentials of all vertices minimizing (2). Reassigning the order of all vertices of the graph and putting the labeled vertices forward, (2) can be rewritten into

$$D[x] = \frac{1}{2} \begin{bmatrix} x_L^T, x_U^T \end{bmatrix} \begin{bmatrix} L_L & B \\ B^T & L_U \end{bmatrix} \begin{bmatrix} x_L \\ x_U \end{bmatrix}$$

$$= \frac{1}{2} \left(x_L^T L_L x_L + 2 x_U^T B^T x_L + x_U^T L_U x_U \right),$$
(3)

where x_L and x_U are two vectors whose elements represent the potentials of labeled vertices and unlabeled vertices, respectively. Setting the derivative of D[x] with respect to x_U equal to zero, one can obtain a system of linear equations

$$L_U x_U = -B^T x_L, (4)$$

where x_U is a $|V_U|$ dimensional vector whose elements are unknown quantities needing to be solved. If the graph is connected, or if every connected component contains a seed, then (4) will be nonsingular.

For each label *s*, a system of linear equations can be established as

$$L_U x^s = -B^T p^s. ag{5}$$

If one assigns a unit potential to the labeled vertices with label *s* and zero to other labeled vertices, it will generate an electrostatic field. The potentials of unlabeled vertices can be obtained by the solution of (5). By comparing the potentials of each unlabeled vertex, its label is assigned the same as the labeled vertex corresponding to the greatest potential. Thus the community structure is detected.

From the perspective of discrete potential theory, the solution to (5) can be interpreted as a circuit theory. Based on the three fundamental equations of circuit theory, Kirchhoff's Current Law, Ohm's Law, and Kirchhoff's Voltage Law, one can also get an equivalent system of (5) [15, 19].

In [15], the solutions of (5) have been obtained by conjugate gradient decent algorithm, and a novel semisupervised community detection algorithm was proposed. Several experimental results demonstrate the effectiveness of their approach.

3. The Proposed Algorithm

It should be noted that the coefficient matrix L_U in (5) must be a symmetric positive definite matrix while solving the nonhomogeneous linear equations (5) by conjugate gradient decent algorithm. Obviously, Laplacian matrix L is not a positive definite matrix since every row of the Laplacian matrix sums to zero, 0 is always its eigenvalue, and the corresponding eigenvector is (1, 1, ..., 1). This fact compels us to develop a new method to detect communities in network while considering the network as an electric circuit. In [16], Wu and Huberman introduced an unsupervised method to solve the system like (5) to discover the communities in complex network in linear time. Since there is no class information in advance, they employed bipartite strategy and some superb skills for the case of multiple communities. In this work, we extend their work to the case of semisupervised community detection.

In what follows, we would like to present a novel method to find community structure in complex networks by the process of voltage transmission.

For a given network, we suppose each edge to be a resistor with the same resistance. One attaches all the labeled vertices with label s to anode of a battery and other labeled vertices to negative pole so that they have fixed voltages, say 1 and 0. Based on these assumptions, the network can be viewed as an electric circuit with current flowing through each edge (resistor). By solving Kirchhoff equations, one can obtain the voltage value of each unlabeled vertex which of course should be within (0, 1). In this case, the voltage value of each vertex can be thought of as the membership degree similar as in FCM algorithm, which reflects clearly the relationship between a vertex and the sth community. In turn, we can get k voltages of a vertex for the different labels if there are k classes. In semisupervised learning methods, it is required that at least one sample must be labeled in each class. This indicates that the class parameter *k* is known previously.

Physically, if node v_i connects to *m* neighbors $v_1, v_2, ..., v_m$ in an electric circuit, the Kirchhoff equation [20] tells us that the total current flowing into *v* should sum up to zero; that is,

$$\sum_{j=1}^{m} I_j = \sum_{j=1}^{m} \frac{V_j - V_i}{R} = 0,$$
(6)

where I_j is the current flowing from v_j to v and V_j is the voltage at neighbor node v_j .

It is easy to rewrite (6) into the following form:

$$V_i = \frac{1}{m} \sum_{j=1}^m V_j.$$
 (7)

That is to say, the voltage of a node is the average of those voltages of its neighbors.

Suppose the number of communities to be k; then the label set $K = \{1, 2, ..., k\}$. In addition, we also assume that there must be at least one labeled vertex in each community. Divide the vertex set V into two parts, $V_L = \{(v_1, y_1), (v_2, y_2), ..., (v_l, y_l)\}$ (labeled vertices), where $y_i \in K$ is the label of vertex v_i and $V_U = \{v_{l+1}, ..., v_n\}$ (unlabeled vertices) such

that $V_L \cap V_U = \emptyset$. One also defines set $V_L^s = \{v_i \mid v_i \text{ with label } s\} = \{v_i \mid v_i \in V_L, y_i = s\}$. Denote V_i^s as the voltage of vertex v_i in the electrostatic field generated by vertices with label *s* and $N(v_i)$ as the set of neighbors of v_i .

If we reassign the order of all vertices of the graph and put the labeled vertices forward and labeled vertices with label *s* first, following (6), one can get the system:

$$V_i^s = 1, \text{ for } i = 1, 2, \dots, r,$$
 (8)

$$V_i^s = 0, \text{ for } i = r + 1, r + 2, \dots, l,$$
 (9)

$$V_{i}^{s} = \frac{1}{d_{i}} \sum_{\nu_{j} \in N(\nu_{i})} V_{j}^{s} = \frac{1}{d_{i}} \sum_{j} V_{j}^{s} A_{ij},$$
(10)

for
$$i = l + 1, l + 2, ..., n$$
.

Equation (10) is a linear system with n - l variables and can be put into a symmetrical form as follows:

$$V_i^s = \frac{1}{d_i} \sum_{j=l+1}^n V_j^s A_{ij} + \frac{1}{d_i} A_{i1} + \dots + \frac{1}{d_i} A_{ir},$$
(11)
for $i = l+1, l+2, \dots, n.$

Define

$$V^{s} = \begin{pmatrix} V_{l+1}^{s} \\ \vdots \\ V_{n}^{s} \end{pmatrix},$$

$$B = \begin{pmatrix} \frac{A_{l+1l+1}}{d_{l+1}} & \cdots & \frac{A_{l+1n}}{d_{l+1}} \\ \vdots & \vdots \\ \frac{A_{nl+1}}{d_{n}} & \cdots & \frac{A_{nn}}{d_{n}} \end{pmatrix},$$

$$C = \begin{pmatrix} \frac{A_{l+11} + \cdots + A_{l+1r}}{d_{l+1}} \\ \vdots \\ \frac{A_{n1} + \cdots + A_{nr}}{d_{n}} \end{pmatrix},$$
(12)

and then the matrix form of Kirchhoff equation is

$$V^s = BV + C, (13)$$

which has the solution

$$V^{s} = (I - B)^{-1} C.$$
(14)

Generally, it will take $O((n-l)^3)$ time to solve this system. Wu-Huberman algorithm [16] skillfully avoids this difficulty by solving (8)–(10) for r = 1 and l = 2. This method seems naturally to be a semisupervised learning method. We now extend it to the case of semisupervised learning.

Specifically, we first set $V_i^s = 1$, for $v_i \in V_L^s$, and $V_i^s = 0$, for $v_i \in V - V_L^s$.

Starting from $N(v_i)$ ($v_i \in V_L^s$), one consecutively updates the voltages of $v_i \in V_U$ to

$$V_i^s = \frac{1}{d_i} \sum_{v_j \in N(v_i)} V_j^s = \frac{1}{d_i} \sum_j V_j^s A_{ij}.$$
 (15)

The updating process adopts breadth-first search algorithm and it will end when we get voltages for all vertices in V_U . This process is called a round. One spends an amount of $O(d_i)$ time calculating neighbor voltage of vertex v_i and |V| time setting initial voltages; therefore the complexity in one round is O(|V|+|E|). After repeating the updating process for a finite number of rounds, one will reach an approximate solution of (14) within a certain precision which only depends upon the number of iteration rounds.

Unlike Wu and Huberman's method [20], we do not need to compute the ideal voltage gap and know roughly the size of each community. As a result, we get a (n - l)-dimensional voltage vector. The component V_i^s reflects the relationship of vertex v_i and sth community. For each label s in label set K = $\{1, 2, ..., k\}$, we repeat this process. Therefore, for each vertex v_i , we obtain a voltage vector $(V_i^1, V_i^2, ..., V_i^k)$. The element V_i^j can be considered as the membership degree which vertex v_i belongs to the *j*th community. The vertex v_i is within the *j*th community if $V_i^j = \max V_i^s$, $1 \le s \le k$. That is to say, largest voltage of each vertex indicates to which community the vertex v_i should belong.

4. Experiments

To validate the proposed algorithm, one would like to test it on four real networks and three benchmarks which are widely used to test the validity of various community division methods. The experimental platform is based on Windows 7 Ultimate Service Pack 1 with Intel[®] Core[™] i5-3470 CPU 3.20 GHz, 4.00 GB memory, ×64 Operating system, and Java 1.8 Eclipse RCP Luna sr1.

4.1. Three Evaluation Indices of Clustering. To assess the quality of partition, we here use the *F*-measure, *P*-measure, and modularity *Q* to quantify the cluster results. The *F*-measure is a harmonic combination of the precision and recall values used in information retrieval [21].

If n_i is the number of the members of class *i*, and n_{ij} is the number of the members of class *i* in cluster *j*, then the precision P_{ij} and recall R_{ij} can be defined as

$$P_{ij} = \frac{n_{ij}}{n_j},$$

$$R_{ij} = \frac{n_{ij}}{n_i}.$$
(16)

 F_{ii} is denoted by

$$F_{ij} = \frac{2 \times P_{ij} \times R_{ij}}{P_{ij} + R_{ij}}.$$
(17)

The corresponding *F*-measure (FM) of the whole clustering result is defined as

$$FM = \sum_{i} \frac{n_i}{N} \max_{j} F_{ij},$$
(18)

where N is the total number of the members in the data set. In general, the high value of F-measure indicates the

better cluster result. The purity of a cluster represents the fraction of the cluster corresponding to the largest class of data assigned to that cluster; thus the purity of cluster *j* is defined as

$$P_j = \frac{1}{n_j} \max_i n_{ij}.$$
 (19)

The purity of the whole clustering result is defined as

$$PM = \sum_{j} \frac{n_j}{N} P_j.$$
 (20)

In general, the larger the purity value is, the better the clustering result is.

In order to quantify the validity of community division of a complex network and to optimize the chosen splitting, we use, following [22], the concept of modularity. It is defined as follows: given a network division, Let e_{ij} be the fraction of edges in the network that connect vertices in group *i* to those in group *j*, and let $a_i = \sum_j e_{ij}$. Then the modularity *Q* is defined as

$$Q = \sum_{i} (e_{ii} - a_{i})^{2}.$$
 (21)

It measures the fraction of edges that fall between communities minus the expected value of the same quantity in a random graph with the same community division. Obviously, the larger *Q* corresponds to the ideal community structure.

4.2. Experiment on Four Real Networks. Testing an algorithm essentially means analyzing a network with a well-defined community structure and recovering its communities. In this subsection, four classical complex networks with known community structures are selected to test the introduced algorithm. The description of these four networks can be found everywhere [1, 11, 16, 23]. Taking Zachary Karate Club network with two communities, for example, we first choose randomly one node in each community and label it. Afterwards, the algorithm can work on this network and a community division is detected. The values of FM, PM, and Q can be computed according to the obtained partition. It is possible that the community division may be changed with the different selection of initial labeled notes. To evaluate the validity of the proposal objectively, we calculate the average values of three indices by choosing randomly 10 groups of initial labeled notes. Along this way, we also compute three indices values by adding the number of labeled notes in each community. In Table 1, we list the average values of three indices by selecting randomly 10 groups of different labeled
						Labol	numbor				
Network	Index					Laber	lumber				
iterioin	maex	1	2	3	4	5	6	7	8	9	10
	FM	0.8473	0.9571	0.9783	0.9902	0.9919	0.9952	0.9968	0.9903	0.9968	0.9936
Dolphins	PM	0.8484	0.9613	0.979	0.9903	0.9919	0.9952	0.9968	0.9903	0.9968	0.9936
	Q	0.2476	0.338	0.3634	0.3757	0.3743	0.3805	0.3773	0.3743	0.3788	0.3774
	FM	0.8503	0.9179	0.9561	0.9667	0.9845	0.9913	0.9914	0.9939	0.9957	1
Football	PM	0.867	0.9235	0.9574	0.9669	0.9843	0.9913	0.9913	0.9939	0.9957	1
	Q	0.5544	0.5738	0.5763	0.5744	0.5657	0.5628	0.5608	0.5612	0.5582	0.554
	FM	0.764	0.8112	0.9409	0.9369	0.9615	0.9764	0.9853	0.9853	0.9882	0.9882
Zachary	PM	0.7235	0.7941	0.9412	0.9382	0.9618	0.9765	0.9853	0.9853	0.9882	0.9882
	Q	0.1655	0.2162	0.352	0.3449	0.3574	0.3642	0.3686	0.3705	0.3691	0.3704
	FM	0.7003	0.772	0.8541	0.8588	0.8805	0.8825	0.898	0.8994	0.9021	0.9249
Polbooks	PM	0.7095	0.7762	0.8676	0.8629	0.88	0.8838	0.8934	0.8971	0.8943	0.921
	Q	0.2732	0.3703	0.4593	0.4704	0.4581	0.472	0.4688	0.4632	0.4772	0.4623

TABLE 1: The average values of indices for four real networks.

TABLE 2: The average run time of the proposed algorithm for four real networks.

Network		Label number											
	1	2	3	4	5	6	7	8	9	10			
Dolphins	0.7015	0.2953	0.239	0.161	0.1156	0.0812	0.0812	0.0672	0.0562	0.0515			
Football	1.4094	0.6734	0.4562	0.3048	0.228	0.1826	0.1343	0.1107	0.0843	0.064			
Zachary	0.1174	0.0704	0.0455	0.0312	0.0283	0.0171	0.0172	0.0109	0.0095	0.0093			
Polbooks	1.7469	0.8718	0.5392	0.3938	0.2783	0.2688	0.2202	0.189	0.1735	0.1408			

notes and the label number (number of labeled nodes in each community) varies from 1 to 10.

From Table 1, it is easy to see that we can detect an ideal community division for these four networks by the proposed algorithm when we label 3 nodes in each community. The accuracy of network partition is greater than or equal to 94% except polbooks network. Three indices values are ascending or varying slightly with the increasing of labeled nodes. These results also show that one can detect a good network partition by labeling a small quantity of nodes in each community. For football network, we can get the same partition accuracy as in [15] while the number of the labeled vertices randomly selected is from 1 to 4.

In Table 2, the average run times of the proposed algorithm for four real networks are presented. It is shown that the run times decrease with the increase of labeled nodes. This is reasonable because the number of nodes that need to be divided is reduced.

Figures 1 and 2 show the variety of run time of the proposed algorithm and the values of three indices for dolphins network and karate network, respectively.

4.3. Experiment on Three Benchmarks. For testing community detection algorithms on graphs with overlapping communities, several artificial networks or benchmarks are introduced. Among them, the most famous benchmark for community detection is a class of networks introduced by Girvan and Newman (GN) [24]. Each network has 128 nodes, divided into four communities with 32 nodes. The average degree of the network is 16 and the nodes have approximately the same degree, as in a random graph.

In what follows, we apply the proposed algorithm to detect the communities on this benchmark. For each fixed number of labeled nodes, one also selects randomly 10 groups of different initial labeled nodes to compute the average values of three indices. The benchmark can be thought of as the network with apparent community structure if mixing parameter μ < 0.5. From Table 3, one can see that four communities in this benchmark are detected accurately when mixing parameter $\mu < 0.25$ and the number of labeled nodes is equal to or greater than 4. If we take $\mu = 0.5$ and label 10 nodes in each community, 90% of nodes in this benchmark can be partitioned correctly. When $\mu > 0.5$, this benchmark is with overlapping community structures. Although the partition accuracy becomes higher and higher with the increasing of number of labeled nodes, we can not find ideal communities in this network. Particularly, our algorithm fails to divide it into four groups when $\mu = 1$ and number of labeled nodes in each groups is less than 10.

Assuming that both the degree and the community size distributions are power laws, Lancichinetti et al. [25] designed a more general benchmark for testing community detection algorithms on graphs. Some parameters used in this benchmark are explained as follows:

N: number of nodes, k: average degree, max k: maximum degree,



FIGURE 1: Dolphins network.





 μ : mixing parameter,

 t_1 : minus exponent for the degree sequence,

 t_2 : minus exponent for the community size distribution,

min c: minimum for the community sizes,

max c: maximum for the community sizes,

on: number of overlapping nodes,

om: number of memberships of the overlapping nodes,

C: [average clustering coefficient] not mandatory.

In this benchmark, *N*, *k*, max *k*, and μ have to be specified. For the others, the program can use default values: $t_1 = 2$; $t_2 = 1$; on = 0; om = 0; min *c* and max *c* will be chosen close to the degree sequence extremes.

If we set parameters N = 128, k = 16, max k = 16, min c = 32, and max c = 32, a kind of Girvan-Newman benchmark will be obtained.

To test the validity of our algorithm on large network, we apply the proposed algorithm to this benchmark with parameters $N = 10^5$, k = 20, max $k = 10^4$, and $t_2 = 1$. The mixing parameter μ is varied from 0.1 to 0.6. For each fixed μ , one takes $t_1 = 2$ and $t_1 = 3$, respectively. Unlike the GN benchmark, the community size is power laws in this network. Therefore, it is proper to label nodes in each community in terms of node proportion. The minimal proportion which we will take is 10% because of the requirement that there

	T., J.,		Label number											
μ	Index	1	2	3	4	5	6	7	8	9	10			
	FM	1	1	1	1	1	1	1	1	1	1			
0.0625	PM	1	1	1	1	1	1	1	1	1	1			
	Q	0.6875	0.6875	0.6875	0.6875	0.6875	0.6875	0.6875	0.6875	0.6875	0.6875			
	FM	1	1	1	1	1	1	1	1	1	1			
0.1250	PM	1	1	1	1	1	1	1	1	1	1			
	Q	0.625	0.625	0.625	0.625	0.625	0.625	0.625	0.625	0.625	0.625			
	FM	0.9733	1	1	1	1	1	1	1	1	1			
0.1875	PM	0.9735	1	1	1	1	1	1	1	1	1			
	Q	0.5262	0.5625	0.5625	0.5625	0.5625	0.5625	0.5625	0.5625	0.5625	0.5625			
	FM	0.9264	0.9734	0.9977	1	1	1	1	1	1	1			
0.2500	PM	0.9266	0.9734	0.9977	1	1	1	1	1	1	1			
	Q	0.4135	0.4686	0.4973	0.5	0.5	0.5	0.5	0.5	0.5	0.5			
	FM	0.8592	0.8994	0.941	0.9789	0.993	0.9984	1	0.9992	0.9992	0.9992			
0.3125	PM	0.8601	0.9	0.9414	0.9789	0.993	0.9984	1	0.9992	0.9992	0.9992			
	Q	0.3112	0.3417	0.3819	0.4163	0.4311	0.4359	0.4375	0.4369	0.4368	0.4368			
	FM	0.7277	0.8017	0.8659	0.8973	0.9514	0.9679	0.9796	0.9899	0.9946	0.9953			
0.3750	PM	0.7336	0.8031	0.8664	0.8976	0.9516	0.968	0.9797	0.9899	0.9945	0.9953			
	0	0.2146	0.2411	0.2735	0.2965	0.3373	0.3498	0.3598	0.3668	0.3704	0.372			
	FM	0.5788	0.6648	0.761	0.8031	0.8475	0.8907	0.913	0.9388	0.9501	0.9654			
0.4375	PM	0.5922	0.6695	0.7617	0.8039	0.8492	0.8914	0.9133	0.9391	0.95	0.9656			
	0	0.1495	0.1614	0.1882	0.2119	0.2271	0.2495	0.2648	0.2757	0.2822	0.2923			
	FM	0.4698	0.5567	0.6344	0.6806	0.7386	0.7615	0.8168	0.8546	0.8771	0.9054			
0.5000	PM	0.4781	0.5695	0.6383	0.682	0.7391	0.7625	0.818	0.8554	0.8773	0.9062			
	0	0.1173	0.1265	0.1298	0.1456	0.1614	0.1701	0.1822	0.1931	0.2021	0.2135			
	FM	0.41	0.4697	0.5389	0.5613	0.6142	0.6576	0.6784	0.7318	0.7464	0.7826			
0.5625	PM	0 4086	0.4766	0.5406	0.5641	0.618	0.6594	0.682	0.7344	0.7469	0.7828			
	0	0.1142	0 1021	0 1133	0.1128	0 1194	0.1278	0.002	0.1401	0.1377	0.1509			
	FM	0.3528	0 3844	0.434	0.4679	0 4989	0 5401	0.5776	0 5963	0.636	0.6725			
0.6250	PM	0.3469	0 3891	0.4352	0.4719	0.5031	0.5422	0.5805	0.5984	0.6375	0.6726			
	0	0.125	0.099	0.1047	0.1099	0.1123	0.1075	0.1105	0 1144	0.1186	0.118			
	FM	0.125	0.3519	0.1047	0.1077	0.4179	0.1075	0.4791	0.4971	0.5191	0.5641			
0.6875	PM	0.3352	0.3406	0.3789	0.4078	0.4179	0.402	0.4791	0.4971	0.5191	0.5656			
0.0075	0	0.3352	0.0400	0.3012	0.4078	0.4211	0.4050	0.4020	0.4904	0.5155	0.3030			
	EM	0.3547	0.1042	0.1070	0.100	0.3613	0.1151	0.3033	0.1207	0.1142	0.1172			
0 7500	DM	0.3347	0.3421	0.3406	0.3445	0.3633	0.3805	0.393	0.4233	0.4490	0.475			
0.7500	ГМ О	0.5211	0.5254	0.3400	0.3443	0.3055	0.3803	0.393	0.4227	0.4308	0.475			
	EM	0.1012	0.1055	0.1004	0.1107	0.1200	0.1100	0.1140	0.1123	0.1100	0.1150			
0.8125	FIVI DM	0.3297	0.3301	0.3107	0.323	0.3209	0.3413	0.3322	0.3010	0.3939	0.4100			
0.0125	PM	0.5205	0.3148	0.304/	0.5141	0.3104	0.330	0.3484	0.3009	0.3901	0.418			
		0.1219	0.1055	0.1109	0.1065	0.1045	0.109	0.121	0.1119	0.1082	0.1042			
0.9750	FM	0.3461	0.3324	0.3307	0.3255	0.3166	0.3255	0.31/6	0.337	0.341/	0.3657			
0.8750	PM	0.5156	0.51/2	0.318	0.3187	0.3109	0.31/2	0.3141	0.3352	0.3406	0.3649			
	Q	0.1028	0.1044	0.108	0.1081	0.1151	0.1188	0.1158	0.1145	0.1055	0.1041			
0.0275	FM	0.3314	0.3252	0.3442	0.3266	0.327	0.3187	0.3125	0.3107	0.3199	0.337			
0.93/3	PM	0.307	0.3133	0.3289	0.318	0.3195	0.3156	0.3102	0.3125	0.3211	0.3391			
	<u>Q</u>	0.099	0.1073	0.1027	0.097	0.1086	0.1052	0.1029	0.1037	0.0981	0.1048			
1 0000	FM	0.3286	0.3312	0.3455	0.3359	0.3318	0.3237	0.3136	0.3096	0.306	0.3257			
1.0000	PM	0.3109	0.3156	0.3321	0.3297	0.325	0.318	0.3102	0.3062	0.3078	0.3242			
	Q	0.1139	0.1018	0.0981	0.1067	0.102	0.1076	0.096	0.1002	0.1034	0.0967			

TABLE 3: The average values of indices on GN benchmark.

TABLE 4: The mean values of indices on power law benchmarks.

Notwork	Indox		Label	proportio	on (%)	
INCLWOIK	muex	10	20	30	40	50
	FM	0.9585	0.9627	0.9707	0.9739	0.9795
pw-0.1-2-1	PM	0.958	0.9622	0.9703	0.9736	0.9792
	Q	0.8152	0.8155	0.8204	0.8233	0.8256
	FM	0.9602	0.9658	0.9709	0.9755	0.98
pw-0.1-3-1	PM	0.9524	0.9594	0.9657	0.9718	0.9775
	Q	0.789	0.7987	0.807	0.8148	0.8231
	FM	0.8968	0.9247	0.9332	0.9514	0.9579
pw-0.2-2-1	PM	0.8956	0.9225	0.9318	0.9499	0.9568
	Q	0.6756	0.6861	0.6922	0.701	0.7016
	FM	0.8923	0.9125	0.9241	0.9362	0.9465
pw-0.2-3-1	PM	0.8711	0.8953	0.9097	0.9252	0.9383
	Q	0.6079	0.6323	0.6478	0.6635	0.6772
	FM	0.8102	0.8582	0.8841	0.9066	0.9264
pw-0.3-2-1	PM	0.8125	0.8537	0.8798	0.9025	0.9231
	Q	0.5312	0.5539	0.5602	0.5739	0.5828
	FM	0.7064	0.7521	0.7933	0.8245	0.8574
pw-0.3-3-1	PM	0.6776	0.7228	0.7651	0.7995	0.8369
	Q	0.3901	0.4159	0.4466	0.4707	0.4988
	FM	0.7013	0.7532	0.81	0.8386	0.8712
pw-0.4-2-1	PM	0.7083	0.7537	0.8059	0.8354	0.8675
	Q	0.4218	0.4383	0.4562	0.4719	0.4814
	FM	0.7405	0.8253	0.8679	0.8994	0.9226
pw-0.4-3-1	PM	0.7125	0.7957	0.8401	0.8747	0.9025
	Q	0.351	0.4014	0.4291	0.4512	0.4698
	FM	0.5425	0.6709	0.7346	0.7663	0.8152
pw-0.5-2-1	PM	0.5603	0.671	0.7335	0.764	0.8119
	Q	0.2966	0.3432	0.3619	0.3702	0.385
	FM	0.5021	0.6245	0.6955	0.7542	0.8059
pw-0.5-3-1	PM	0.4752	0.5846	0.6519	0.7112	0.7689
	Q	0.2017	0.2365	0.2601	0.2851	0.3123
	FM	0.4258	0.5589	0.6464	0.7004	0.7551
pw-0.6-2-1	PM	0.4394	0.5607	0.6423	0.6964	0.7496
	Q	0.1988	0.2389	0.2642	0.2802	0.2882
	FM	0.3891	0.5211	0.612	0.6895	0.7537
pw-0.6-3-1	РМ	0.3665	0.4811	0.5632	0.6383	0.7084
	Q	0.1494	0.1739	0.191	0.2112	0.2309

exists one labeled node at least in each community and the fact that there are small size communities in this network. Applying the proposed algorithm on this benchmark by labeling randomly of two groups of different initial nodes, one obtains some results reported in Table 4. There are nearly 90% of nodes which can be classified correctly in this network while $\mu < 0.2$ and 10% nodes in each community are labeled. In this case, there is no distinct variety of three indices values with the increasing of label proportion. This fact indicates that one can detect a good community division on the network with apparent community structure although a few nodes are labeled. The values in each column are

descending with the increasing of mixing parameter μ . This shows that a good network partition will not be found by the proposed algorithm for the network which communities overlap seriously.

Figure 3 presents the comparison of run time of our algorithm on two benchmarks with different parameters and label nodes numbers or label proportions. The increasing of labeled nodes number or label proportions implies that the number of unlabeled nodes in benchmarks is descending, and therefore it needs less and less time to partition network into groups.

We now present our experimental results on the LFR benchmark and further compare our proposal with GN algorithm [24], spectral clustering algorithm [1], NMF algorithm [20], and SNMF-SS algorithm [11] by a normalized mutual information index (NMI).

The LFR benchmark is designed by Lancichinetti et al. [25] and widely employed to test the performance of community structure identification. It allows user to specify distributions for both the community sizes and the degree distribution and then generates vertices and communities by sampling from those distributions. The mix parameter μ represents the average ratio of intracommunity adjacencies to total adjacencies. The large μ corresponds to the network with apparent community structure. In this paper, the input parameters of the LFR benchmark are the same for our algorithm and the comparative algorithms. For the different values of $\mu \in \{0.50, 0.6, 0.7, 0.8, 0.9\}$, we generated 50 instances for each of LFR benchmark graphs whose node degree is taken from a power law distribution with exponent 2 and community size from a power law distribution with exponent 1. Each graph has 1000 vertices, average degree of 15, maximum degree of 50, maximum for the community sizes of 50, and minimum for the community sizes of 5. The definition of NMI can be found everywhere [11, 15, 26].

From Figure 4, we can see that the values of NMI obtained by our algorithm are bigger than those gotten by the other four algorithms. The peak value of our approach is 0.732 at $\mu_{avg} = 0.9$. This value is bigger than the one 0.7 computed by SNMF-SS algorithm. Because the decrease of μ means that the LFR benchmark is with the obscure community structure, it is difficult to detect communities correctly for five algorithms. It is reasonable that the NMI values obtained by five algorithms become smaller and smaller as μ decreases. The NMF algorithm seems to be stable since it has a small decrease speed. The performance of our proposal decreases greatly while μ is greater than 0.6. This fact implies that our algorithm can not apply the networks with nonapparent community structure. However, compared with other four algorithms, our algorithm can gain the best performance.

5. Conclusions

In this paper, we propose a semisupervised community detection algorithm for partitioning network into groups. This approach amalgamates the discrete potential theory and Wu-Huberman algorithm. The complexity O(|V| + |E|) of the introduced approach indicates that it can be applied to detect



FIGURE 3: The comparison of run time of our algorithm on two benchmarks.



FIGURE 4: The comparative results of five algorithms on the LFR benchmark.

community on large network. The validity of our proposal is demonstrated by applying it to four real networks and three benchmarks. The experimental results show that a good community division of a complex network is obtained by labeling a small quantity of nodes in each community. However, it is difficult to classify correctly the network with heavily overlapping communities or obscure community structure by our method. This fact can be seen from the experimental result on LFR benchmark. Therefore, it is worthwhile to further introduce new and fast algorithm to deal with this case.

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

Acknowledgments

This work is supported by NSFC (under Grants nos. 61373127 and 41471140).

References

- S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3–5, pp. 75–174, 2010.
- [2] L. Chen, Q. Yu, and B. Chen, "Anti-modularity and anti-community detecting in complex networks," *Information Sciences*, vol. 275, pp. 293–313, 2014.
- [3] K. Li and Y. Pang, "A unified community detection algorithm in complex network," *Neurocomputing*, vol. 130, pp. 36–43, 2014.
- [4] L. Zhang, J. Cao, and J. Li, "Complex networks: Statistical properties, community structure, and evolution," *Mathematical Problems in Engineering*, vol. 2015, Article ID 590794, 7 pages, 2015.
- [5] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.
- [6] W. Li, "A constrained power method for community detection in complex networks," *Mathematical Problems in Engineering*, vol. 2014, Article ID 804381, 6 pages, 2014.
- [7] Z. Chen, Z. Xie, and Q. Zhang, "Community detection based on local topological information and its application in power grid," *Neurocomputing*, vol. 170, pp. 384–392, 2015.

Mathematical Problems in Engineering

- [8] L. Zhou, K. Lü, P. Yang, L. Wang, and B. Kong, "An approach for overlapping and hierarchical community detection in social networks based on coalition formation game theory," *Expert Systems with Applications*, vol. 42, no. 24, pp. 9634–9646, 2014.
- [9] K. Liu, J. Huang, H. Sun, M. Wan, Y. Qi, and H. Li, "Label propagation based evolutionary clustering for detecting overlapping and non-overlapping communities in dynamic networks," *Knowledge-Based Systems*, vol. 89, pp. 487–496, 2015.
- [10] L. Ma, M. Gong, J. Liu, Q. Cai, and L. Jiao, "Multi-level learning based memetic algorithm for community detection," *Applied Soft Computing*, vol. 19, pp. 121–133, 2014.
- [11] X. Ma, L. Gao, X. Yong, and L. Fu, "Semi-supervised clustering algorithm for community structure detection in complex networks," *Physica A: Statistical Mechanics and its Applications*, vol. 389, no. 1, pp. 187–197, 2010.
- [12] T. C. Silva and L. Zhao, "Semi-supervised learning guided by the modularity measure in complex networks," *Neurocomputing*, vol. 78, no. 1, pp. 30–37, 2012.
- [13] Z.-Y. Zhang, "Community structure detection in complex networks with partial background information," *Europhysics Letters*, vol. 101, no. 4, Article ID 48005, 2013.
- [14] Z.-Y. Zhang, K.-D. Sun, and S.-Q. Wang, "Enhanced community structure detection in complex networks with partial background information," *Scientific Reports*, vol. 3, no. 11, p. 48005, 2013.
- [15] D. Liu, X. Liu, W. Wang, and H. Bai, "Semi-supervised community detection based on discrete potential theory," *Physica A: Statistical Mechanics and its Applications*, vol. 416, pp. 173–182, 2014.
- [16] F. Wu and B. A. Huberman, "Finding communities in linear time: a physics approach," *European Physical Journal B*, vol. 38, no. 2, pp. 331–338, 2004.
- [17] Q.-M. Zhang, L. Lü, W.-Q. Wang, Y.-X. Zhu, and T. Zhou, "Potential theory for directed networks," *PLoS ONE*, vol. 8, no. 2, Article ID e55437, 2013.
- [18] F. Wang and C. Zhang, "Semisupervised learning based on generalized point charge models," *IEEE Transactions on Neural Networks*, vol. 19, no. 7, pp. 1307–1311, 2008.
- [19] L. Grady, "Random walks for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 11, pp. 1768–1783, 2006.
- [20] S. Zhang, R.-S. Wang, and X.-S. Zhang, "Identification of overlapping community structure in complex networks using fuzzy c-means clustering," *Physica A: Statistical Mechanics and its Applications*, vol. 374, no. 1, pp. 483–490, 2007.
- [21] C. Luo, Y. Li, and S. M. Chung, "Text document clustering based on neighbors," *Data & Knowledge Engineering*, vol. 68, no. 11, pp. 1271–1288, 2009.
- [22] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, Article ID 026113, 2004.
- [23] D. W. Zhang, F. D. Xie, Y. Zhang, F. Y. Dong, and K. Hirota, "Fuzzy analysis of community detection in complex networks," *Physica A: Statistical Mechanics and its Applications*, vol. 389, no. 22, pp. 5319–5327, 2010.
- [24] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, pp. 7821–7826, 2002.
- [25] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Physical Review E*, vol. 78, no. 4, Article ID 046110, 2008.

[26] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics*, vol. 406, Article ID P09008, 2005.

Research Article

Optimization of the Distribution and Localization of Wireless Sensor Networks Based on Differential Evolution Approach

Armando Céspedes-Mota,¹ Gerardo Castañón,¹ Alberto F. Martínez-Herrera,¹ and Leopoldo Eduardo Cárdenas-Barrón²

¹Department of Electrical and Computer Engineering, Tecnológico de Monterrey, Eugenio Garza Sada No. 2501 Sur, Monterrey, NL, Mexico

²School of Engineering and Sciences, Tecnológico de Monterrey, Eugenio Garza Sada No. 2501 Sur, Monterrey, NL, Mexico

Correspondence should be addressed to Leopoldo Eduardo Cárdenas-Barrón; lecarden@itesm.mx

Received 9 November 2015; Accepted 7 February 2016

Academic Editor: Hou-Sheng Su

Copyright © 2016 Armando Céspedes-Mota et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Location information for wireless sensor nodes is needed in most of the routing protocols for distributed sensor networks to determine the distance between two particular nodes in order to estimate the energy consumption. Differential evolution obtains a suboptimal solution based on three features included in the objective function: area, energy, and redundancy. The use of obstacles is considered to check how these barriers affect the behavior of the whole solution. The obstacles are considered like new restrictions aside of the typical restrictions of area boundaries and the overlap minimization. At each generation, the best element is tested to check whether the node distribution is able to create a minimum spanning tree and then to arrange the nodes using the smallest distance from the initial position to the suboptimal end position based on the Hungarian algorithm. This work presents results for different scenarios delimited by walls and testing whether it is possible to obtain a suboptimal solution with inner obstacles. Also, a case with an area delimited by a star shape is presented showing that the algorithm is able to fill the whole area, even if such area is delimited for the peaks of the star.

1. Introduction

It is well-known that location is a fundamental problem in wireless sensor networks (WSN) and it is essential based on either the distance or the connectivity among the nodes in the wireless sensor network. It is necessary to truthfully localize the sensors to determine the value of different parameters such as temperature and geographic coordinates of a given location, detect the occurrence of events, classify a detected object, and track an object. Sensors usually have low energy consumption, network self-organization, collaborative signal processing, and querying abilities. In consequence, it is crucial that the required data be disseminated and localized to the proper end users [1, 2].

A WSN is comprised of a large number of sensor nodes, which are densely positioned into a given area. Basically, they are a set of almost-static nodes and the nodes are not connected to each other. Their distribution inside a given area with minimum energy connectivity is an NP-complete problem [3–6].

For applications like automatic location, tracking, and guidance, the well-known global positioning system (GPS) is a space-based navigation system that provides location any time and is widely used. One way to acquire location information of sensor nodes is to equip each sensor with a GPS, but, in practice, in sensor networks with a large number of sensor nodes, only a limited portion of sensors are equipped with a GPS called *anchors* in the literature [7, 8]. An example of location-based protocol is presented by [9], it maintains a minimum energy communication network for wireless networks using low power GPS, and also it helps to calculate a minimum-power topology for mobile and stationary nodes by the use of a sink or concentrator node. Moreover, topology optimization finds the best network topology under certain constraints assuming that nodes must maintain communication to at least one neighbor [10].

Localization algorithms usually assume that sensors are distributed in regular areas without cavities or obstacles, but outdoor deployment of wireless sensor networks is very different. To solve this problem a reliable anchor-based localization technique is proposed by [7] in which the localization error due to the irregular deployment areas is reduced. Locating sensors in irregular areas is also analyzed by [8]; the problem is formulated as a constrained leastpenalty problem. The effects of anchors density, range error, and communication range on localization performances are analyzed. Nonconvex deployment areas such as C-shaped and S-shaped topologies are used. Moreover, in [11] the authors focused on the area coverage problem in a nonconvex region, as the region of interest limited by the existence of obstacles, administrative boundaries, or geographical conditions. The authors apply the discrete particle swarm optimization (PSO) algorithm with the use of a grid system to discretize the region of interest.

WSN deployment consists in determining the positions for sensor nodes in order to obtain the desire coverage, localization, connectivity, and energy efficiency. Available PSO solutions for the deployment problem are analyzed centrally on a base station to determine positions of the sensor nodes, the mobile nodes, or base stations as described in [12–14]. A node localization approach using PSO is described in [15–17].

On the other hand, flocking is the event or singularity in which a large number of agents (sensors) with limited environmental information and simple rules are organized into a coordinated motion and are widely used in many control areas including mobile sensor networks. Flocking of multiagents with virtual leader is presented in [18], in which the sensors in a network can be moved by using the improved algorithm. However, coordinated control of multiagent systems with leaders points to making all agents reach a coordinated motion with leaders. The case with multiple leaders is named containment control, in which every control input of agents is subject to saturation owing to its upper and lower limits in multiagent systems. This case is cleverly described in [19], focused on switching networks. The containment control with periodically intermittent communication and input saturation is deeply analyzed by [20] in which the authors design both state feedback and output feedback containment control protocols with intermittent input saturation via low gain feedback. This technique achieves a practical semiglobal containment of such agents on fixed networks.

There are some existing results in the literature that analyze the wireless sensors distribution in the threedimensional space such as in [21–23]. These scenarios analyze the distribution of the sensor networks under certain conditions that are not considered in this research work. The three-dimensional sensor distribution is out of the scope of this research paper but is considered as a future work. For instance, to extend this research paper to the threedimensional case, it is necessary to consider new objective functions, new scenarios with regular and irregular areas, and new upper and lower bounds; also, instead of computing the coverage area, it will be necessary to compute the involved volume and this requires more computational time to obtain results with low granularity.

In this research work, to optimize the distribution and location of sensor nodes, the Differential Evolution Algorithm (DEA) is applied. This technique has been successfully applied on different problems due to its simplicity of use [24-26]. Its taxonomy is in the category of bioinspired techniques. For each cycle denoted as generation, DEA uses a set of potential random solutions in a given search space, called individuals. In this research paper, an individual contains the information of the coordinates of the sensor nodes (x, y)and their communication radii values. For each individual in the population, a descendant is generated from three parents. One parent, called the main parent, is disturbed by the vector of differences of the other two parents. In the case of the descendant having a better performance according to the objective function, it substitutes the individual, else the original individual is retained and is passed on to the next generation of the algorithm and the descendant is eliminated. This process is repeated till it attains the stop condition. The readers are referred to [24] for a complete theoretical analysis of the DE algorithm.

Early work to solve the node distribution optimizing area and energy based on Differential Evolution Algorithm is presented in authors of [27]. But they only consider a typical square area without obstacles and the work lacks information about the energy consumption and the mobility of the sensor nodes; also, the localization of sensors is not analyzed. Extended cases for irregular areas are shown by [28], in which the interference rate is included and how much nodes are turned on at same time. In [29], the problem is divided in uniformity (node spreading) and connectivity (node clustering) regarding the area; the energy is analyzed by dividing the problem in operational consumed energy and data-exchange consumed energy.

Differential evolutions with heuristic algorithms for nonconvex optimization on sensor network localization are analyzed in [30]; here, the connections and disconnections constraints are considered by the authors. Also, the connectivitybased localization problem is analyzed as an optimization problem with both convex and nonconvex constraints. In [31], the minimization equation of WSN localization error problem and an approach of differential evolution for minimization of localization error in WSN are studied considering that sensor node localization is the ability of an individual node to determine the location information. There are also surveys related to localization in wireless sensor network described in [32, 33].

Compared to the previous listed works, here, the contributions are focused on building and establishing communication links even in the presence of obstacles or very restricted boundaries. DEA is used to optimize the distribution and as a consequence the location of the sensor nodes to minimize the overlap by using an extra function as shown by [28]. The main difference with [28] is that the overlap considered as the interference is also managed in terms of restrictions to minimize the overlap very quickly. It also applies for [29]. Besides, an extra parameter called random-M is used in the mutation operator to avoid the stagnation of DEA as suggested in [25].

In this research work, the results are presented for two area boundaries: inner star and a typical square room. But, instead of having an empty square room, the room includes subdivisions called walls, which will act as obstacles regarding the signal propagation, showing the optimization of the fitness function formed by the area coverage rate, the energy consumption, and an additional parameter that is related to the overlap among the nodes: the redundancy. For the used configurations, the initial node positions are randomly assigned. For the inner star, a minimum spanning tree (MST) is used at the end of each generation to check the connectivity of the nodes. For the scenarios with obstacles, such verification is performed for each individual of the population. At the end of each generation, the Hungarian algorithm [34] is used to find the shortest distance from their initial to the final location. In all scenarios here presented, the algorithm determines the initial and the final location of the sensor nodes. Each sensor node is located with respect to any of each other following the branches of the minimum spanning tree. The distance between nodes is also calculated by adding the respective distances of the branches among them.

The rest of this paper is organized as follows. Section 2 describes the methodology. Section 3 presents the DEA and minimum spanning tree algorithms for wireless sensor networks distribution and location. Section 4 summarizes the pseudocode applied on DEA. Section 5 presents the numerical results and a discussion. Finally, Section 6 presents some conclusions.

2. Description of the Method

2.1. Sensor Coverage Model. There is a particular interest in the sensor distribution and location where communication with neighboring nodes is more energy efficient. Also, a particular interest in the sensor distribution and location on irregular areas is considered. The main purpose of the sensor coverage model is to obtain a balance among the maximum effective coverage area, the minimum communication sensor energy, and reducing overlap.

The node set on the target area A_t is given by

$$\mathcal{N} = \left\{ N_1, N_2, \dots, N_z \right\},\tag{1}$$

and here *z* is the cardinality of \mathcal{N} .

The coverage range of a node N_i is defined as a circle centered at its coordinates (x_i, y_i) with sensing radius r_i . (x_i, y_i) coordinates define the position of sensor nodes and the subindex *i* is the sensor node index. A grid point (x, y) is covered by a sensor node *i* if and only if its distance to the center (x_i, y_i) of the circle is not larger than the sensing radius r_i .

A random variable I_i is defined in order to describe the event that the sensor node N covers a given point (x, y). Then, the probability of event I_i , given by $P\{I_i\}$, is equal to the coverage probability $P_{cov}(x, y, N_i)$. This probability is a binary valued function which is expressed as

$$P_{\rm cov}(x, y, N_i) = \begin{cases} 1 & \text{if } (x - x_i)^2 + (y - y_i)^2 \le r_i^2 \\ 0 & \text{otherwise,} \end{cases}$$
(2)

where a point (x, y) is covered by a sensor node N_i if and only if its distance to the center (x_i, y_i) is less than or equal to the sensing radius r_i . It is obvious that a point must be evaluated against all the nodes in the network; thus

$$S(x, y) = \sum_{i=1}^{z} P_{cov}(x, y, N_i),$$
 (3)

and here S(x, y) represents the number of times that the coordinate (x, y) is covered by the node set.

However, to find the proper covered area by the set of nodes, the function R(x, y) is calculated to check if the coordinate is covered at least once or if it is not covered at all N_i ;

$$R(x, y) = \begin{cases} 1 & S(x, y) \ge 1 \\ 0 & S(x, y) = 0. \end{cases}$$
(4)

The covered area is calculated as

$$A_{\rm cov}(N) = \frac{\sum_{y=1}^{n} \sum_{x=1}^{m} R(x, y)}{A_t}.$$
 (5)

Equation (5) is applied in our simulation by sweeping all (x, y) points of the target area $m \times n$. For different area configurations, this area is replaced by the bounded area of interest. Now, $f_1(x)$ is expressed as

$$f_1(x) = 1 - A_{\rm cov}(N)$$
. (6)

2.2. *Redundant Area*. The uncovered nonnormalized area is computed as

$$A_{\text{zeros}} = A_t \left(1 - A_{\text{cov}} \left(N \right) \right). \tag{7}$$

The redundant area is computed as

Red
$$(x, y) = \begin{cases} 1 & S(x, y) > 1 \\ 0 & S(x, y) = \{0, 1\}, \end{cases}$$
 (8)

which verify if a coordinate is covered by more than one node. As a result,

$$A_{\rm red}(N) = \frac{\sum_{y=1}^{n} \sum_{x=1}^{m} \text{Red}(x, y)}{A_t - A_{\rm zeros}},$$
(9)

where $A_{cov}(N)$ is the area covered by the nodes, $A_{red}(N)$ is the redundant area, and $A_{zeros}(N)$ is the area not covered by the nodes. Now, $f_2(x)$ is defined as

$$f_2(x) = A_{\rm red}(N)$$
. (10)

The function $f_2(x)$ is equivalent to the one given by [28].

2.3. Energy Consumption. There exist several models for wireless networks efficient energy consumption; for example, see [6, 35, 36], among others. In this paper, the total energy consumption in maintaining the network connectivity is given by

$$E_{\rm cst} = \mu \sum_{i=1}^{z} r_i^{\alpha},\tag{11}$$



FIGURE 1: Wall analysis to compute area coverage and to form a MST.

and here μ is the node sensing parameter in milliwatts per square meter (mW/m²) and r_i is the sensor radius of N_i in meters (m), and $\alpha = 2$ since our communication medium is the air [37]. In this research work $\mu = 0.005$ (mW/m²) is used. Now, $f_3(x)$ is expressed as

$$f_3(x) = E_{\rm cst}(N)$$
. (12)

Nevertheless, to normalize $f_3(x)$, it is needed to use a constant which considers the maximum value that $f_3(x)$ may obtain. Then, the possible maximum value is used in its inverse form as

$$n_3 = \frac{1}{\mu z r_{\max}^2},\tag{13}$$

and here r_{max} is the maximum sensor radius in meters (m).

2.4. Wall Behavior Analysis. To simplify the analysis of the obstacles, it has adopted on-off criterion to compute the area coverage in the same way that a channel is modelled with the well-known binary symmetric channel model [38], by omitting existing effects such as signal propagation through barriers or obstacles. A similar approach is used in [28]. Figure 1 shows how the MST should be formed. Additionally, it also shows how the area can be treated and computed if a wall crosses the area coverage of a given node. It can see that N_4 is used to connect N_3 and N_5 to prevent the wall. That means a link should not cross a wall. About the computed area coverage, dark areas in N_1 and N_5 are not computed because they are not located aside from the center of the area coverage; that is, they are located at the other side of the wall. Such restriction is considered to compute the area coverage in order to give feedback on this research work.

3. The Differential Evolution Algorithm

The DEA generates a set of N_p individuals, each one of dimension *D*. Particulary for this problem, the dimension *D* includes the coordinates (x_i, y_i) and the radii (r_i) of the sensor nodes in the same vector (x_i, y_i, r_i) . Each individual is a potential solution of the problem, generated randomly but bounded by the constraints of the problem to solve. To perform its duties, the algorithm tunes the solutions

with the following constants: the mutation constant (M), the recombination constant (C_r) , and the population size (N_p) which are defined by the user, where $0 \le M \le 1$ and $0 \le C_r \le 1$. Given *G* generations (cycles), operations like mutation, recombination, and selection are applied to each individual to obtain new and improved individuals that should be able to substitute the old ones. If it is successful, each time a generation is executed, a better suboptimal solution is obtained, which is near to the global optimal solution. The following is a description of the operators used in the DE algorithm that finds the most promising region in the search space.

3.1. Objective Functions. In this paper, the DE algorithm seeks to obtain a balance among the maximal effective coverage rate, the minimum sensor power communication, and the overlap. To convert the network coverage rate into minimal functions the opposite effect of the coverage rate is applied. Therefore, the network effective coverage rate is expressed by

Min
$$f_1(x) = 1 - A_{cov}(N)$$
, (14)

the redundant area is determined by

Min
$$f_2(x) = A_{red}(N)$$
, (15)

and the minimum network energy communication is determined by

Min
$$f_3(x) = E_{cst}(N)$$
. (16)

The values of $\text{Min } f_1(x)$, $\text{Min } f_2(x)$, and $\text{Min } f_3(x)$ are part of the fitness function for measuring the result. A linear combination of the objective functions converts the original multiobjective function into a single-objective function as follows:

Min
$$\sum_{k=1}^{K} w_k f_k(x)$$
, where $w_k \ge 0$; $\sum_{k=1}^{K} w_k = 1$, (17)

and here w_k are weight coefficients expressing the relative importance of each objective function and k is the number of objective functions that the total objective function f(x) has.

3.2. The Fitness Function. The scenario where DEA is to be applied is limited by the area boundaries and the small amount of energy for each node. Additionally, it is important, aside from the optimization of the area coverage and the energy, to improve the separation among the nodes by adding an extra function which helps to prevent the overlap. Then, the fitness function is defined by

$$f(x) = w_1 f_1(x) + w_2 f_2(x) + w_3 n_3 f_3(x), \qquad (18)$$

where w_1 , w_2 , and w_3 are weight coefficients and n_3 is the normalization coefficient of $f_3(x)$. These values are chosen in such a way that the whole objective function is in the interval [0, 1] and $w_1 + w_2 + w_3 = 1$. If the value is minimal, it means that a good combination of maximum area, minimum energy, and minimum overlap is achieved.

3.3. The Mutation Operator. Let $x^{r_1}, x^{r_2}, x^{r_3} \in \{1, ..., N_p\}$: $x^{r_1} \neq x^{r_2} \neq x^{r_3} \neq x^i$. x^{r_1}, x^{r_2} , and x^{r_3} , different from the current target vector x^i , and M is a scaling factor named as the mutation constant, which must satisfy M > 0. For each target vector x^i , $i = 1, ..., N_p$, the mutant individual m^i is created using (19) [25]:

$$m^{i} = x^{r1} + M\left(x^{r2} - x^{r3}\right),$$
(19)

where the mutation operator is used to control the magnitude of the difference between two individuals, and it has a strong influence into the convergence of DEA.

3.4. The Recombination Operator. Let t^i be the trial vector, rand a uniform random number in the interval [0, 1], and C_r the recombination constant. To obtain the trial vector from the mutation vector and the target vector, (20) holds. Consider

$$t^{i,j} = \begin{cases} m^{i,j} & \text{if rand} < C_r \ \forall j \\ x^{i,j} & \text{otherwise.} \end{cases}$$
(20)

As it is known, each individual is formed for D components in the way that $j = \{1, 2, ..., D\}$. If rand $\leq C_r$, then the *j*th element of the mutation vector is chosen to be the *j*th component of the trial vector; otherwise, the *j*th of the trial vector is selected instead. The recombination operator helps to increase the diversity in the mutation process, so an adequate assignment will lead to obtaining accurate solutions [25].

3.5. *The Selection Operator*. Let Pop_i be the vector population of the next generation. To obtain it, the selection operation checks that

$$\operatorname{Pop}_{i} = \begin{cases} t^{i} & \text{if } f(t^{i}) < f(x^{i}) \\ x^{i} & \text{otherwise.} \end{cases}$$
(21)

If $f(t^i) < f(x^i)$, it means that the trial vector is a better individual than the target vector, thus being accepted as the new individual [25]. From these improved individuals either t^i or x^i , the best solution is registered until DEA finishes its duties.

3.6. The Random-M Mutation Parameter. Let rand be a random number in the range of (0 < rand < 1) that changes during the evolution process. Then, (22) is a modified version of (19) and shows how the main operator changes with this addition:

$$m^{i} = x^{r_{1}} + \text{rand} * M(x^{r_{2}} - x^{r_{3}}).$$
 (22)

By multiplying M by rand, randomly increment or decrement its value. This permits the algorithm to get benefits from both a high convergence speed and a strong reduction of the probability of stagnation. After introducing rand, a significant improvement was perceived in the convergence of the algorithm. About other successful results, readers can read [25, 39].

4. Pseudocode for DEA in a WSN

As explained in Section 3 DEA is tuned with M, C_r , and N_p , and the number of generations is given by G. The complete population set is stored at Pop. Each individual is formed by Pop_i = $x_{i,1}, \ldots, x_{i,z}; y_{i,1}, \ldots, y_{i,z}; r_{i,1}, \ldots, r_{i,z}$, where each member belongs to each N_i formed by a triad (x_i, y_i, r_i) , (x_i, y_i) being random coordinates. To start with DEA, it is necessary to consider an initial population P_{initial} , where $P_{\text{initial}} = \text{Pop}_1$ for the sake of simplicity. Remembering the steps followed by DEA, three different individuals x^{r1}, x^{r2}, x^{r3} are obtained. Here, these individuals are obtained from Pop, where x^{r1}, x^{r2}, x^{r3} are equivalent to having $x_{r1,j}, y_{r2,j}, r_{r1,j}$.

The validation process of t^i is based on the triad (x_i, y_i, r_i) regarding the boundaries, r_{max} and r_{min} (or the smallest radio available in the node set $r_{n\text{min}}$). If $r > r_{\text{max}}$, (23) is applied to regenerate r as

$$r_{\rm adj} = r_{\rm min} + w_1 * \left(\frac{{\rm rand}\left(\right)}{2} + \frac{1}{2}\right) * \left(r_{\rm max} - r_{\rm min}\right).$$
 (23)

On the other hand, if (x, y) are outside the boundaries, (x, y) coordinates are regenerated. One goal is to reduce the overlap of these nodes on each population member, and then in this way better energy and area combinations will be obtained. A distance among the nodes should be maintained to prevent excessive separations and this condition is controlled by the largest radius of the involved nodes multiplied by a parameter *scale*, typically r_{max} . The easiest way is to check the triad (x_i, y_i, r_i) of N_i regarding to its neighbors. A strategy to prevent wasting area is to move (x_i, y_i) away from the boundaries. This is also controlled by *scale* that multiplies r_i . In this work, wall scenarios are included, and it is important to note that it is possible that an area coverage of N_i can be cut out when it is crossing a wall. Then, the effective area coverage will be identified in the side where (x_i, y_i) is located. Additionally, to check if the obtained area covers enough surface, A_{th} is used as the minimum percentage acceptable area. Here, the parameter scale can be treated as fixed scale or variable *scale* = *scale* + *delta*.

The Prim algorithm [40] obtains the minimum spanning tree (MST) while the Dijkstra algorithm [41] gives a star topology. Prim-Dijkstra [42] uses the location of the sensor nodes and determines a topology that is a mixture of both MST and a star topology. In this research work, the MST is formed by using arbitrarily a central node (cnode = 5) and the maximum distance to generate a link ($r_i + r_j$, $i \neq j$). If $d(N_i, N_j) > r_i + r_j$, the link does not exist. To satisfy this constraint the Prim-Dijkstra algorithm [42] is used. Prim-Dijsktra will perform its duties when each link of the MST does not cross forbidden zones, boundaries, or walls; otherwise the nodes that do not fulfill these restrictions will not be neighbors of N_i .

The objective function launches the area, the energy, and the redundancy aside from the fitness. As it has been described through this work, the minimum fitness indicates that efficient combination of area usage, energy, and redundancy has been obtained.

Finally, the initial and final location of sensor nodes are given by Pop_{initial} and Pop_{best}, respectively. The Hungarian

procedure DEA
Set the control parameters
Create Pop
Evaluate fitness, area, energy, redundancy
Start generations counter $s = 0$
Repeat until stop conditions are completed:
(tree is built and $s > G$) or
(tree is built and area coverage $\geq A_{th}$)
for $i = 1$ to N_p do
Select from Pop
Mutation m^i , equation (19);
Recombination t^i , equation (20);
Validate t^i .
Update (x, y, r) if needed;
De-overlap t^i , update (x, y) if needed;
Selection, equation (21)
end for
Pop _{best} fitness, area, energy and redundancy
Apply Hungarian algorithm to Pop _{best} versus Pop _{initial} .
Apply Prim-Dijkstra(cnode, α) to array of links;
if Some nodes are not connected to the tree then
Continue;
else nodes are connected
Tree has been build.
end if
s = s + 1
<pre>scale = scale + delta (Variable option until G)</pre>
End repeat
return Pop _{best} , <i>Fitness</i> , <i>Area</i> , <i>Energy</i> , <i>redundancy</i>
end procedure

ALGORITHM 1: DEA applied to WSN.

Algorithm [34] helps to find the minimal distance that a node will move away to the ending location from its original location. It receives Pop_{best} and $Pop_{initial}$ to determine the final location of the sensors with respect to the initial location. By using (x_i, y_i) , the Hungarian Algorithm uses a matrix of distances, where each distance is calculated by using the initial coordinates with respect to the final coordinates. The initial coordinate of a node is evaluated against all the final coordinates, to obtain the desired minimum distances. Then, the solution with the minimum distance in which each node goes from the initial to the final position is chosen. In this work, in scenarios with walls, the distance from one node to any other one is through the branches of the MST preventing crossing any wall or obstacle. Algorithm 1 summarizes the behavior of DEA.

5. Numerical Results and Discussion

The algorithm is implemented in machines with the following characteristics:

MATLAB R2013a. Windows 7 Enterprise 2009, service pack 1. Processor: Intel (R) Xeon (R) CPU. W3690 @ 3.47 GHz 3.46 GHz. Installed memory (RAM): 8.00 GB.

System type: 64-bit operating system.

For the inner star scenario, the effective sensor radius is between 3.5 and 8 meters, and, the weights are $w_1 = 3*0.90/4$, $w_2 = 0.90/4$, and $w_3 = 0.10$. Here, two different strategies were tested. In the first one, the value for *scale* is fixed to 0.7000, while in the second one *scale* is increased in steps of 0.0070.

For the wall scenarios, the effective sensor radius of the sensor nodes is between 6 and 8 meters and the target areas are 80 (m) × 80 (m). DEA was tested based on the increment of *scale* in steps of 0.008 units when each generation is executed, thus controlling the separation among nodes instead of spreading them immediately and also helping with the speed of the algorithm. Here, $w_1 = 3 * 0.8/4$, $w_2 = 0.8/4$, and $w_3 = 0.2$. If the node coverage area is crossed by a wall, the region of the coverage area that does not contain the center of the node should not be considered in the computation of the area because the wall reduces the signal propagation. For simplification purposes in the calculation of the area coverage on each node, a criteria on-off is adopted. If the center is behind a wall, the area crossed by a wall is not considered.

As it was explained in the previous section, the algorithm stops when the number of established generations is reached and the tree is formed, or a threshold area is reached and the tree is formed. The sensor nodes are identified as $N_1, N_2, ..., N_i, ..., N_z$, and the circles show their sensor range. The values of M = 0.8, $N_p = 35$, and $C_r = 0.2$ are used.

For all, the inner star and wall scenarios, 40 tests were computed using both variable and fixed criteria. The percentage of covered area, energy consumption in maintaining the network connectivity, redundant area, and radii values are determined and analyzed. Also, the execution time for each test is determined. Increasing the number of sensor nodes, the noncovered areas of all scenarios can be solved. The initial and final location of sensor nodes can be obtained.

5.1. Inner Bounded Star. The set of Figure 2 show the final distribution obtained with fixed and variable criteria, when DEA is applied. The initial positions of the nodes are randomly set, without forming a tree. The star is delimited by the following parameters: s = 12, p = (0 : s), $r_1 = 40$, $r_2 = 20$, $t = (-1/4 : 1/s : 3/4) * 2 * \pi$, $r = (r_1 + r_2)/2 + (r_1 - r_2)/2 * (-1)^p$, and

$$xv = r * \cos(t) + 40,$$

 $yv = r * \sin(t) + 40,$
(24)

where xv and yv are used in the Matlab command inpolygon (inpolygon(x, y, xv, yv)) and (x, y) are the coordinates of the point to be tested whether it is inside the figure or not.

Testing the fixed criterion, the final distribution fits the inner star as it can be seen in Figure 2(a), the smallest nodes appear as connector between the nodes with larger area, and sometimes they appear at peaks of the star. Additionally, the nodes are distributed regarding the overlaps, showing that the algorithm tries to arrange the smallest nodes into the peaks or at least to arrange these nodes with an average area coverage. It is noted that the tree is formed successfully. Also, the results

Criteria	Area%	Energy (mW)	Radii (m)	Redundancy%	Time (s)	Gen.
Variable	91.83	6.1192	6.2856	48.37	546.9349	100
Fixed	93.33	5.3197	5.8639	37.19	889.1790	100

TABLE 1: Results for inner star. Tests # 24, 30 nodes.



FIGURE 2: Examples of final distribution inner star, (a) fixed criterion, (b) variable criterion.

of the variable criterion (Figure 2(b)) show that the nodes are distributed; nevertheless the overlap is increased.

Table 1 shows examples of fixed and variable criteria with 30 nodes and effective area of 2399.9200 m^2 under the scenario shown in Figure 2. It can be seen that both cases converge to 100 generations. Even when the fixed case presents better metrics than the variable one, the time spent to compute the solution is larger (fixed case has 889.1790 (s) while variable case has 546.9349 (s)). To state and prove this trend, it is needed to perform more tests.

Table 2 shows the results using 30 nodes and an effective area of 2399.9200 m² for the tests numbers 31/40 and 35/40 of the 40 tests under the scenario shown in Figure 2; the mean and variance values of time, radii, percentage of redundancy, energy, and the covered area of the network were obtained. Here, both fixed and variable criteria are compared under the same conditions. The results are the average of these tests using 100 generations. The variable criterion is faster than the fixed one in 1.6527 times. It can be explained because the separation among the nodes (*scale* parameter) is relaxed for the variable criterion has a better solution than the variable one (which is reflected in the area coverage 89.66% versus 88.90%, energy 5.0037 (mW) versus 5.3347 (mW), and overlap 34.39% versus 40.94%).

For the next scenarios one will choose the variable criterion due to its high speed and low variance in terms of

TABLE 2:	Inner	star.	Mean	and	variance	results,	variable	and	fixed
riteria.									

Test	Variable	Fixed	Gen.
Success test/40	35/40	31/40	100
Mean time (s)	544.8065	900.4085	100
Mean redundant (%)	40.94	34.39	100
Mean radii (m)	5.8493	5.6670	100
Mean energy (mW)	5.3347	5.0037	100
Mean area (%)	88.90	89.66	100
Var time	6.6988	117.3945	100
Var redundant	0.0016	0.0019	100
Var radii	0.0330	0.0294	100
Var energy	0.0955	0.0877	100
Var area	2.3411e - 004	2.6960e - 004	100

area coverage and time execution in the search of the solution space. The solution here obtained with the variable criterion is as good as the fixed one. Adequate tuned parameters and weights will help the variable criteria to be near the fixed one.

It is important to note that Figure 2(a) shows only the optimized location of the sensor nodes. The initial random location and the final optimized location of sensor nodes in Figure 2(a) in terms of (x, y) coordinates are shown in Table 3.

TABLE 3: Inner star, fixed criterion. Initial and final location of sensors nodes.

Node	X_{initial} (m)	$Y_{\text{initial}}(\mathbf{m})$	X_{final} (m)	Y_{final} (m)
N_1	38.9	45.8	39.7	52.8
N_2	14.2	58.7	18.3	52.5
N_3	42.8	73.4	39.8	70.5
N_4	60.9	49.0	60.6	52.7
N_5	54.2	40.1	58.7	33.1
N_6	40.4	40.6	42.6	43.1
N_7	60.4	58.2	66.6	55.3
N_8	41.6	62.8	40.7	64.4
N_9	43.8	13.1	40.6	12.0
N_{10}	52.4	51.8	52.7	48.2
N_{11}	28.8	39.2	28.7	38.3
N_{12}	28.4	36.0	23.6	35.4
N_{13}	15.3	47.1	21.8	45.3
N_{14}	28.0	26.5	19.8	26.7
N_{15}	47.1	57.7	47.8	56.1
N_{16}	27.5	24.1	33.0	23.3
N_{17}	41.7	29.0	40.9	29.2
N_{18}	18.0	49.0	34.9	49.8
N_{19}	37.9	19.5	40.6	20.3
N_{20}	19.8	51.4	27.2	51.5
N_{21}	58.3	43.8	58.9	45.2
N_{22}	9.5	25.3	11.4	24.7
N_{23}	31.9	35.7	36.2	34.2
N_{24}	27.1	28.1	25.6	29.4
N_{25}	46.8	28.9	52.8	28.5
N_{26}	41.2	21.9	46.8	26.8
N_{27}	29.7	28.2	44.9	34.1
N_{28}	25.8	54.2	35.2	56.8
N_{29}	48.2	49.9	50.4	38.4
N_{30}	57.3	39.0	65.6	26.5

Now, four sets of walls are analyzed. The results were obtained using 100 generations for the first, second, and fourth set of walls and 120 generations for the third set of walls in the DEA.

5.2. First Set of Walls. The set of walls shown in Figure 3 can be constructed by setting the walls through the coordinates $x_a = 56 : 80, y_a = 40, x_b = 50, y_b = 0 : 24, x_c = 30, y_c = 0 : 24, x_d = 0 : 24, y_d = 40, x_e = 30, y_e = 56 : 80, and x_f = 50, y_f = 56 : 80. The distribution of the nodes is affected by the position of the walls. It is also noted that the final tree avoids crossing the walls, accomplishing the restriction of not crossing any obstacle. Some nodes lost their area coverage because of the walls; nevertheless other nodes can compensate that effect by covering the affected area. It is important to note that the distances between the walls and nodes 28 and 59 are 70 (cm) and 120 (cm), respectively. The circles are arranged far from the area boundary.$

Table 4 shows the results obtained in the test number 5 of the 40 tests under the scenario shown in Figure 3. The covered



FIGURE 3: Final distribution of nodes for the first set of walls.



FIGURE 4: Final distribution of nodes for the second set of walls.

area is greater than 92%, the energy is 16.8013 (mW), and the redundancy is lower than 53%.

5.3. Second Set of Walls. The wall shown in Figure 4 can be constructed by setting the wall through the coordinates $x_a = 1 : 60$, $y_a = 45$. It can be seen that the MST goes around the wall and the nodes are arranged away from the boundaries.

Table 5 shows the obtained parameters with the test number 24 of the 40 tests under the scenario shown in Figure 4. The area is greater than 93%, the energy is 16.8589 (mW), and the redundancy is lower than 52%.

Criteria	Area%	Energy (mW)	Radii (m)	Redundancy%	Time (s)	Gen.
Variable	92.18	16.8013	7.1866	52.24	1226.3465	100

TABLE 5: Results for the second set of walls, test # 24.

Criteria	Area%	Energy (mW)	Radii (m)	Redundancy%	Time (s)	Gen.
Variable	93.87	16.8589	7.1981	51.32	1498.9526	100

TABLE 6: Results for the third set of walls, test 38.

Criteria	Area%	Energy (mW)	Radii (m)	Redundancy%	Time (s)	Gen.
Variable	92.88	16.5964	7.1421	50.69	1739.6244	120



FIGURE 5: Final distribution of nodes for the third set of walls.

5.4. Third Set of Walls. Case three was configured as $x_a = 30$, $y_a = 11 : 70$, $x_b = 50$, and $y_b = 11 : 70$, which seems as a set of parallel walls, shown in Figure 5. It can be seen that the tree goes around the wall and the nodes are arranged away from the boundaries. Here, also, it is important to note that the distances between the walls and nodes 53 and 65 are 80 (cm) and 70 (cm), respectively.

Table 6 shows the obtained parameters with the test number 38 of the 40 tests under the scenario shown in Figure 5. The area is greater than 92%, the energy is 16.5964 (mW), and the redundancy is lower than 51%. Here, a case is presented where the number of generations is larger compared to the established one, 120 generations instead of 100 generations. As it can be shown through this paper, sometimes the algorithm needs more generations to obtain a tree.

5.5. Fourth Set of Walls. Case four was configured as $x_a = 1$: 25, $y_a = 40$, $x_b = 76$: 80, $y_b = 40$, and $x_c = 40$, $y_c = 26$: 75 which seems as a divided corridors inside a room, shown in



FIGURE 6: Final distribution of nodes for the fourth set of walls.

Figure 6. It can be seen that the tree goes around the wall and the nodes are arranged away from the boundaries.

Table 7 shows the obtained parameters with case 38 of the 40 tests under the scenario shown in Figure 6. The area is greater than 93%, the energy is 16.7526 (mW), and the redundancy is lower than 52%.

It is important to remember that, to locate one sensor respect any other one, it is simple through the branches of the MST. For example, the distance between sensor nodes N_7 and N_{20} , in Figure 6, is not the euclidian distance between them. It is necessary to follow the route $N_7 \rightarrow N_{38} \rightarrow N_{62} \rightarrow N_{19} \rightarrow N_{39} \rightarrow N_{16} \rightarrow N_{20}$ and add each distance to obtain the total distance between sensor nodes N_7 and N_{20} .

5.6. Summary. Table 8 shows a comparison summary of the parameters obtained with DEA such as the generation number in which the sensors configuration was obtained, the area covered, the energy average of the network, and the degree of overlap. The same 40 tests were done for each

TABLE 4: Results for the first set of walls, test # 5.

TABLE 7: Results for the fourth set of walls, test (38.
--	-----

Criteria	Area%	Energy (mW)	Radii (m)	Redundancy%	Time (s)	Gen.
Variable	93.19	16.7526	7.1761	51.99	1480.5491	100

TABLE 8: Mean and variance results for the four set of walls, with 65 nodes and effective area of 6400.0000 (m²).

Case	Case 1	Case 2	Case 3	Case 4
Success test/40	34/40	40/40	36/40	39/40
Mean time (s)	1228.6005	1417.2937	1531.0622	1515.7459
Mean redundant (%)	50.13	53.05	50.50	51.40
Mean radii (m)	7.1963	7.1891	7.1879	7.1858
Mean energy (mW)	16.8495	16.8144	16.8092	16.7988
Mean area (%)	92.25	93.83	93.24	93.41
Var time	4.4281	1187.1325	1735.6760	1129.1838
Var redundant	1.7137e - 004	1.2889e - 004	1.8643e - 004	1.7708e - 004
Var radii	9.4720e - 004	4.4480e - 004	9.6199e - 004	6.5851e - 004
Var energy	0.0213	0.0097	0.0210	0.0143
Var area	3.2485e - 005	1.9754e - 005	2.9491 <i>e</i> - 005	3.6057e - 005

case, using 100 generations. The area average is larger than 93% with an energy average less than 16.85 (mW) and the redundancy average less than 54%. However, as it can be shown in Table 8, the covered area is quite similar in all cases. The rest of the nodes near the wall are compensated with the nodes at the other side of the wall.

6. Conclusions

This research work shows the feasibility to apply the DEA in presence of obstacles or strict restrictions. Results obtained with the inner star show that the circles can be arranged even into the peaks of the star by using either large or small circles. Results could be improved if more restrictions are included, for instance, the area management on the peaks, where the smallest circle can be placed instead of the largest circles. Even so, the results seem promising.

With respect to the presented cases with obstacles, the construction of the trees and the distribution and location of the nodes are feasible. The trees avoid crossing the obstacles and the circles have a separation among them and regarding the area boundaries. This is shown because the area obtained is greater than 90% for all the cases with a maximum overlap of 51%. The presented results can be improved if a restriction regarding the obstacles is adopted moving the circles away of the barriers, obtaining more useful area for each circle. Nonetheless, the DEA minimizes the wasted area by putting the neighbor nodes of the circle that lose its area coverage. This work is the initial step to study the distribution of sensor nodes in presence of obstacles, represented in this work as a set of given walls. An accurate modelling regarding the signal propagation and a larger separation among the nodes could also improve the results presented in this research work. As further work it is recommended to implement the flocking of multiagents with virtual leader technic described in Section 1 in order to move the sensors in the network considering the position and velocity of the centre of mass

of all sensors (agents). Also, it is important to implement the case with multiple leaders (containment control) focused on both switching networks and fixed networks described in Section 1, in which the input signals of sensors may be saturated in practice.

Conflict of Interests

The authors declare that they have no competing interests.

Acknowledgment

This research was partially supported by School of Engineering and Sciences of Tecnológico de Monterrey.

References

- I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–105, 2002.
- [2] P. Santi, "Topology control in wireless ad hoc and sensor networks," ACM Computing Surveys, vol. 37, no. 2, pp. 164–194, 2005.
- [3] K. P. Eswaran and R. E. Tarjan, "Augmentation problems," *SIAM Journal on Computing*, vol. 5, no. 4, pp. 653–665, 1976.
- [4] W.-T. Chen and N.-F. Huang, "The strongly connecting problem on multihop packet radio networks," *IEEE Transactions on Communications*, vol. 37, no. 3, pp. 293–295, 1989.
- [5] L. M. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc, "Power consumption in packet radio networks," *Theoretical Computer Science*, vol. 243, no. 1-2, pp. 289–305, 2000.
- [6] M. X. Cheng, M. Cardei, J. Sun et al., "Topology control of ad hoc wireless networks for energy efficiency," *IEEE Transactions* on Computers, vol. 53, no. 12, pp. 1629–1635, 2004.
- [7] B. Xiao, L. Chen, Q. Xiao, and M. Li, "Reliable anchor-based sensor localization in irregular areas," *IEEE Transactions on Mobile Computing*, vol. 9, no. 1, pp. 60–72, 2010.

- [8] W. Shi and V. W. S. Wong, "A two-phase algorithm for locating sensors in irregular areas," in *Proceedings of the 53rd IEEE Global Communications Conference (GLOBECOM '10)*, pp. 1–6, Miami, Fla, USA, December 2010.
- [9] V. Rodoplu and T. H. Meng, "Minimum energy mobile wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1333–1344, 1999.
- [10] L. Li, J. Y. Halpern, P. Bahl, Y.-M. Wang, and R. Wattenhofer, "A cone-based distributed topology-control algorithm for wireless multi-hop networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 1, pp. 147–159, 2005.
- [11] A. S. Majid and E. Joelianto, "Optimal sensor deployment in non-convex region using discrete particle swarm optimization algorithm," in *Proceedings of the 1st IEEE Conference on Control, Systems and Industrial Informatics (ICCSII '12)*, pp. 109–113, IEEE, Bandung, Indonesia, September 2012.
- [12] N. A. B. Ab Aziz, A. W. Mohemmed, and M. Y. Alias, "A wireless sensor network coverage optimization algorithm based on particle swarm optimization and voronoi diagram," in *Proceedings of the IEEE International Conference on Networking*, *Sensing and Control (ICNSC '09)*, pp. 602–607, Okayama, Japan, March 2009.
- [13] H. Safa, W. El-Hajj, and H. Zoubian, "Particle swarm optimization based approach to solve the multiple sink placement problem in WSNs," in *Proceedings of the IEEE International Conference on Communications (ICC '12)*, pp. 5445–5450, Ottawa, Canada, June 2012.
- [14] D. S. Deif and Y. Gadallah, "Classification of wireless sensor networks deployment techniques," *IEEE Communications Surveys* and Tutorials, vol. 16, no. 2, pp. 834–855, 2014.
- [15] R. V. Kulkarni and G. K. Venayagamoorthy, "Particle swarm optimization in wireless-sensor networks: a brief survey," *IEEE Transactions on Systems, Man and Cybernetics—Part C: Applications and Reviews*, vol. 41, no. 2, pp. 262–267, 2011.
- [16] A. Tamizharasi, R. Arthi, and K. Murugan, "Bio-inspired algorithm for optimizing the localization of wireless sensor networks," in *Proceedings of the 4th International Conference* on Computing, Communications and Networking Technologies (ICCCNT '13), pp. 1–5, Tiruchengode, India, July 2013.
- [17] X. Zhang, T. Wang, and J. Fang, "A node localization approach using particle swarm optimization in wireless sensor networks," in *Proceedings of the International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI '14)*, pp. 84–87, IEEE, Beijing, China, October 2014.
- [18] H. Su, X. Wang, and Z. Lin, "Flocking of multi-agents with a virtual leader," *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 293–307, 2009.
- [19] H. Su and M. Z. Chen, "Multi-agent containment control with input saturation on switching topologies," *IET Control Theory & Applications*, vol. 9, no. 3, pp. 399–409, 2015.
- [20] H. Su, G. Jia, and M. Z. Q. Chen, "Semi-global containment control of multi-agent systems with intermittent input saturation," *Journal of the Franklin Institute*, vol. 352, no. 9, pp. 3504–3525, 2015.
- [21] A. Nordio and C.-F. Chiasserini, "Field reconstruction in sensor networks with coverage holes and packet losses," *IEEE Transactions on Signal Processing*, vol. 59, no. 8, pp. 3943–3953, 2011.
- [22] Z. Mi, Y. Yang, and H. Ding, "Self-organized connectivity control and optimization subjected to dispersion of mobile ad hoc sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 672436, 15 pages, 2012.

- [23] L. Liu and H. Ma, "On coverage of wireless sensor networks for rolling terrains," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 1, pp. 118–125, 2012.
- [24] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341– 359, 1997.
- [25] V. Feoktistov, Differential Evolution: In Search of Solutions, Springer Optimization and Its Applications, Springer, New York, NY, USA, 2006.
- [26] S. Das and P. N. Suganthan, "Differential evolution: a survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.
- [27] L. Jin, J. Jia, and D. Sun, "Node distribution optimization in mobile sensor network based on multi-objective differential evolution algorithm," in *Proceedings of the 4th International Conference on Genetic and Evolutionary Computing (ICGEC* '10), pp. 51–54, IEEE, Shenzhen, China, December 2010.
- [28] J.-H. Seok, J.-Y. Lee, C. Oh, J.-J. Lee, and H. J. Lee, "RFID sensor deployment using differential evolution for indoor mobile robot localization," in *Proceedings of the 23rd IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems (IROS '10)*, pp. 3719–3724, Taipei, Taiwan, October 2010.
- [29] S. Hojjatoleslami, V. Aghazarian, and A. Aliabadi, "DE based node placement optimization for wireless sensor networks," in *Proceedings of the 3rd International Workshop on Intelligent Systems and Applications (ISA '11)*, pp. 1–4, Wuhan, China, May 2011.
- [30] D. Qiao and G. Pang, "A modified differential evolution with heuristics algorithm for non-convex optimization on sensor network localization," *IEEE Transactions on Vehicular Technol*ogy, 2015.
- [31] R. Harikrishnan, V. J. S. Kumar, and P. S. Ponmalar, "Differential evolution approach for localization in wireless sensor networks," in *Proceedings of the IEEE International Conference* on Computational Intelligence and Computing Research (ICCIC '14), pp. 1–4, Coimbatore, India, December 2014.
- [32] J. Kuriakose, V. Amruth, and N. S. Nandhini, "A survey on localization of wireless sensor nodes," in *Proceedings of the International Conference on Information Communication and Embedded Systems (ICICES '14)*, pp. 1–6, Chennai, India, Feburary 2014.
- [33] L. Cheng, C. Wu, Y. Zhang, H. Wu, M. Li, and C. Maple, "A survey of localization in wireless sensor network," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 962523, 12 pages, 2012.
- [34] J. Munkres, "Algorithms for the assignment and transportation problems," *Journal of the Society for Industrial and Applied Mathematics*, vol. 5, no. 1, pp. 32–38, 1957.
- [35] U. K. Chakraborty, S. K. Das, and T. E. Abbott, "Energyefficient routing in hierarchical wireless sensor networks using differential-evolution-based memetic algorithm," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC* '12), pp. 1–8, Brisbane, Australia, June 2012.
- [36] H. Ali, W. Shahzad, and F. A. Khan, "Energy-efficient clustering in mobile ad-hoc networks using multi-objective particle swarm optimization," *Applied Soft Computing Journal*, vol. 12, no. 7, pp. 1913–1928, 2012.
- [37] T. Rappaport, Wireless Communications: Principles and Practice, Prentice Hall, Upper Saddle River, NJ, USA, 2nd edition, 2001.
- [38] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

- [39] F. Lezama, G. Castañón, and A. M. Sarmiento, "Differential evolution optimization applied to the wavelength converters placement problem in all optical networks," *Computer Net*works, vol. 56, no. 9, pp. 2262–2275, 2012.
- [40] R. C. Prim, "Shortest connection networks and some generalizations," *Bell System Technical Journal*, vol. 36, no. 6, pp. 1389– 1401, 1957.
- [41] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
- [42] R. S. Cahn, Wide Area Network Design: Concepts and Tools for Optimization, The Morgan Kaufmann Series in Networking, Morgan Kaufmann, Boston, Mass, USA, 1998.

Research Article

Adaptation Algorithm of Geometric Graphs for Robot Motion Planning in Dynamic Environments

Jae-Han Park, Ji-Hun Bae, and Moon-Hong Baeg

Robotics R&D Group, Korea Institute of Industrial Technology (KITECH), Ansan, Republic of Korea

Correspondence should be addressed to Jae-Han Park; hans1024@kitech.re.kr

Received 22 September 2015; Accepted 6 December 2015

Academic Editor: Hou-Sheng Su

Copyright © 2016 Jae-Han Park et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This study proposes an adaptive graph algorithm for collision-free motion planning of articulated robots in dynamic environments. For this purpose, deformations of the configuration space were analyzed according to the changes of the workspace using various simulations. Subsequently, we adopted the principles of gas motion dynamics in our adaptation algorithm to address the issue of the deformation of the configuration space. The proposed algorithm has an adaptation mechanism based on expansive repulsion and sensory repulsion, and it can be performed to provide the entire adaptation using distributed processing. The simulation results confirmed that the proposed method allows the adaptation of the roadmap graph to changes of the configuration space.

1. Introduction

Robot motion planning in dynamic environments has been regarded by researchers as a challenging problem. The motion planning problem, particularly for articulated robots, is known to be difficult because of the geometrical and algebraic complexities resulting from the increase in the degrees of freedom (DOFs) of the robot. The representative statement of this motion planning problem is known as the configuration space formulation. The key function of a configuration space is to represent the robot as a point in an appropriate space and map the obstacles in this space. This mapping transforms the problem of planning the motion of a dimensioned object into a planning problem for the motion of a point. It also makes the motion constraints of the robot more explicit [1–3].

For collision-free manipulation, especially in dynamic environments, planning algorithms must overcome additional challenging problems [4, 5]. The first problem is the unpredictable radical deformation of the configuration space owing to the change of the workspace. Various simulations have shown that the extent of the changes in the shape of the obstacles in the configuration space is closely related to the distance between the robot and the obstacles in the workspace. As a result, small changes in the workspace could cause extreme deformation of the configuration space when the body of the robot approaches the obstacles in the workspace. Thus, it is essential to manage these radical deformations of the configuration space. The second problem is the large processing time required for reconstructing the configuration space, because articulated robots generally have high-DOF kinematic structures.

Although extensive work has been performed on robot motion planning, most of the widely used algorithms utilize sampling-based methodologies. These sampling-based methods construct geometric graphs in collision-free configuration spaces to obtain solutions to motion planning problems; moreover, they employ a variety of strategies for generating samples and connecting the samples with paths. Sampling-based algorithms, such as probabilistic roadmaps (PRM) [6] and rapidly exploring random trees (RRT) [7], have been shown to perform well in practice [8-11] and have theoretical guarantees, including probabilistic completeness [12, 13]. Recently, with the increasing requirements for the use of robots in dynamic or time-varying environments, the extension of the motion planning problem to dynamic environments has emerged as an important issue. The application of sampling-based planning algorithms is mostly limited to static environments because they cannot guarantee fast responses to the processing time constraints and do not have the computing abilities to manage the deformation of the



FIGURE 1: Mapping relations between the workspace and the configuration space.

configuration space. Some studies have attempted to extend sampling-based methods to dynamic environments by incorporating the notion of time as an additional dimension in the configuration space [14, 15]. Other approaches are based on the replanning method, which interleaves planning with execution, in order to study unpredictable environments [5, 16, 17]. Nevertheless, it may be difficult for these algorithms to handle dynamic constraints in real applications because the amount of computation can increase exponentially with the extension of the configuration space dimensions.

In this paper, we propose an algorithm for geometric graphs that can adapt to a changeable configuration space topology. We contrived this algorithm via inspiration from recent investigations, such as multiagent systems [18–20], consensus control [21–24], and distributed computing [25–27]. In the proposed adaptation algorithm, each node moves in a free configuration space independently, similar to the intelligent agent of multiagent systems, by detecting the deformation of the configuration space; then, the graph updates its topology with respect to the changed node information. It was confirmed that the graph can perform the entire adaptation algorithm. Moreover, the algorithm is based on the distributed computation algorithm; hence, it could be executed in parallel by many-core systems.

2. Dynamic Environments

2.1. Configuration Space Formulation. The configuration space \mathbf{Q} is the space of all possible configurations of the system; a point \mathbf{q} in this space fully describes the volume of the robot $\mathbf{R}(\mathbf{q})$ in the workspace \mathbf{W} [1, 2]. Figure 1 shows the mapping relations between the workspace and the configuration space for a two-link manipulator. In articulated robots, the configuration space \mathbf{Q} generally represents the joint space, which is a compact subset of the *n*-dimensional Euclidean space \mathbf{R}^n , where *n* is the DOF of the robot. The configuration space can be partitioned into two regions, one representing

the collision-free space \mathbf{Q}_{free} and the other representing the region $\mathbf{Q}_{\text{collide}}$, in which the robot collides with the workspace obstacles. Every obstacle WO_i in the workspace is mapped as a *configuration space obstacle* QO_i :

$$QO_i = \left\{ \mathbf{q} \in \mathbf{Q} \mid \mathbf{R}\left(\mathbf{q}\right) \cap WO_i \neq \phi \right\}. \tag{1}$$

The union of all the configuration space obstacles

$$\mathbf{Q}_{\text{collide}} = \bigcup_{i=1}^{m} QO_i \tag{2}$$

is called the configuration obstacle region, and the set

$$\mathbf{Q}_{\text{free}} = \mathbf{Q} \setminus \mathbf{Q}_{\text{collide}} = \mathbf{Q} \setminus \bigcup_{i=1}^{m} \mathbf{Q}O_i$$

$$= \left\{ \mathbf{q} \in \mathbf{Q} \mid \mathbf{R}(\mathbf{q}) \cap \left(\bigcup_{i=1}^{m} WO_i\right) = \phi \right\}$$
(3)

is called the *free space* or *free configuration space* \mathbf{Q}_{free} . Any configuration \mathbf{q} in \mathbf{Q}_{free} is called a free configuration, indicating that the robot does not collide with any obstacles in the workspace.

2.2. Deformation of Configuration Space. Dynamic environments are workspaces in which various changes can occur. Changes in the workspace lead to deformation of the configuration space; this relationship between the two spaces is determined by the robot kinematics equation $\mathbf{R}(\mathbf{q})$. In general, the relationship between the workspace and the configuration space is nonlinear because almost all robot kinematics equations are described by nonlinear functions. Thus, the relations between the two spaces are difficult to describe and analyze mathematically. By observing various deformations of the configuration space, we can categorize them into scaling, translation, and merging deformations.



FIGURE 2: Free configuration space ratio with respect to the distance of the robot base from an obstacle in the workspace.

2.2.1. Scaling Deformation. Obstacles in the configuration space change in size according to the change of the workspace. The size of an obstacle is determined by the distance between the coordinate of the robot base and that of the obstacle in the workspace. Figure 2 shows the ratio of free configuration space in the two-dimensional workspace and in the configuration space as a function of the distance between the obstacle and the robot base in the workspace. Obstacles placed outside the working area of the robot do not exist in the configuration space. However, if an obstacle moves toward the robot base within the robot working area, the ratio of free configuration space is gradually decreased. In particular, the ratio of configuration space rapidly decreases when the obstacle and the robot base are close. In Figure 2, there is a radical decrease of 70% to 10% in the ratio of free configuration space when the distance decreases from 30 cm to 10 cm. These results confirm that the change of distance between the obstacle and the robot base changes the scale of the obstacle in the configuration space; in particular, when the distance is small, the change of scale could be radical.

Figure 3 shows the deformation of the configuration space for varying obstacle distances in the workspace. The left and right images show the configuration space and the corresponding workspace, respectively. In Figure 3(a), the entire space is the free configuration space because all objects are placed outside the working area of the robot. Figure 3(b) shows the introduction of an obstacle into the configuration space caused by the movement of the obstacle in the workspace. As a result, the ratio of free configuration space decreases to 93.3%. Furthermore, Figure 3(c) shows the move radical deformation of the configuration space by the approach of the obstacle to the robot base. Note that while the movement distances from Figures 3(a) to 3(b) and from Figures 3(b) to 3(c) are similar, the changes of the obstacle size and the free configuration space ratio are larger in Figure 3(c).

2.2.2. Translation Deformation. The position of an obstacle in the configuration space is determined by the direction of the obstacle in the workspace with respect to the robot base coordinate. Thus, if the obstacle position changes while preserving the distance from the robot base in the workspace, a translation motion of the obstacle occurs in the configuration space. Figure 4 shows a simplified example of the result of the translation motion in the configuration space. In Figure 4(a), the obstacle is placed at 0° with respect to the robot base coordinate in the workspace; in the configuration space, the obstacle is located at the origin. If the obstacle moves to the direction of 90° while maintaining the same distance, there is a translation motion to the right direction of the obstacle in the configuration space without a notable change of size, as in Figure 4(b). In contrast, as shown in Figure 4(c), if the obstacle moves to the direction of -90° , there is a translation motion to the left of the obstacle in the configuration space. From these results, it is evident that the change of direction of the object with respect to the robot base coordinate in the workspace produces translation motions of the obstacle position in the configuration space.

2.2.3. Merging Deformation. Obstacles in the configuration space consist of the union set of all obstacles converted from the workspace. In general, there is no merging phenomenon in the workspace because changes in the workspace are rigid body motions without a scale change. However, in the configuration space, there can be a merging phenomenon with respect to the change of scale and position of the obstacles. Figure 5 shows a merging process in the configuration space according to the movements of multiple objects in the workspace. The merging phenomenon can be more prominent if the objects and the robot are close; this complicates the prediction of the deformation of the configuration space.

In the mobile manipulation problem, which is a challenging task in robot motion planning, the robot position and direction change by the movement of the mobile robot. If the robot moves by turning, the obstacles near the robot move by turning in the opposite direction in the workspace; thus, it can be predicted that all these obstacles move in the same direction. In addition, if the robot base coordinate moves by the motion of the mobile base, some obstacles will be closer, and some will be farther from the robot base. In this case, some obstacles in the configuration space will be expanded, and some will be contracted, according to their distances. Therefore, in mobile manipulation problems involving the movement of the mobile base, the environments become dynamic even when the workspace is a fixed environment. Figure 6 shows the deformation of the configuration space when the robot base moves in different directions. As shown in Figures 6(b) and 6(c), closer obstacles in the workspace undergo expansion, and farther obstacles undergo contraction; furthermore, some merging occurs by the expansions of an obstacle.

In dynamic environments, the deformations of the configuration space are combinations of the three nonlinear deformations of *scaling*, *translation*, and *merging*. Sometimes, small changes in the workspace can generate considerable deformations in the configuration space; the combined changes from multiple obstacle movements in the workspace appear as a complex deformation in the configuration space. These radical deformations of the configuration space further



FIGURE 3: Scaling deformation of the configuration space caused by the movement of a workspace object. (a) No object in the configuration space; (b) appearance of the configuration space object; (c) expansion of the configuration space object by movement toward the robot in the workspace.



FIGURE 4: Translation of a configuration space object generated by a moving workspace object. (a) Object at 0° ; (b) positive translation of the configuration space object (object at 90°); (c) negative translation of the configuration space object (object at -90°).



FIGURE 5: Merging of configuration space objects caused by moving workspace objects. (a) No configuration space object; (b) appearance of configuration space objects; (c) merging of configuration space objects by movement in the workspace.



FIGURE 6: Expansion and contraction of configuration space objects by the movement of the robot base. (a) Robot placed at (0, 0); (b) robot moves to (24, 23); (c) robot moves to (-21, -23).



FIGURE 7: Neighbor node and sensing node. (a) Neighbor node for neighbor radius r_{η} ; (b) sensing nodes around node \mathbf{q}_i for sensing radius r_s .

complicate the motion planning problem in dynamic environments.

2.3. Processing Time for Modified Configuration Space. Another important problem that complicates motion planning in dynamic environments is the processing time required for the reconstruction of a modified configuration space [28, 29]. Three-dimensional sensors in the workspace can detect free ranges that are not occupied by obstacles. However, in the configuration space of articulated robots, there are no methods of detecting free ranges that are a set of collision-free motions. A motion of the robot in the workspace is expressed as a point in the configuration space; a collision test result for a robot motion in the workspace provides only the binary information of whether collision occurs or not for a point of the configuration space. Thus, the entire free configuration space can be constructed by the mapping of the binary collision test results for all configuration space points. However, this mapping is impossible in actual problems; hence, the free configuration space is represented with abstracted graphs that consist of nodes and edges.

Most robot motion planning algorithms have been derived from sampling-based algorithms, such as PRM and RRT; these methods describe the complex configuration space with geometric graphs through the construction of nodes and edges. In the PRM method, after a learning phase in which a graph for the entire configuration space is constructed, the optimal path between a given starting point and a goal point can be obtained from the learned graph. In contrast, the RRT method identifies a reachable path by expanding the graph using a tree-type approach. In the PRM method, the obtained path is the optimal path based on the previously constructed graph; the PRM method cannot be applied in dynamic environments because the graph must be reconstructed if the configuration space changes. Because the RRT method constructs a new graph for every pathfinding query, it can be applied in changeable environments. However, the path obtained by the RRT method is not optimal



FIGURE 8: Expansive repulsion factor between two nodes in a neighbor area.

and requires considerable time for constructing the new graph. In addition, sampling-based algorithms perform tests for the construction of nodes and edges; particularly, tests for edge construction require significantly more time because collisions for line segments are determined by tests performed for numerous points on the edge. With the purpose of resolving these problems, a faster edge-test algorithm that uses the upper bound of robot motion has been proposed [30, 31]. Recently, with the development of parallel computing technologies [32–34], some approaches use GPU technology to improve processing time of robot collision checking [35–37]. Nevertheless, the reconstruction of updated graphs in real time to represent changes in dynamic environments remains a difficult problem.

3. Adaptive Roadmap Algorithm

In this study, we attempt to solve the problem of motion planning in dynamic environments. For this purpose, we propose an adaptation algorithm of geometric graphs, called



FIGURE 9: Sensory repulsion factors generated by sensing nodes.

the *adaptive roadmap algorithm*, for collision-free motion planning in a deformable configuration space. The motion planning algorithm uses graphical information previously constructed by methods, such as PRM, to rapidly determine the optimal path from the learned roadmap. However, the learning or constructing process of the roadmap consumes much time; thus, it has been impossible to apply PRM algorithms to the motion planning problem in dynamic environments.

The proposed adaptive roadmap algorithm performs the adaptation process by changing the structure of the graph according to the deformation of the configuration space. To adjust the roadmap to changeable environments, each node in the graph moves automatically to the free configuration space. Then, the node updates its edge information with respect to the changed node information. These processes are distributed and performed in parallel for all nodes. Hence, this algorithm can adapt the roadmap to the change of the configuration space by changing the graph structure. In the proposed method, some new concepts, such as the neighbor node, neighbor radius, and sensing node, are introduced.

3.1. Definitions. Definitions used are as follows:

- (i) *Node* (**q**_i): a sampled point in the *n*-dimensional free configuration space **Q**_{free} as a vertex of the graph;
- (ii) Edge (e_{ij}): a line segment that connects nodes q_i and q_j in the graph; the connection between nodes q_i and q_j is established when a route from q_i to q_j exists through a local planner or a navigation control function in the configuration space;
- (iii) Roadmap (G): an abstract geometric graph that represents the complex shape and structure of the free configuration space; a roadmap consists of a set of

nodes V and a set of edges E, where N is the number of nodes of the graph:

$$\mathbf{G} = \{ \mathbf{V}, \mathbf{E} \mid \mathbf{V} = \{ \mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_N \}, \mathbf{E}$$
$$= \{ \mathbf{e}_{12}, \mathbf{e}_{15}, \dots, \mathbf{e}_{Nk} \} \};$$
(4)

(iv) *Neighbor node*: a node located near node \mathbf{q}_i , within a distance r_{η} ; neighbor nodes can interact with node \mathbf{q}_i , and the set of neighbor nodes η_i can be represented as follows:

$$\boldsymbol{\eta}_{i} = \left\{ \boldsymbol{q}_{j} \in \boldsymbol{\mathrm{V}} \mid \left\| \boldsymbol{q}_{j} - \boldsymbol{q}_{i} \right\|_{2} \leq r_{\eta}, \ j \neq i \right\};$$
(5)

(v) Sensing node $(\Delta \mathbf{q}_j \in \mathbf{R}^n)$: a node that is sampled by Poisson-disk sampling [38] on $\partial \mathbf{B}_{r_s} = \{\mathbf{x} \in \mathbf{R}^n \mid \|\mathbf{x}\|_2 = r_s\}$, which is the surface of an *n*-dimensional hypersphere of radius r_s ; sensing nodes detect the deformation of the configuration space around each node; the set of sensing nodes **S** can be represented as follows:

$$\mathbf{S} = \left\{ \Delta \mathbf{x}_{l} \in \partial \mathbf{B}_{r_{s}} \mid l = 1, \dots, n_{s}, \left\| \Delta \mathbf{x}_{j} - \Delta \mathbf{x}_{k} \right\|_{2} \\ > r_{p}, \ \forall j, k \in \{1, \dots, n_{s}\} \right\},$$
(6)

where r_p and n_s denote the Poisson-disk sampling radius and the number of sensing nodes, respectively.

Figure 7 illustrates the concepts of neighbor node and sensing node for a node \mathbf{q}_i .

3.2. Strategies for Adaptation of Roadmap. The roadmap method is an abstract representation of the complex topology of the configuration space, with geometric graphs that consist of nodes and edges. In the previous section, we confirmed that a radical deformation of the configuration space can appear in dynamic environments. When changes occur in the workspace, they create various deformations of the configuration space; then, the constructed roadmap becomes useless. Thus, in this research, we propose an adaptation algorithm for roadmaps by adjusting the graph topology to the changed environments.

In the proposed adaptation algorithm, we adopted the motion dynamics of gases [39]. Gases are distributed evenly in the entire space of a container, regardless of its shape, by physical phenomena, such as diffusion and internal interactions, and collisions with the surface of the space boundary. The adaptation mechanism of the proposed algorithm is based on two repulsion factors: the expansive repulsion factor and the sensory repulsion factor. The expansive repulsion factor is a repulsive interaction between nodes in the roadmap. This factor can diffuse the distribution of nodes throughout the configuration space. The sensory repulsion factor represents the repulsion from sensing nodes around a node; these sensing nodes can detect the deformation of the configuration space. Because of the repulsion from the sensing nodes, nodes can move naturally toward the free area of the configuration space.



FIGURE 10: Interaction factors in a node. (a) Expansive repulsion from neighbor nodes; (b) sensory repulsion from sensing nodes that detect collisions; (c) total internal interaction factors; (d) movement of a node.

3.2.1. Expansive Repulsion Factor. The expansive repulsion factor expresses an interaction between nodes that can repel other nodes within the effective close area. Nodes in the effective close area are defined as neighbor nodes, and the threshold value that determines neighbor nodes is the neighbor radius r_{η} . Figure 8 shows the interactions between neighbor nodes in the neighbor area, which can be represented by the soft sphere model with diameter r_{η} .

If the distance r between two nodes is smaller than the neighbor radius r_{η} , expansive repulsion occurs. The strength of the interaction between the two nodes is determined by the repulsive potential function, which increases with decreasing node distance r. If the distance r is larger than r_{η} , the repulsive potential function has a very small value that cannot affect other nodes. In this study, we adopted the potential function proposed by Khatib [40]:

$$f_{\rm rep}\left(r, r_{\eta}\right) = \begin{cases} \frac{1}{2} \left(\frac{1}{r} - \frac{1}{r_{\eta}}\right)^2, & \left(0 < r \le r_{\eta}\right), \\ 0, & (\text{otherwise}). \end{cases}$$
(7)

The expansive repulsion factor results from all such interactions between neighbor nodes. Thus, using the repulsive potential function $f_{rep}(r, r_{\eta})$, the expansive repulsion factor for node \mathbf{q}_i can be represented as follows:

$$\mathbf{F}_{\text{expansive}}^{i} = \sum_{j} f_{\text{rep}} \left(r_{ij}, r_{\eta} \right) \cdot \frac{\mathbf{q}_{i} - \mathbf{q}_{j}}{r_{ij}}, \qquad (8)$$
$$\left(r_{ij} = \left\| \mathbf{q}_{i} - \mathbf{q}_{j} \right\|_{2}, \ \mathbf{q}_{j} \in \boldsymbol{\eta}_{i} \right),$$

where r_{ij} and η_i are the distance of nodes *i* and *j* and the set of neighbor nodes, respectively.

3.2.2. Sensory Repulsion Factor. Sensory repulsion represents the repulsive interactions from sensing node $\Delta \mathbf{q}$, which are distributed evenly around a node and can detect the deformation of the configuration space. Because of the repulsion from sensing nodes, nodes can move naturally toward the free area of the configuration space. Three-dimensional sensors in the workspace can detect free ranges that are not occupied by obstacles. However, in the configuration space for articulated robots, there are no methods of detecting the free range. A motion of the robot in the workspace is expressed as a point in the configuration space; a collision test result for a robot motion in the workspace gives only the binary information of whether collision occurs or not



FIGURE 11: An example of the adaptive roadmap.

for a point in the configuration space. Thus, the data from the sensing nodes produce a binary-type random vector. Therefore, the sensory repulsive factor is similar to the bangbang controller. Figure 9 shows the sensory repulsion factor generated by the sensing nodes, where $s(\mathbf{q})$ is the collision detection function in the workspace. The result of $s(\mathbf{q})$ is simple binary information; however, complex procedures are required to obtain this result, such as kinematic analysis in the workspace, generation of a collision check model, and collision testing with the environment model:

$$s(\mathbf{q}) = \begin{cases} 1, & \text{(collision detected)}, \\ 0, & \text{(otherwise)}. \end{cases}$$
(9)

The sensing nodes generate repulsion factors in the opposition direction for node \mathbf{q}_i . The sensory repulsion factor results from all such repulsive interactions of the sensing nodes. Thus, it can be represented as follows:

$$\mathbf{F}_{\text{sensory}}^{i} = -\sum_{j=1}^{n_{\text{S}}} \left(s \left(\mathbf{q}_{i} + \Delta \mathbf{q}_{j} \right) \cdot \Delta \mathbf{q}_{j} \right) \quad \left(\Delta \mathbf{q}_{j} \in \mathbf{S} \right), \quad (10)$$

where **S** is the set of sensing nodes.

3.2.3. Difference Equation for Adaptive Roadmap Algorithm. The adaptation mechanism of the proposed adaptive roadmap algorithm is based on two repulsion factors: the expansive repulsion factor $F_{expansive}$ and the sensory repulsion factor $F_{sensory}$. Figure 10 shows the overall interactions of the proposed algorithm for a node. In Figure 10(a), the expansive repulsion is illustrated as interactions with neighbor nodes. Figure 10(b) shows the sensory repulsion by detecting collision states in the configuration space. The red color of the sensing node indicates detection of a collision in the workspace. Figure 10(c) displays the total repulsion factors applied to the node, and Figure 10(d) shows the movement of the node caused by various interactions. Thus, based on this adaptation mechanism, the difference equation of node \mathbf{q}_i can be expressed as follows:

$$\mathbf{q}_{i} [k+1] = \mathbf{q}_{i} [k] + G_{1} \cdot \mathbf{F}_{\text{expansive}}^{i} [k] + G_{2}$$

$$\cdot \mathbf{F}_{\text{sensory}}^{i} [k], \qquad (11)$$

where G_1 and G_2 are the gain of the expansive and sensory repulsion factor, respectively, and

$$\mathbf{F}_{\text{expansive}}^{i}\left[k\right] = \sum_{\mathbf{q}_{j}\in\boldsymbol{\eta}_{i}} \frac{f_{\text{rep}}\left(r_{ij}\left[k\right], r_{\eta}\right)}{r_{ij}\left[k\right]} \cdot \left(\mathbf{q}_{i}\left[k\right] - \mathbf{q}_{j}\left[k\right]\right), \\ \left(r_{ij}\left[k\right] = \left\|\mathbf{q}_{i}\left[k\right] - \mathbf{q}_{j}\left[k\right]\right\|_{2}\right),$$

$$f_{\text{rep}}\left(r, r_{\eta}\right) = \begin{cases} \frac{1}{2} \left(\frac{1}{r} - \frac{1}{r_{\eta}}\right)^{2}, & \left(0 < r \le r_{\eta}\right), \\ 0, & (\text{otherwise}), \end{cases}$$
$$\mathbf{F}_{\text{sensory}}^{i}\left[k\right] = -\sum_{\Delta \mathbf{q}_{j} \in \mathbf{S}} \left(s \left(\mathbf{q}_{i}\left[k\right] + \Delta \mathbf{q}_{j}\right) \cdot \Delta \mathbf{q}_{j}\right), \\ s\left(\mathbf{q}\right) = \begin{cases} 1, & (\text{collision detected}), \\ 0, & (\text{otherwise}). \end{cases}$$

(12)

Figure 11 shows an example of the total adaptive roadmap, which consists of 12 nodes, and each node has eight sensing nodes. The adaptive roadmap algorithm performs the entire adaptation using distributed processing of these complex procedures in each node. Thus, the computation of (11) should be performed in parallel at each node. Figure 12 shows the flowchart for processing the adaptive roadmap algorithm. In the flowchart, four procedures in the blue boxes can be performed dispersively at each node by parallel computing techniques, such as multithreading or GPU processing.

4. Simulation Results

4.1. Adaptation Results for Workspace Change. To verify the feasibility of the proposed adaptive roadmap method, simulations were performed in a simplified test environment. If the robot has three or more DOFs, the free configuration space becomes a complex high-dimensional space, which is difficult to visualize and identify. Thus, we considered a test environment with a two-link robot with a two-dimensional configuration space. In the test simulations, we set the number of nodes N to 100 for constructing the roadmap, and the gain of the expansive and sensory repulsion factor were set to 40 and 1.0, respectively. To perform the adaptation at each node, distributed computing with the multithreading technique was implemented, and 100 threads were allocated for the processing of each node.

Each joint of the robot was set to have an operating limit range of $-170^{\circ}-170^{\circ}$; thus, the configuration space had the same topology as the Euclidean space. If the two-link robot has no operating limit, the configuration space is a torus-type space; in the case of higher dimensions, the configuration space becomes a more complex space that is difficult to handle. Thus, we assumed that the robot had operating limits to simplify the configuration space, because most actual robots have joint limits.

Figure 13 shows the test environments for the simulations. Figure 13(a) shows the workspace conditions, a two-link robot, and five movable obstacles. Figure 13(b) shows the configuration space with respect to the workspace displayed in Figure 13(a). Obstacles in the configuration space are illustrated with the same colors as in the workspace.

Figure 14 shows the processing procedures of the adaptive roadmap algorithm. Figure 14(a) displays the initial state of the roadmap, in which 100 nodes were scattered randomly in the configuration space. Figures 14(b) and 14(c) illustrate



FIGURE 12: Flowchart for processing the adaptive roadmap algorithm.

the states of the adaptive roadmap after the 5th and 10th iterations, respectively, in which the nodes disperse in the free configuration space. Figure 14(d) shows the state of the adaptive roadmap at equilibrium, after 100 iterations; at this state, the nodes are distributed evenly in most of the free configuration space. Thus, the coverage, designated by the yellow area, of the distribution of the nodes is almost complete, and the edges are well connected throughout the free configuration space. These processing procedures confirmed that the adaptive roadmap can change the structure of a random initial state to adapt to the changed configuration space.

Figure 15 shows the motion planning results for two test cases using the roadmap information after performing the adaptation. Figures 15(a) and 15(c) represent the roadmap graphs and the planned paths of the robot motion in the configuration space. Figures 15(b) and 15(d) show the operational trajectories of robot motion in the workspace according to the planned paths. From the motion planning results of Figure 15, it can be confirmed that the collision-free robot motions were



FIGURE 13: Test workspace and the corresponding configuration space. (a) Two-link robot and obstacles in workspace; (b) configuration space for the workspace in (a).



FIGURE 14: Running procedures of the adaptive roadmap algorithm. (a) Initial state; (b) state of the roadmap after the 5th iteration; (c) state of the roadmap after the 10th iteration; (d) state of the roadmap after the 100th iteration.

obtained easily using the adapted roadmap graph, and the planned motion paths were similar to the optimal path.

Next, we tested the behavior of the adaptive roadmap algorithm for changes of the workspace. Figure 16 shows the modified test workspace and the corresponding configuration space. In the changed workspace of Figure 16(a), the positions of five obstacles were changed, and the configuration space of Figure 16(b) was largely deformed, accordingly. Figure 17 shows the processing procedures of the adaptive roadmap algorithm for the changes of the workspace. Figure 17(a) displays the initial state of the roadmap. Figures 17(b) and 17(c) show the states of the adaptive roadmap after the 5th and 10th iterations, respectively, in which the nodes disperse in the changed configuration space. Figure 17(d) shows the state of the adaptive roadmap at equilibrium, after 100 iterations. In this state, the coverage with respect to the



FIGURE 15: Motion planning results and trajectories of robot motion for two test cases (start motion (green); goal motion (red)). (a) Roadmap graph and the planned motion (blue) in the configuration space for the collision-free path from $[155^{\circ}, 78^{\circ}]$ to $[-85^{\circ}, -39^{\circ}]$; (b) motion trajectory in the workspace for (a); (c) roadmap graph and the planned motion (blue) in the configuration space for the collision-free path from $[-9^{\circ}, 10^{\circ}]$ to $[-160^{\circ}, -33^{\circ}]$; (d) motion trajectory in the workspace for (c).



FIGURE 16: Changed workspace and the corresponding configuration space. (a) Two-link robot and obstacles in the workspace; (b) configuration space of the workspace shown in (a).

distribution of nodes is almost complete, and the edges are well connected throughout the changed configuration space. The results of this simulation confirmed that the adaptive roadmap can perform a successful adaptation in the case of a radical deformation of the configuration space. Figure 18 shows the motion planning results in the changed workspace, shown in Figure 16, for two test cases using the adapted roadmap information. Figures 18(a) and 18(c) represent the roadmap graphs and the planned paths of robot motion in the configuration space. Figures 18(b) and



FIGURE 17: Adaptation procedures for the changed configuration space. (a) Initial state; (b) state of the roadmap after the 5th iteration; (c) state of the roadmap after the 10th iteration; (d) state of the roadmap after the 10th iteration.

18(d) show the operational trajectories of robot motion in the workspace according to the planned paths. Similar to the results of Figure 15, the paths of robot motion for the two test cases were obtained for the changed workspace, and the acquired motion paths seem to be close to the optimal path.

4.2. Adaptation Results for Various Deformations. Subsequently, the behavior of the adaptive roadmap was observed in dynamic environments. To generate continuous changes of the workspace, the positions of all obstacles were modified slightly and randomly for every simulation step, and considerable obstacle position changes were applied at every 50th step.

Figure 19 shows the results of the adaptive roadmap under dynamic environment conditions. Figure 19(a) presents the results for the total expansive repulsion. The red line shows the free configuration ratio, and the green line shows the total expansive repulsion. When a radical change occurs, the total expansive repulsion exhibits a spiky transient pattern; however, it returns to a stable state after some iterations. The gray line in Figure 19(b) represents the coverage for the configuration space. When a radical change occurs, the coverage briefly decreases; however, it stays near 100% throughout the running of the adaptive roadmap algorithm.

Figure 20 shows the steady state of the adaptive roadmap for various configuration space conditions. The free

configuration space ratio ranged from 89.6% to 38.7%, which confirmed that the adaptive roadmap converged to equilibrium for various environments. Further, we have published a video of the continuous behavior of the adaptive roadmap algorithm. The detailed adaptation process of the roadmap graph can be seen at [41].

5. Conclusions

In this paper, we propose an algorithm that can adapt to a radically changeable configuration space. For this purpose, we first analyzed the deformation of the configuration space with respect to the change of the workspace. To address the issue of the deformation of the configuration space, we presented an adaptation algorithm for the roadmap graph based on distributed processing. In this algorithm, each node in the graph moves in the free configuration space by detecting the deformation of the configuration space; then, the node updates its edge information with respect to the changed node information. The adaptation mechanism of the proposed algorithm is based on two basic repulsion factors: the expansive repulsion factor and the sensory repulsion factor. Moreover, the adaptive roadmap can perform the entire adaptation by distributed processing of the complex procedure in each node. To verify the effectiveness of the proposed method, simulations of a continuously changing



FIGURE 18: Motion planning results and trajectories of robot motion for two test cases (start motion (green); goal motion (red)). (a) Roadmap graph and the planned motion (blue) in the configuration space for the collision-free path from $[-155^\circ, 35^\circ]$ to $[37^\circ, 157^\circ]$; (b) motion trajectory in the workspace for (a); (c) roadmap graph and the planned motion (blue) in the configuration space for the collision-free path from $[-4^\circ, 59^\circ]$ to $[-165^\circ, 62^\circ]$; (d) motion trajectory in the workspace for (c).



FIGURE 19: Simulation results of the adaptive roadmap for various conditions of the configuration space volume (red). (a) Total expansive repulsion (green); (b) coverage of the roadmap (gray).

workspace were performed in a test environment of a twolink manipulator case. The simulation results confirmed that the graph could perform the entire adaptation with respect to the change of the configuration space using the proposed algorithm. As future work, high-dimensional configuration space and complex problems, such as redundancy and mobile manipulation for actual robots, will be considered. Thus, to overcome these problems, we should deliberately utilize massively parallel processing technology using GPUs.



FIGURE 20: Topologies of the adaptive roadmap at equilibrium for various conditions of the free configuration space ratio. (a) 89.6%; (b) 69.3%; (c) 79.5%; (d) 59.9%; (e) 59.3%; (f) 54.1%; (g) 57.8%; (h) 48.68%; (i) 38.7%.

Moreover, the application of the current results to the study of complex systems will require further investigation. The adaptive roadmap algorithm is strongly related to the complex system that interacts with many independent agents internally. Therefore, the further development of the adaptive algorithm by combining it with complex systems science should be considered as important future work.

Conflict of Interests

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

Acknowledgments

This research was financially supported by the Ministry of Trade, Industry and Energy (MOTIE) and Korea Institute for

Advancement of Technology (KIAT) through the Promoting Regional Specialized Industry.

References

- [1] J. C. Latombe, Robot Motion Planning, Kluwer Academic, 1991.
- [2] H. Choset, K. M. Lynch, S. Hutchison et al., Principles of Robot Motion: Theory, Algorithms and Implementations, The MIT Press, 2005.
- [3] J. T. Schwartz and M. Sharir, "On the piano movers' problem. III. Coordinating the motion of several independent bodies: the special case of circular bodies moving amidst polygonal barriers," *International Journal of Robotics Research*, vol. 2, no. 3, pp. 46–75, 1983.
- [4] C. Park, J. Pan, and D. Manocha, "Real-time optimizationbased planning in dynamic environments using GPUs," in *Proceedings of the IEEE International Conference on Robotics* and Automation (ICRA '13), pp. 4090–4097, IEEE, Karlsruhe, Germany, May 2013.
- [5] T. Kunz, U. Reiser, M. Stilman, and A. Verl, "Real-time path planning for a robot arm in changing environments," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '10)*, pp. 5906–5911, Taipei, Taiwan, October 2010.
- [6] L. E. Kavraki, P. Švestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.
- [7] S. M. LaValle and J. J. Kuffner Jr., "Randomized kinodynamic planning," *International Journal of Robotics Research*, vol. 20, no. 5, pp. 378–400, 2001.
- [8] L. E. Kavraki, J. C. Latombe, R. Motwani, and P. Raghavan, "Randomized query processing in robot path planning," in *Proceedings of the 27th Annual ACM Symposium on Theory of Computing (STOC '95)*, pp. 353–362, Las Vegas, Nev, USA, June 1995.
- [9] J. Pan, L. Zhang, and D. Manocha, "Collision-free and smooth trajectory computation in cluttered environments," *International Journal of Robotics Research*, vol. 31, no. 10, pp. 1155–1175, 2012.
- [10] J. Vannoy and J. Xiao, "Real-time adaptive motion planning (RAMP) of mobile manipulators in dynamic environments with unforeseen changes," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1199–1212, 2008.
- [11] M. S. Branicky, M. M. Curtiss, J. Levine, and S. Morgan, "Sampling-based planning, control and verification of hybrid systems," *IEE Proceedings: Control Theory and Applications*, vol. 153, no. 5, pp. 575–590, 2006.
- [12] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.
- [13] A. M. Ladd and L. E. Kavraki, "Measure theoretic analysis of probabilistic path planning," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 229–242, 2004.
- [14] K. Hauser, "On responsiveness, safety, and completeness in realtime motion planning," *Autonomous Robots*, vol. 32, no. 1, pp. 35–48, 2012.
- [15] F. Belkhouche, "Reactive path planning in a dynamic environment," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 902–911, 2009.

- [16] S. Petti and T. Fairchard, "Safe motion planning in dynamic environments," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2210–2215, Edmonton, Canada, August 2005.
- [17] W. Lu, G. Zhang, and S. Ferrari, "An information potential approach to integrated sensor path planning and control," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 919–934, 2014.
- [18] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.
- [19] F. Cucker and S. Smale, "Emergent behavior in flocks," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 852–862, 2007.
- [20] W. Ren and Y. Cao, Distributed Coordination of Multi-agent Networks: Communications and Control Engineering Series, Springer, Berlin, Germany, 2011.
- [21] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [22] J. Wang, Z. Liu, and X. Hu, "Consensus control design for multiagent systems using relative output feedback," *Journal of Systems Science and Complexity*, vol. 27, no. 2, pp. 237–251, 2014.
- [23] J. Hu and Y. S. Lin, "Consensus control for multi-agent systems with double-integrator dynamics and time delays," *IET Control Theory & Applications*, vol. 4, no. 1, pp. 109–118, 2010.
- [24] G. Xie and L. Wang, "Consensus control for a class of networks of dynamic agents," *International Journal of Robust and Nonlin*ear Control, vol. 17, no. 10, pp. 941–959, 2007.
- [25] L. Xiao, S. Boyd, and S.-J. Kim, "Distributed average consensus with least-mean-square deviation," *Journal of Parallel and Distributed Computing*, vol. 67, no. 1, pp. 33–46, 2007.
- [26] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: a mathematical theory of network architectures," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, 2007.
- [27] F. Zhang, J. Jia, and L. Yang, "Adaptive synchronization in complex network with different order node dynamics," *Discrete Dynamics in Nature and Society*, vol. 2015, Article ID 919146, 7 pages, 2015.
- [28] J. Schulman, Y. Duan, J. Ho et al., "Motion planning with sequential convex optimization and convex collision checking," *International Journal of Robotics Research*, vol. 33, no. 9, pp. 1251–1270, 2014.
- [29] J. Pan, I. A. Sucan, S. Chitta, and D. Manocha, "Real-time collision detection and distance computation on point cloud sensor data," in *Proceedings of the IEEE International Conference* on Robotics and Automation (ICRA '13), pp. 3592–3599, IEEE, Karlsruhe, Germany, May 2013.
- [30] F. Schwarzer, M. Saha, and J.-C. Latombe, "Adaptive dynamic collision checking for single and multiple articulated robots in complex environments," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 338–353, 2005.
- [31] F. Schwarzer, M. Saha, and J. C. Latombe, "Exact collision checking of robot paths," in *Algorithmic Foundations of Robotics V.*, Springer, Berlin, Germany, 2004.
- [32] M. Damrudi and N. Ithnin, "An optimization of tree topology based parallel cryptography," *Mathematical Problems in Engineering*, vol. 2012, Article ID 871091, 10 pages, 2012.
- [33] S. K. Samudrala, J. Zola, S. Aluru, and B. Ganapathysubramanian, "Parallel framework for dimensionality reduction of large-scale datasets," *Scientific Programming*, vol. 2015, Article ID 180214, 12 pages, 2015.
- [34] B. He, L. Tang, J. Xie, X. Wang, and A. Song, "Parallel numerical simulations of three-dimensional electromagnetic radiation with MPI-CUDA paradigms," *Mathematical Problems in Engineering*, vol. 2015, Article ID 823426, 9 pages, 2015.
- [35] C. Lauterbach, Q. Mo, and D. Manocha, "gProximity: hierarchical GPU-based operations for collision and distance queries," *Computer Graphics Forum*, vol. 29, no. 2, pp. 419–428, 2010.
- [36] J. Pan and D. Manocha, "GPU-based parallel collision detection for fast motion planning," *International Journal of Robotics Research*, vol. 31, no. 2, pp. 187–200, 2012.
- [37] D. Kim, J.-P. Heo, J. Huh, J. Kim, and S.-E. Yoon, "HPCCD: hybrid parallel continuous collision detection using CPUs and GPUs," *Computer Graphics Forum*, vol. 28, no. 7, pp. 1791–1800, 2009.
- [38] M. S. Ebeida, S. A. Mitchell, A. Patney, A. A. Davidson, and J. D. Owens, "A simple algorithm for maximal poisson-disk sampling in high dimensions," *Computer Graphics Forum*, vol. 31, no. 2, pp. 785–794, 2012.
- [39] E. A. Jackson, *Equilibrium Statistical Mechanics*, Prentice-Hall, Upper Saddle River, NJ, USA, 1968.
- [40] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986.
- [41] YouTube, Adaptive roadmap algorithm, November 2015, https://www.youtube.com/watch?v=aanjEEkZ6n8.

Research Article Heat Transfer Analysis and Modification of Thermal Probe for Gas-Solid Measurement

Hong Zhang and Xiangying Qi

National Key Laboratory of Science and Technology on Multispectral Information Processing, School of Automation, Huazhong University of Science and Technology, Wuhan 43004, China

Correspondence should be addressed to Hong Zhang; sunracer@mail.hust.edu.cn

Received 22 November 2015; Accepted 5 January 2016

Academic Editor: Yuan Fan

Copyright © 2016 H. Zhang and X. Qi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The presented work aims to measure the gas-solid two-phase mass flow-rate in pneumatic conveyor, and a novel modified thermal probe is applied. A new analysis of the local heat transfer coefficients of thermal probe is presented, while traditional investigations focus on global coefficients. Thermal simulations are performed in Fluent 6.2 and temperature distributions of the probe are presented. The results indicate that the probe has obviously stable and unstable heat transfer areas. Based on understanding of probe characteristics, a modified probe structure is designed, which makes the probe output signal more stable and widens the measuring range. The experiments are carried out in a special designed laboratory scale pneumatic conveyor, and the modified probe shows an unambiguous improvement of the performance compared with the traditional one.

1. Introduction

It is usual that the solid powder is transported in industrial plant by pneumatic conveyor. The particularly important examples are the transport of granular and pulverized coal for boiler and blast furnace. In those processes, on-line measurement of mass flow-rate in gas-solid two-phase flow is a common problem that has never been solved successfully [1].

Lots of methods were used to solve this problem, including static electricity, microwave, tomography, and pressure drop. Nowadays, the thermal method is also applied to the gas-solid measurement. As a long time used sensor which measures gas mass flow in industry, the thermal probe is extended to measure the gas-solid two-phase flow. The first use of thermal probe to measure gas-solid flow should be owed to Moriyama et al. [2] since 1985. The authors developed a noninvasive differential temperature sensing method for the mass flow rate 0–1,000 kg/h and mass flow ratio 0–47.2 kg/kg. The researchers including Zheng et al. [3, 4], Liu et al. [5], Yuan and Lu [6], and Zhang and Li [7] have focused on this area recently.

Our previous works include investigations on traditional heat balance method, thermal probe method [7, 8], and data

fusion to achieve better performance [9]. By applying the thermal probe theory to measure the mass flow-rate of gassolid two-phase flow, the variety of solids loads in pneumatic conveying system, but two shortages are observed including signal fluctuation and severe on-linearity causing narrow measurement range.

Commonly, the thermal probe has fast response time and high sensitivity and the issues of worse zero stability and signal fluctuation. To apply the thermal probe to detect the gas-solid flow, the heat transfer characteristics research in detail is needed to solve the problems of stability and range extension. Theoretical analysis and numerical simulation help us to study and find new solution to improve the probe performance in gas-solid.

While the details of heat transfer behavior in gas-solid flow are not yet well understood, the early experimental results have already supported the thermal probe's application in solids mass flow-rate measurement [10–12]. Experimental investigations have proved that the wall Nusselt number is a function of the solid mass loading ratio. The Nusselt number will significantly increase in higher solids loading ratios because the heat transfer is enhanced caused by gas boundary layer influence due to solids. By using a four-way coupling Eulerian-Lagrangian model to simulate nonisothermal gas-solid flow in a vertical pipe, El-Behery et al. pointed out that the presence of solid particles reduces the boundary layer thickness and this reduction increases as the mass loading ratio increases [13]. With the recent improvement of the related mathematical models, the computational fluid dynamics (CFD) tools are strong enough to describe the heat transfer and dynamics behavior of gas-solid two-phase flow. The commercial software such as Fluent has been validated in many situations [14]. Fluent 6.2 was chosen as a simulation tool in analyzing the heat transfer of the thermal probe in the presented work.

2. Analysis of Heat Transfer Equation of Thermal Probe in Single-Phase Flow

For a thin heated cylinder placed in fluid (namely, hot wire), the way of heat transfer between cylinder and fluid is mainly performed as forced convection. King concluded an empirical formula of convective heat transfer between "infinite" cylinder and fluid through experimental and theoretical studies in 1914:

$$Nu = A + BRe^{1/2}.$$
 (1)

In (1), *A* and *B* are constant. With regard to the specific fluid within identified temperature and pressure, fluid density ρ , cylinder diameter *d*, and fluid viscosity μ can be regarded as constant and integrated into *A* and *B*. According to the definition of Nusselt number and the relationship between Nusselt number Nu and fluid flow *U* of convective heat transfer, heat transfer coefficient of cylinder *h* can be obtained as

$$h = \frac{k}{d} \left(A + BU^{1/n} \right), \tag{2}$$

where k is the fluid heat conductivity and d is the cylinder diameter.

While the hot wire conducts convective heat transfer with fluid, it is heated by an external electrical source. Then the hot wire heat transfer equation derived by King's equations can be expressed as

$$\frac{I^2 R_w}{R_w - R_f} = A + B U^{1/2},$$
(3)

where R_w is the resistance of hot wire when it is at a certain temperature T_w and R_f is the resistance of hot wire when it is at fluid temperature. The resistances of hot wire R_w and R_f are functions of temperature, and $I^2 R_w$ is heating power in the thermal equilibrium state; moreover, $(R_w - R_f)$ has a linear relationship with the temperature difference and is characterized as the temperature difference between hot wire and fluid. *A* and *B* can be determined by calibration.

Considering that (1) is based on an infinite hot wire, the rate of length and diameter of hot wire is large enough (the rate of length and diameter of typical hot wire is more than 300, which are, resp., 1.25 mm and 4μ m). If cylinder length/diameter rate is taken into account, the index *n* on velocity *U*, which is available for calibration, should be

changed. Therefore, for the cylinder with finite-length, (3) can be rewritten as

$$\frac{I^2 R_w}{R_w - R_f} = A + BU^n.$$
(4)

Notably, *A* and *B* here contain fluid parameters such as fluid density, viscosity, thermal conductivity, and specific heat, and they are assumed to be constant during the measurement. However, these parameters may vary along with the state of temperature, fluid component, and so on. The related investigations belong to temperature and components compensation of thermal probe and are not discussed in this paper.

2.1. Fluid Density Is Not a Constant. As the range of temperature and pressure is rather large, fluid density is bound to fluctuate. The following heat transfer equation of thermal probe reveals that heat transfer of thermal probe is related with local mass flow of fluid q_M , so the fluid density can be expressed in the output of thermal probe and then thermal probe can be used to directly measure the mass flow of fluid:

$$\frac{I^2 R_w}{R_w - R_f} = A + B \left(q_M\right)^n.$$
⁽⁵⁾

This equation is called heat transfer equations of thermal mass flow meter. In other words, the local mass flow flowing around thermal probe has a determined nonlinear relationship with temperature of speed probe (which is characterized by R_w), temperature of fluid (which is characterized by R_f), and the heating current. The local mass flow of thermal probe can be obtained when the heat transfer probe works in constant power mode or constant temperature differential mode.

2.2. Prandtl Number Is Not a Constant. At high Reynolds number, Prandtl number should not simply be seen as a constant in more general equations of fluid heat transfer because there was a certain degree of error in (1). Researches at home and abroad on heat transfer criteria equation Nu = f(Re, Pr) of flowing around cylinder show that a more precise heat transfer expression is needed in a larger measurement range.

In different flow states, different heat transfer criteria equations can be used to analyze coefficients A and B. Here, criteria equation obtained from Kramers experiment is adopted and can be described as

$$Nu = 0.42Pr^{0.2} + 0.57Pr^{0.33}Re^{0.50}.$$
 (6)

It is appropriate at 0.01 < Re < 10000 and 0.71 < Pr < 1000 range.

Meanwhile, constants A and B in (4) should be

$$A = 0.42 \frac{\pi k l}{\alpha_0 R_0} \left(\frac{c_p \mu}{k}\right)^{0.2},$$

$$B = 0.57 \frac{\pi k l}{\alpha_0 R_0} \left(\frac{c_p \mu}{k}\right)^{0.33} \left(\frac{d}{\mu}\right)^n.$$
(7)

Mathematical Problems in Engineering

Among them, the temperature coefficient α_0 of heating resistance, the resistance R_0 at the reference temperature, the length l of heat cylinder, and the diameter d are inherent constants of the probe; the thermal conductivity k of fluid, the specific heat capacity c_p , and the dynamic viscosity μ are related to the temperature. When determining the specific values of these parameters, the film temperature is applied, which is defined as the average values of the temperature of probe and the temperature of fluid; namely,

$$T_{\rm film} = \frac{\left(T_w - T_f\right)}{2}.$$
 (8)

Leaving aside the small differences of specific coefficients in expressions of *A* and *B*, due to utilizing different heat transfer criteria equations, equations could be used to qualitatively analyze and research calibration constants for thermal mass flow probe.

The heat transfer criteria equations listed above are empirical relationship obtained by considering the overall heat transfer coefficient of the probe. Actually, the distribution of the local heat transfer coefficient has a big difference. The next section will specifically discuss the local heat transfer coefficient of the probe to help to improve the design of the probe in the later chapters.

3. Analysis of Local Heat Transfer Coefficient of Thermal Probe

It can be seen from the whole heat transfer theory described above that the thermal probe measures the velocity of flow and mass flow was actually achieved by measuring the heat transfer characteristics of the thermal probe. So it is necessary to further study the local heat transfer coefficient of thermal probe. The heat transfer coefficient is closely related to the fluid mechanics behavior while fluid flows around the thermal probe.

3.1. Analysis of Flow Patterns While Fluid Flows around the Probe. When the fluid passes along a perpendicular direction to the axis of the cylinder, the fluid has characteristics with boundary layer. The fluid flowing across the probe may cause emerging inverse flow, whirlpool, and vortex beam. When the Reynolds number is very low, the boundary layer separation does not occur and the fluid flowing around cylinder appears as a status of climb stream. With the increase of the Reynolds number, a certain frequency of vortex will be started up.

The angle φ starting at the windward stagnation can be used to represent the position of the boundary layer separation point (Figure 1). When the Reynolds number 10 < $\text{Re}_D \le 1.5 \times 10^5$, the boundary layer separation point occurs in the range of $\varphi = 80^\circ \sim 85^\circ$; when $\text{Re}_D > 1.5 \times 10^5$, the boundary layer is turned into a turbulent flow before separating, and the occurrence of separation is pushed back to the point at $\varphi = 140^\circ$.

3.2. The Distribution of Local Heat Transfer Coefficient around Probe. The growth of boundary layer will inevitably lead to



FIGURE 1: Separation of the boundary layer flow around a cylinder.

changes in the local heat transfer coefficient of a cylindrical probe. The heat transfer experiment for air flowing around the cylinder conducted by Giedt [15] revealed the distribution of the local Nu_D number along the wall (Figure 2). The curves show that the local heat transfer coefficient decreases with increasing angle in the range of $\varphi = 80^{\circ} \sim 85^{\circ}$, because the laminar boundary layer is thickened continuously; the heat intensity is also weakened. At low Reynolds number, the rise of heat transfer coefficient reflects the separation of the boundary layer of fluid flowing around probe, because the disturbance of the fluid flowing off the probe strengthens the heat transfer. At low Reynolds number, the curve shows two distinct fluctuations in the region of fluid flowing off the probe, and at this time the separation of the boundary layer occurs at $\varphi = 140^{\circ}$; in the range of $\varphi \approx 80^{\circ} \sim 140^{\circ}$, the increase of heat transfer coefficient is the result of the change from the boundary layer into a turbulent flow; after $\varphi > 140^{\circ}$, the enhance of heat transfer results from the fluid flowing off the probe. From the three-dimensional perspective, as $\text{Re}_D >$ 300, the fluid flowing off the probe shows cycle distribution in the longitudinal direction of the cylinder, and the local heat transfer characteristics will become more complex.

If the configuration of thermal probe is a cylinder, the following conclusions can be drawn through the above analysis:

- (a) The heat transfer coefficient of the probe increases with the increase of Reynolds number, and the heat transfer coefficient of the probe reflects the change in flow rate.
- (b) The distribution of the local convective heat transfer coefficient on the surface of the probe in the circumferential direction fluctuates greatly.
- (c) As the cut-off point to $\varphi \approx 80^{\circ}$, the local heat transfer coefficient of the thermal probe towards flow



FIGURE 2: Giedt's local heat transfer coefficient of air flow around a cylinder.

surface performs good regularity, while it is irregular in the full range of Reynolds number in the opposite direction.

4. The Numerical Simulation of Thermal Probe around Air

Numerical simulations have been done to reveal the details of heat transfer characteristics of thermal probe in fluid.

4.1. Policy of Simulation. As a famous fluid simulation software, Fluent is chosen as the simulation tool and ICEM as the grid tool. Due to the asymmetry of the circular pipe with thermal probe, the simulation should be done under 3D model.

4.2. The Numerical Model and Grid. The thermal probe is comprised of two parts, the temperature detection part and velocity detection part, respectively. The diameter of probe defined to be 4 mm. To avoid the influence of wake flow formed by temperature detection part, a heat insulation sub in velocity detection part is designed.

A whole view of simulation pipe, whose diameter is 50 mm, is shown in Figure 3(c). The pipe wall is set to be thermally isolated. And the probe is settled in the middle of pipe. To insure the flow stableness, the pipe length is set to be 5 times as long as the pipe diameter.

The model grid generated by an unstructured grid method uses ICEM software as shown in Figure 3.

4.3. Colored Distribution of Probe Heat Transfer Coefficient. As illustrated in Figure 4, under the situation of air velocity 8.79 m/s, temperature 600 K, and temperature difference between the velocity detection part and temperature detection part, the local heat transfer coefficient distribution corresponds well with the above analysis. The windward side of the probe is highly coefficient and more stable, and also a 3D distribution along the length can be observed.

5. The Modification of Probe Design

5.1. The Meaning of Local Heat Transfer Coefficient Distribution for Design of Thermal Probe. From the measurement point of view, it is expected that the heat transfer characteristics of the thermal probe are of stability and can change monotonously with the velocity of fluid. Then the probe should be designed to take full advantage of the heat transfer characteristics towards flow surface.

Known from the foregoing analysis and simulation results, making the local heat transfer structure of the thermal probe better can improve measurement performance of probe. The basic idea is that using hot-film probe and designing the layout configuration of hot-film on the probe makes the hot-film located on the most sensitive parts of the gas-solid heat transfer and reduces the transient measuring fluctuations caused by fluid trail (dynamic characteristics are not okay).

Based on this idea, the "partition" design of thermal probe can be taken into consideration, that is, taking the flow surface of probe as the heat exchange section and the tail part of probe as the adiabatic section (Figure 5), thus expanding the measuring range of thermal probe and effectively reducing fluctuations of the signal. The bolder idea is designing the probe to be ellipse or airfoil so that the fluid will not be separated in a wide range and the boundary layer region of the laminar flow will increase.

On the other hand, as the heat transfer area becomes small, the overall heat transfer coefficient of the probe will become small. In this way, in order to obtain a wide measuring range and stable signal, it needs to sacrifice part of the measuring system sensitivity. On balance, this partition design has more advantages.

5.2. The Modification Thermal Probe. Since the partition design method can improve the performance of the sensor, modification should be done.

First of all, the velocity detection part has a heat insulation sub covered by the wake flow of the temperature detection part. Mainly considering the strength and heat insulation characteristics, it is made of ceramics (as shown in Figure 3(a)).

And then, the partition design method was used inside the probe; the intersecting surface is shown in Figure 6. Considering the heat transfer and machining operation characteristics carefully, the velocity part was divided into two areas. One is for heat transfer and the other is for heat insulation, and the boundary angle is 160°. The heat transfer area is made of stainless steel and heat insulation area is also made of ceramics. The heating part is formed by a strut with platinum resistance wire on it, which is installed in the hole of heat transfer area. This design makes the probe choose the





(c) The whole pipe

FIGURE 3: Model and grid of pipe with thermal probe. (1) Temperature detection part. (2) Heat transfer sub of velocity detection part. (3) Heat insulation sub of velocity detection part.



FIGURE 4: The colored heat transfer coefficient distribution.

stable and sensitive area of the probe cylinder. Consequently, the fluctuating output of measurement system is controlled, and then measuring range is extended.



FIGURE 5: Heat transfer configuration design of probe at the cross section.

6. The Experimental System and Results

6.1. The Experimental System. The experimental system, which is based on the pipeline of diameter 50 mm, has been designed and developed to facilitate the measurement of mass flow rate of the air in pipeline. Compressed air in pipeline is generated by a Roots blower and the speed is controlled by an invertor. According to the characteristics of the blower, the standard speeds in pipeline can be calculated and converted to the invertor frequency. The thermal probes (traditional



FIGURE 6: Intersecting surface of the probe parts.



FIGURE 7: Curves of output voltages. V_1 : old type. V_2 : new type.

type probe and new modified probe) are inserted into the certain parts of the pipe to be tested and compared. A digital data acquisition device is used to deal with the outputs of the sensors and transmit the data to an industrial PC with USB interface. The industrial PC records the data to be analyzed.

6.2. The Experimental Results. The experiments are carried out while the temperature and speed of air vary to evaluate the performance of the measurement system.

While keeping the flow velocity to a certain value (about 5.3 m/s) and the temperature varying from 20° C to 60° C, the output of the system is around 1.80 V with a little variation, as is shown in Table 1, which means that the probe has a good temperature compensation ability.

While changing the flow velocity at room temperature, it can be found that a typical nonlinear curve is fit to the theory of the thermal probe. But compared with the unmodified probe, the new curve is more flat and more linear (Table 2 and Figure 7).

TABLE 1: Result of experiment with conveying nothing.

Flow velocity (m/s)	V_1 (V)	<i>T</i> (°C)
5.28	1.75	20.4
5.30	1.73	30.0
5.32	1.79	39.8
5.27	1.78	50.3
5.30	1.79	60.2

TABLE 2: The flow velocity test with old probe and new probe V_1 .

Temperature	Flow velocity	V_1 (V)	V_2 (V)
(°C)	(m/s)	(Old type)	(New type)
21.0	0.00	0.01	0
20.7	1.76	1.46	1.3
20.4	3.51	1.66	1.55
20.3	5.27	1.78	1.63
20.2	7.03	1.85	1.78
20.2	8.79	1.90	1.9
20.2	10.54	1.95	2.1
20.4	12.30	2.00	2.2
20.5	14.06	2.02	2.6
20.5	15.81	2.04	2.8
20.8	17.57	2.05	3.7

7. Conclusions

The probe used in a gas-solid mass flow-rate measurement system is designed and the analysis of the circuits is presented. As a key part of a data fusion method, the new modified probe changes the flow and heat transfer structure. The experiment in single phase flow illustrates the improvement of the performance of probe. The probe has potential to be of benefit to the future test of gas-solid two-phase flow.

Conflict of Interests

The authors declare that they have no conflict of interests.

References

- Y. Zheng and Q. Liu, "Review of techniques for the mass flow rate measurement of pneumatically conveyed solids," *Measurement*, vol. 44, no. 4, pp. 589–604, 2011.
- [2] T. Moriyama, S. Fujii, K. Abe, and M. Kobayashi, "Mass flowmeter using heat transfer for dense phase solid gas two phase flow," in *Proceedings of Conference of the Fluid Control and Measurement*, pp. 795–800, 1985.
- [3] Y. Zheng, J. R. Pugh, D. McGlinchey, and R. O. Ansell, "Simulation and experimental study of gas-to-particle heat transfer for non-invasive mass flow measurement," *Measurement*, vol. 41, no. 4, pp. 446–454, 2008.
- [4] Y. Zheng, D. McGlinchey, J. Pugh, and Y. Li, "Experimental investigation on heat transfer mechanisms of pneumatically conveyed solids' plugs as a means to mass flow rate measurement," *Flow Measurement and Instrumentation*, vol. 40, pp. 232– 237, 2014.
- [5] L. Liu, Y. Wang, F. D. Zhou, and H. Qian, "Measurement of air flow and concentration of pulverized coal with two ellipseshaped thermal probes," *Journal of Xi'an Jiaotong University*, vol. 48, no. 1, pp. 23–26, 2008.
- [6] Z. Yuan and Z. Lu, "A new method for measuring the solid flow rate in gas-solid two phase flow," *Journal of Combustion Science* and Technology, vol. 5, no. 1, pp. 52–56, 1999.
- [7] H. Zhang and C. Li, "The experiments of the measuring method for gas-solid two-phase mass flow," *Journal of Huazhong University of Science and Technology (Nature Science)*, vol. 30, pp. 75–77, 2002.
- [8] C. Li, Y. Shao, and W. Li, "Measuring gas-solid two-phase mass flow based on thermal sensors fusion," *Journal Huazhong University of Science and Technology*, vol. 29, no. 8, pp. 50–52, 2001.
- [9] H. Zhang, W. Zhang, and D. Ling, "Data fusion based thermal sensors for mass flow measurement in pneumatic conveying," *Mathematical Problems in Engineering*, vol. 2014, Article ID 314920, 8 pages, 2014.
- [10] S. M. El-Behery, W. A. El-Askary, M. H. Hamed, and K. A. Ibrahim, "Hydrodynamic and thermal fields analysis in gassolid two-phase flow," *International Journal of Heat and Fluid Flow*, vol. 32, no. 3, pp. 740–754, 2011.
- [11] L. Farbar and M. J. Morley, "Heat transfer to flowing gas-solids mixtures in a circular tube," *Industrial & Engineering Chemistry*, vol. 49, no. 7, pp. 1143–1150, 1957.
- [12] L. Farbar and C. A. Depew, "Heat transfer effects to gassolids mixtures using solid spherical particles of uniform size," *Industrial & Engineering Chemistry Fundamentals*, vol. 2, no. 2, pp. 130–135, 1963.
- [13] G. Jepson, A. Poll, and W. Smith, "Heat transfer from gas to wall in a gas-solids transport line," *Transactions of the Institution of Chemical Engineers*, vol. 41, pp. 207–211, 1963.
- [14] M. Liu, C. Zhou, J. Xie, C. Lu, and Z. Wang, "Numerical investigation of performance of a fast gas-solid separator," *Powder Technology*, vol. 275, pp. 30–38, 2015.
- [15] W. H. Giedt, "Investigation of variation of point unit-heattransfer coefficient around a cylinder normal to an air stream," *Transactions of the ASME*, vol. 71, pp. 375–381, 1949.

Research Article

Global Asymptotic Stability of Switched Neural Networks with Delays

Zhenyu Lu,¹ Kai Li,¹ and Yan Li²

 ¹College of Electrical and Information Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China
 ²College of Science, Huazhong Agricultural University, Wuhan 430070, China

Correspondence should be addressed to Zhenyu Lu; luzhenyu76@163.com

Received 1 September 2015; Accepted 30 November 2015

Academic Editor: Yuan Fan

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

This paper investigates the global asymptotic stability of a class of switched neural networks with delays. Several new criteria ensuring global asymptotic stability in terms of linear matrix inequalities (LMIs) are obtained via Lyapunov-Krasovskii functional. And here, we adopt the quadratic convex approach, which is different from the linear and reciprocal convex combinations that are extensively used in recent literature. In addition, the proposed results here are very easy to be verified and complemented. Finally, a numerical example is provided to illustrate the effectiveness of the results.

1. Introduction

In the past thirty years, neural networks have found extensive applications in associative memory, pattern recognition, and image processing [1–3]. It is true that most applications of neural networks are heavily dependent on the dynamic behaviors of neural networks, especially on global asymptotic stability of neural networks. On the other hand, time delays are inevitably encountered in the hardware implementation due to the finite switching speed of amplifier, which may destroy the system performance and become a source of oscillation or instability in neural networks. Therefore, stability of neural networks with delays has attracted increasing attention and lots of stability criteria have been reported in the literature [4, 5].

As a special class of hybrid systems, switched systems are organized by a switching rule that orchestrates the switching. In reality, neural networks sometimes have finite modes that switch from one to another at different times according to a switching law. In [6, 7], the authors studied the stability problem of different kinds of switched neural networks with time delays. Different from the model in these works, in this paper, we consider a class of neural networks with state-dependent switchings. Our switched neural networks model is general and it generalizes the conventional neural networks.

Recently, convex analysis has been significantly employed in the stability analysis of time-delay systems [8–17]. According to the feature of different convex functions, different convex combination approaches are adopted in the literature, such as the linear convex combination [8–10], reciprocal convex combination [11–13], and quadratic convex combination [15–17]. In [8, 9, 12, 14, 16, 17], convex combination technology was successfully used to derive some stability criteria for neural networks with time delays. It should be pointed out that the lower bound of the time delay in [8, 9, 16] is zero, which means the information on the lower bound of the time delay cannot be sufficiently used. Namely, the conditions obtained in [9, 14, 16] fail to take effect on the stability of neural networks when the lower bound of the time delay is strictly greater than zero.

In this paper, some delay-dependent stability criteria in terms of LMIs are derived. The advantages are as follows. Firstly, differential inclusions and set-valued maps are employed to deal with the switched neural networks with discontinuous right-hand sides. Secondly, our results employ the quadratic convex approach, which is different from the linear and reciprocal convex combinations that are extensively used in recent literature on stability. Thirdly, the lower bound τ_1 of the time-varying delays is not zero and its information is adequately used to construct the Lyapunov-Krasovskii functional. Fourthly, we resort to neither Jensen's inequality with delay-dividing approach nor the free-weighting matrix method compared with previous results.

The organization of this paper is as follows. Some preliminaries are introduced in Section 2. In Section 3, based on the quadratic convex approach, delay-dependent stability criteria in terms of LMIs are established for switched neural networks with time-varying delays. Then, an example is given to demonstrate the effectiveness of the obtained results in Section 4. Finally, conclusions are given in Section 5.

Notations. Throughout this paper, \mathbb{R}^n denotes the *n*-dimensional Euclidean space. A^T and A^{-1} denote the transpose and the inverse of the matrix A, respectively. A > 0 ($A \ge 0$) means that the matrix A is symmetric and positive definite (semipositive definite). * represents the elements below the main diagonal of a symmetric matrix. The identity and zero matrices of appropriate dimensions are denoted by I and 0, respectively. SYM(A) is defined as SYM(A) = $A + A^T$. diag{ \cdots } denotes a block-diagonal matrix.

2. System Description and Preliminaries

In this paper, we consider a class of switched neural networks with delays as follows:

$$\begin{split} \dot{x}_{i}(t) &= -d_{i}\left(x_{i}(t)\right)x_{i}(t) + \sum_{j=1}^{n} a_{ij}\left(x_{i}(t)\right)f_{j}\left(x_{j}(t)\right) \\ &+ \sum_{j=1}^{n} b_{ij}\left(x_{i}(t)\right)f_{j}\left(x_{j}\left(t - \tau_{j}(t)\right)\right), \end{split}$$
(1)
$$t \geq 0, \ i = 1, 2, \dots, n, \end{split}$$

where

$$d_{i}(x_{i}(t)) = \begin{cases} d_{i}^{*}, & x_{i}(t) \leq 0, \\ d_{i}^{**}, & x_{i}(t) > 0, \end{cases}$$

$$a_{ij}(x_{i}(t)) = \begin{cases} a_{ij}^{*}, & x_{i}(t) \leq 0, \\ a_{ij}^{**}, & x_{i}(t) > 0, \end{cases}$$

$$b_{ij}(x_{i}(t)) = \begin{cases} b_{ij}^{*}, & x_{i}(t) \leq 0, \\ b_{ij}^{**}, & x_{i}(t) > 0, \end{cases}$$
(2)

 $x_i(t)$ is the state variable of the *i*th neuron, and $a_{ij}(x_i(t))$ and $b_{ii}(x_i(t))$ denote the feedback connection weight and the delayed feedback connection weight, respectively. $f_j : R \rightarrow R$ is bounded continuous function; $\tau_j(t)$ corresponds to the transmission delay and satisfies i, j = 1, 2, ..., n. $d_i^* > 0$, $d_i^{**} > 0, a_{ij}^*, a_{ij}^{**}, b_{ij}^*$, and $b_{ij}^{**}, i, j = 1, 2, ..., n$, are all constant numbers. The initial condition of system (1) is $x(s) = \phi(s) = (\phi_1(s), \phi_2(s), ..., \phi_n(s))^T \in \mathcal{C}([-\tau_2, 0], R^n)$.

The following assumptions are given for system (1):

(H1) For $j \in \{1, 2, ..., n, f_j\}$ is bounded and there exist constants h_i^-, h_i^+ such that

$$h_{j}^{-} \leq \frac{f_{j}(s_{1}) - f_{j}(s_{2})}{s_{1} - s_{2}} \leq h_{j}^{+}, \quad f_{j}(0) = 0,$$
(3)

for all $s_1, s_2 \in R, s_1 \neq s_2$.

(H2) The transmission delay $\tau_j(t)$ is a differential function and there exist constants $0 \le \tau_1 < \tau_2$, μ such that

$$0 \le \tau_1 \le \tau_j(t) \le \tau_2,$$

$$t_i(t) \le \mu,$$
(4)

for all
$$t \ge 0, j = 1, 2, ..., n$$
.

Obviously, system (1) is a discontinuous system; then its solution is different from the classic solution and cannot be defined in the conventional sense. In order to obtain the solution of system (1), some definitions and lemmas are given.

Definition 1. For a system with discontinuous right-hand sides,

$$\frac{dx}{dt} = F(t, x), \quad x(0) = x_0, \ x \in \mathbb{R}^n, \ t \ge 0.$$
 (5)

A set-valued map is defined as

$$\Phi(t, x) = \bigcap_{\delta > 0} \bigcap_{\mu(N) = 0} \overline{\operatorname{co}} \left[F(t, B(x, \delta) \setminus N) \right], \tag{6}$$

where $\overline{co}[E]$ is the closure of the convex hull of set $E, E \in \mathbb{R}^n$, $B(x, \delta) = \{y : ||y - x|| < \delta, x, y \in \mathbb{R}^n, \delta \in \mathbb{R}^+\}$, and $N \in \mathbb{R}^n$, $\mu(N)$ is Lebesgue measure of set N.

A solution in Filippov's sense of system (5) with initial condition $x(0) = x_0 \in \mathbb{R}^n$ is an absolutely continuous function x(t), $t \in [0,T]$, T > 0, which satisfy $x(0) = x_0$ and differential inclusion:

$$\frac{\mathrm{d}x}{\mathrm{d}t} \in \Phi(t, x), \quad \text{for } a.a. \ t \in [0, T].$$
(7)

If F(t, x) is bounded, then the set-valued function $\Phi(t, x)$ is upper semicontinuous with nonempty, convex, and compact values [18]. Then, the solution x(t) of system (5) with

Mathematical Problems in Engineering

initial condition exists and it can be extended to the interval $[0, +\infty)$ in the sense of Filippov.

By applying the theories of set-valued maps and differential inclusions [18–20], system (1) can be rewritten as the following differential inclusion:

$$\dot{x}_{i}(t) \in -\left[\underline{d}_{i}, \overline{d}_{i}\right] x_{i}(t) + \sum_{j=1}^{n} \left[\underline{a}_{ij}, \overline{a}_{ij}\right] f_{j}\left(x_{j}\left(t\right)\right) + \sum_{j=1}^{n} \left[\underline{b}_{ij}, \overline{b}_{ij}\right] f_{j}\left(x_{j}\left(t - \tau_{j}\left(t\right)\right)\right),$$
(8)
for *a.a.* $t \ge 0, \ i = 1, 2, ..., n,$

where $[\underline{\xi}_i, \overline{\xi}_i]$ is the convex hull of $[\underline{\xi}_i, \overline{\xi}_i], \underline{\xi}_i, \overline{\xi}_i \in \mathbb{R}, \underline{d}_i = \min\{d_i^*, d_i^{**}\}, \overline{d}_i = \max\{d_i^*, d_i^{**}\}, \underline{a}_{ij} = \min\{a_{ij}^*, a_{ij}^{**}\}, \overline{a}_{ij} = \max\{a_{ij}^*, a_{ij}^{**}\}, \underline{b}_{ij} = \min\{b_{ij}^*, b_{ij}^{**}\}, \text{and } \overline{b}_{ij} = \max\{b_{ij}^*, b_{ij}^{**}\}$. The other parameters are the same as in system (1).

Definition 2. A constant vector $x^* = (x_1^*, x_2^*, \dots, x_n^*)^T$ is called an equilibrium point of system (1), if, for $i = 1, 2, \dots, n$,

$$0 \in -\left[\underline{d}_{i}, \overline{d}_{i}\right] x_{i}^{*} + \sum_{j=1}^{n} \left[\underline{a}_{ij}, \overline{a}_{ij}\right] f_{j}\left(x_{j}^{*}\right) + \sum_{j=1}^{n} \left[\underline{b}_{ij}, \overline{b}_{ij}\right] f_{j}\left(x_{j}^{*}\right).$$

$$(9)$$

It is easy to find that the origin $(0, 0, ..., 0)^T$ is an equilibrium point of system (1).

Definition 3 (see [18]). A function $x(t) = (x_1(t), x_2(t), \dots, x_n(t))^T$ is a solution of (1), with the initial condition $x(s) = \phi(s) = (\phi_1(s), \phi_2(s), \dots, \phi_n(s))^T \in \mathscr{C}([-\tau_2, 0], \mathbb{R}^n)$, if x(t) is an absolutely continuous function and satisfies differential inclusion (8).

Lemma 4 (see [18]). Suppose that assumption (H1) is satisfied; then solution x(t) with initial condition $\phi(s) = (\phi_1(s), \phi_2(s), \dots, \phi_n(s))^T \in \mathcal{C}([-\tau_2, 0], \mathbb{R}^n)$ of (1) exists and it can be extended to the interval $[0, +\infty)$.

Before giving our main results, we present the following important lemmas that will be used in the proof to derive the stability conditions of the switched neural networks.

Lemma 5 (see [21]). *Given constant matrices* Σ_1, Σ_2 , and Σ_3 , where $\Sigma_1^T = \Sigma_1, \Sigma_2^T = \Sigma_2$,

$$\begin{pmatrix} \Sigma_1 & \Sigma_3 \\ \Sigma_3^T & -\Sigma_2 \end{pmatrix} < 0 \tag{10}$$

is equivalent to the following conditions:

$$\Sigma_2 > 0,$$

$$\Sigma_1 + \Sigma_3 \Sigma_2^{-1} \Sigma_3^T < 0.$$
(11)

Lemma 6 (see [17]). For real symmetric matrices $W_1, W_2, W_3 \in \mathbb{R}^{m \times m}$ and a scalar continuous function τ satisfy $\tau_1 \leq \tau \leq \tau_2$, where τ_1 and τ_2 are constants satisfying $0 \leq \tau_1 \leq \tau_2$. If $W_1 \geq 0$, then

$$\begin{aligned} \tau^2 W_1 + \tau W_2 + W_3 < 0, \quad \forall \tau \in [\tau_1, \tau_2] \\ \iff \tau_i^2 W_1 + \tau_i W_2 + W_3 < 0, \quad (i = 1, 2). \end{aligned}$$
(12)

Lemma 7 (see [16]). Let W > 0, and let y(s) be an appropriate dimensional vector. Then, one has the following facts for any scalar function $\eta(s) \ge 0$, $\forall s \in [t_1, t_2]$:

$$-\int_{t_{1}}^{t_{2}} y^{T}(s) Wy(s) ds \leq (t_{2} - t_{1}) \xi_{t}^{T} F_{1}^{T} W^{-1} F_{1} \xi_{t}$$

$$+ 2\xi_{t}^{T} F_{1}^{T} \int_{t_{1}}^{t_{2}} y(s) ds,$$
(13)

(ii)

$$-\int_{t_{1}}^{t_{2}} \eta(s) y^{T}(s) Wy(s) ds$$

$$\leq \int_{t_{1}}^{t_{2}} \eta(s) ds \xi_{t}^{T} F_{2}^{T} W^{-1} F_{2} \xi_{t} \qquad (14)$$

$$+ 2\xi_{t}^{T} F_{2}^{T} \int_{t_{1}}^{t_{2}} \eta(s) y(s) ds,$$

(iii)

$$-\int_{t_{1}}^{t_{2}} \eta^{2}(s) y^{T}(s) Wy(s) ds$$

$$\leq (t_{2} - t_{1}) \xi_{t}^{T} F_{3}^{T} W^{-1} F_{3} \xi_{t}$$
(15)

$$+ 2 \xi_{t}^{T} F_{3}^{T} \int_{t_{1}}^{t_{2}} \eta(s) y(s) ds,$$

where matrices F_i (i = 1, 2, 3) and vector ξ_t independent of the integral variable are appropriate dimensional arbitrary ones.

3. Main Results

For presentation convenience, in the following we denote $\tau_{21} = \tau_2 - \tau_1$, $H^- = \text{diag}\{h_1^-, h_2^-, \dots, h_n^-\}$, $H^+ = \text{diag}\{h_1^+, h_2^+, \dots, h_n^+\}$, $E_i = [0_{n \times (i-1)n}, I_n, 0_{n \times (10-i)n}]^T$, $i = 1, 2, \dots, 10$. $A_c = [-D, 0, 0, 0, 0, 0, 0, 0, A, B, 0]^T$, $D = \text{diag}(D_i)_{n \times n}$, $D_i = \min\{\underline{d}_i, \overline{d}_i\}$, $A = (A_{ij})_{n \times n}$, $A_{ij} = \max\{|\underline{a}_{ij}|, |\overline{a}_{ij}|\}$, $B = (B_{ij})_{n \times n}$, $B_{ij} = \max\{|\underline{b}_{ij}|, |\overline{b}_{ij}|\}$.

Theorem 8. Suppose assumptions (H1) and (H2) hold; the origin of system (1) is globally asymptotically stable if there exist matrices P > 0, S > 0, $Q_i > 0$ (i = 1, 2, 3, 4, 5), $R_j > 0$ (j = 1, 2, 3, 4), and $\Gamma = \text{diag}(\gamma_1, \gamma_2, \dots, \gamma_n) > 0$, $K_1 = \text{diag}(k_{11}, k_{12}, \dots, k_{1n}) > 0$, $K_2 = \text{diag}(k_{21}, k_{22}, \dots, k_{2n}) > 0$,

and F _i (i	=	1, 2,, 12),	such	that	the	following	two	LMIs
hold:								

$$\Xi = \begin{bmatrix} \Xi_{11} & \tau_1 F_4^T & \sqrt{3} \tau_1 F_6^T & \tau_1 F_5^T & \tau_{21} F_1^T & \sqrt{3} \tau_{21} F_3^T & \tau_{21} F_2^T & \sqrt{3} \tau_{21} F_9^T & \tau_{21} F_8^T \\ * & -\tau_1 Q_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & -\tau_1 R_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & -R_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & -\tau_{21} Q_3 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & * & * & * & -\tau_{21} R_2 & 0 & 0 & 0 \\ * & * & * & * & * & * & -\tau_{21} Q_5 & 0 & 0 \\ * & * & * & * & * & * & * & * & -\tau_{21} Q_4 & 0 \\ * & * & * & * & * & * & * & * & -\tau_{21} Q_5 & 0 & 0 \\ * & * & * & * & * & * & * & * & * & -\tau_{21} R_4 & 0 \\ * & * & * & * & * & * & * & * & * & -\tau_{21} R_4 & 0 \\ * & * & * & * & * & * & * & * & * & -R_3 \end{bmatrix}$$
(16)

where $\Xi_{11} = \Delta_0 + \text{SYM}((\tau_{21}E_2 - E_7)(2F_2 + 3F_3 + 2F_8 + 3F_9)) + \text{SYM}((\tau_1E_1 - (E_1 + E_5))(2F_5 + 3F_6)), \Pi_{11} = \Delta_0 + \text{SYM}([A_c, 0]Q_1[\tau_{21}E_1, E_6]^T) - \text{SYM}(E_7(2F_2 + 3F_3 + 2F_8 + 3F_9)) + \text{SYM}((\tau_2E_1 - (E_5 + E_6))(2F_5 + 3F_6)) \text{ with}$

$$\begin{split} \Delta_{0} &= \text{SYM}\left(\left[E_{1}, E_{5}, E_{6} + E_{7}\right] \\ &\cdot P\left[A_{c}, E_{1} - E_{3}, E_{3} - E_{4}\right]^{T} + E_{8}\Gamma A_{c}^{T}\right) + E_{8}SE_{8}^{T} \\ &- \left(1 - \mu\right)E_{9}SE_{9}^{T} + \left[E_{1}, E_{3}\right]\left(Q_{1} + Q_{4}\right)\left[E_{1}, E_{3}\right]^{T} \\ &- \left(1 - \mu\right)\left[E_{1}, E_{2}\right]Q_{1}\left[E_{1}, E_{2}\right]^{T} + \text{SYM}\left(\left[A_{c}, 0\right]\right] \\ &\cdot Q_{2}\left[\tau_{2}E_{1}, E_{5} + E_{6} + E_{7}\right]^{T} + \left[A_{c}, 0\right] \\ &\cdot Q_{4}\left[\tau_{21}E_{1}, E_{6} + E_{7}\right]^{T}\right) + \left[E_{1}, E_{1}\right]Q_{2}\left[E_{1}, E_{1}\right]^{T} \\ &- \left[E_{1}, E_{4}\right]\left(Q_{2} + Q_{4}\right)\left[E_{1}, E_{4}\right]^{T} + \tau_{2}\left[E_{1}, A_{c}\right] \\ &\cdot Q_{3}\left[E_{1}, A_{c}\right]^{T} + \tau_{21}\left[E_{3}, E_{10}\right]Q_{5}\left[E_{3}, E_{10}\right]^{T} \\ &+ A_{c}\left(\tau_{2}^{2}R_{1} + \tau_{2}^{3}R_{2}\right)A_{c}^{T} + E_{10}\left(\tau_{21}^{2}R_{3} + \tau_{31}^{3}R_{4}\right)E_{10}^{T} \\ &+ \text{SYM}\left(\left[E_{5} + E_{6}, E_{1} - E_{2}\right]F_{4} + \left[E_{7}, E_{2} - E_{4}\right]F_{7} \\ &+ \left[E_{6}, E_{3} - E_{2}\right]F_{10}\right) + \text{SYM}\left(E_{8}K_{1}H^{+}E_{1}^{T}\right] \end{split}$$

$$+ E_{1}H^{-}K_{1}E_{8}^{T} + E_{9}K_{2}H^{+}E_{2}^{T} + E_{2}H^{-}K_{2}E_{9}^{T} \Big) - SYM \Big(E_{8}K_{1}E_{8}^{T} + E_{1}H^{-}K_{1}H^{+}E_{1}^{T} + E_{9}K_{2}E_{9}^{T} + E_{2}H^{-}K_{2}H^{+}E_{2}^{T} \Big).$$
(17)

Proof. Define a vector $\xi_t \in \mathbb{R}^{10n}$ as

$$\xi_{t}^{T} = \left[x^{T}(t), x^{T}(t-\tau(t)), x^{T}(t-\tau_{1}), x^{T}(t-\tau_{2}), \right]$$

$$\int_{t-\tau_{1}}^{t} x^{T}(s) \, \mathrm{d}s, \int_{t-\tau(t)}^{t-\tau_{1}} x^{T}(s) \, \mathrm{d}s, \int_{t-\tau_{2}}^{t-\tau(t)} x^{T}(s) \, \mathrm{d}s, \qquad (18)$$

$$f^{T}(x(t)), f^{T}(x(t-\tau(t))), \dot{x}^{T}(t-\tau_{1}) .$$

Consider the following Lyapunov-Krasovskii functional candidate as follows:

$$V(t) = V_1(t) + V_2(t) + V_3(t) + V_4(t),$$
(19)

where

$$\begin{split} V_1\left(t\right) &= \zeta^T\left(t\right) P\zeta\left(t\right), \\ V_2\left(t\right) &= 2 \sum_{i=1}^n \gamma_i \int_0^{x_i\left(t\right)} f_i\left(s\right) \mathrm{d}s, \end{split}$$

$$\begin{aligned} V_{3}(t) &= \int_{t-\tau(t)}^{t} f^{T}(x(s)) Sf(x(s)) ds \\ &+ \int_{t-\tau(t)}^{t-\tau_{1}} \left[x^{T}(t) \ x^{T}(s) \right] Q_{1} \left[x^{T}(t) \ x^{T}(s) \right]^{T} ds, \\ V_{4}(t) &= \int_{t-\tau_{2}}^{t} \left\{ \left[x^{T}(t) \ x^{T}(s) \right] Q_{2} \left[x^{T}(t) \ x^{T}(s) \right]^{T} \\ &+ (\tau_{2} - t + s) \left[x^{T}(s) \ \dot{x}^{T}(s) \right] Q_{3} \left[x^{T}(s) \ \dot{x}^{T}(s) \right]^{T} \\ &+ (\tau_{2} - t + s)^{2} \dot{x}^{T}(s) R_{1} \dot{x}(s) + (\tau_{2} - t + s)^{3} \dot{x}^{T}(s) \\ &\cdot R_{2} \dot{x}(s) \right\} ds \\ &+ \int_{t-\tau_{2}}^{t-\tau_{1}} \left\{ \left[x^{T}(t) \ x^{T}(s) \right] Q_{4} \left[x^{T}(t) \ x^{T}(s) \right]^{T} \\ &+ (\tau_{2} - t + s) \left[x^{T}(s) \ \dot{x}^{T}(s) \right] Q_{5} \left[x^{T}(s) \ \dot{x}^{T}(s) \right]^{T} \\ &+ (\tau_{2} - t + s)^{2} \dot{x}^{T}(s) R_{3} \dot{x}(s) + (\tau_{2} - t + s)^{3} \dot{x}^{T}(s) \end{aligned}$$

where $\zeta^{T}(t) = \begin{bmatrix} x^{T}(t) & \int_{t-\tau_{1}}^{t} x^{T}(s) ds & \int_{t-\tau_{2}}^{t-\tau_{1}} x^{T}(s) ds \end{bmatrix}$. Calculating the time derivatives of $V_{i}(t)$ (i = 1, 2, 3, 4) along the trajectories of system (8), we obtain

$$\begin{split} \dot{V}_{1}(t) &= 2\zeta^{T}(t) P\dot{\zeta}(t) \leq 2x^{T}(t) \\ &\cdot P \begin{bmatrix} -Dx(t) + Af(x(t)) + Bf(x(t - \tau(t))) \\ x(t) - x(t - \tau_{1}) \\ x(t - \tau_{1}) - x(t - \tau_{2}) \end{bmatrix} \\ &\leq 2\xi_{t}^{T} [E_{1}, E_{5}, E_{6} + E_{7}] P [A_{c}, E_{1} - E_{3}, E_{3} - E_{4}]^{T} \xi_{t} \\ &= \xi_{t}^{T} SYM \left([E_{1}, E_{5}, E_{6} + E_{7}] \\ \cdot P [A_{c}, E_{1} - E_{3}, E_{3} - E_{4}]^{T} \right) \xi_{t}, \\ \dot{V}_{2}(t) &= 2\sum_{i=1}^{n} \gamma_{i} f_{i}(x_{i}(t)) \dot{x}(t) = 2f^{T}(x(t)) \Gamma \dot{x}(t) \\ &\leq 2\xi_{t}^{T} E_{8} \Gamma A_{c}^{T} \xi_{t}, \\ \dot{V}_{3}(t) &= f^{T}(x(t)) Sf(x(t)) - (1 - \dot{\tau}(t)) \\ &\cdot f^{T}(x(t - \tau(t))) Sf(x(t - \tau(t))) \\ &+ \left[x^{T}(t) x^{T}(t - \tau_{1}) \right] Q_{1} \left[x^{T}(t) x^{T}(t - \tau_{1}) \right]^{T} \\ &- (1 - \dot{\tau}(t)) \left[x^{T}(t) x^{T}(t - \tau(t)) \right] Q_{1} \left[x^{T}(t) \right] \end{split}$$

$$\begin{split} &+ 2 \int_{t-\tau_{1}}^{t-\tau_{1}} \frac{\partial \left[x^{T}\left(t\right) \ x^{T}\left(s\right)\right]}{\partial t} Q_{1} \left[x^{T}\left(t\right) \ x^{T}\left(s\right)\right]^{T} ds \\ &\leq f^{T}\left(x\left(t\right)\right) Sf\left(x\left(t\right)\right) - \left(1 - \mu\right) f^{T}\left(x\left(t - \tau\left(t\right)\right)\right) \\ &+ \left[x^{T}\left(t\right) \ x^{T}\left(t - \tau_{1}\right)\right] Q_{1} \left[x^{T}\left(t\right) \ x^{T}\left(t - \tau_{1}\right)\right]^{T} \\ &- \left(1 - \mu\right) \left[x^{T}\left(t\right) \ x^{T}\left(t - \tau\left(t\right)\right)\right] Q_{1} \left[x^{T}\left(t\right) \\ &\cdot x^{T}\left(t - \tau\left(t\right)\right)\right]^{T} + 2 \left[x^{T}\left(t\right), 0\right] \\ &\cdot Q_{1} \int_{t-\tau_{1}}^{t-\tau_{1}} \left[x^{T}\left(t\right) \ x^{T}\left(s\right)\right]^{T} ds \leq \xi_{t}^{T} \left\{E_{8}SE_{8}^{T} \\ &- \left(1 - \mu\right) E_{9}SE_{9}^{T} + \left[E_{1}, E_{3}\right] Q_{1} \left[E_{1}, E_{3}\right]^{T} - \left(1 - \mu\right) \\ &\cdot \left[E_{1}, E_{2}\right] Q_{1} \left[E_{1}, E_{2}\right]^{T} + 2 \left[A_{c}, 0\right] \\ &\cdot Q_{1} \left[\left(\tau\left(t\right) - \tau_{1}\right) E_{1}, E_{6}\right]^{T}\right] \xi_{t}, \\ \dot{V}_{4}\left(t\right) = \left[x^{T}\left(t\right) \ x^{T}\left(t\right)\right] Q_{2} \left[x^{T}\left(t\right) \ x^{T}\left(t - \tau_{2}\right)\right]^{T} \\ &+ \left[x^{T}\left(t\right) \ x^{T}\left(t - \tau_{2}\right)\right] Q_{2} \left[x^{T}\left(t\right) \ x^{T}\left(t - \tau_{2}\right)\right]^{T} \\ &- \left[x^{T}\left(t\right) \ x^{T}\left(t - \tau_{2}\right)\right] Q_{4} \left[x^{T}\left(t\right) \ x^{T}\left(s\right)\right]^{T} ds \\ &+ 2 \left[x^{T}\left(t\right), 0\right] Q_{2} \int_{t-\tau_{2}}^{t-\tau_{1}} \left[x^{T}\left(t\right) \ x^{T}\left(s\right)\right]^{T} ds \\ &+ \tau_{2} \left[x^{T}\left(t\right) \ x^{T}\left(t\right)\right] Q_{3} \left[x^{T}\left(t\right) \ x^{T}\left(t\right)\right]^{T} \\ &+ \tau_{21} \left[x^{T}\left(t - \tau_{1}\right)x^{T}\left(t - \tau_{1}\right)\right] Q_{5} \left[x^{T}\left(t - \tau_{1}\right) \\ &\cdot x^{T}\left(t - \tau_{1}\right)x^{T}\left(t - \tau_{1}\right) Q_{5} \left[x^{T}\left(t - \tau_{1}\right) \\ &\cdot x^{T}\left(t - \tau_{1}\right)\right]^{T} + \dot{x}^{T}\left(t\right) \left(\tau_{2}^{2}R_{1} + \tau_{2}^{3}R_{2}\right) \dot{x}\left(t\right) + \dot{x}^{T}\left(t\right) \\ &- \tau_{1}\left(\tau_{2}^{2}R_{3} + \tau_{2}^{3}R_{4}\right) \dot{x}\left(t - \tau_{1}\right) + V_{x} + V_{y} \\ &= \xi_{t}^{T} \left\{\left[E_{1}, E_{1}\right] Q_{2} \left[E_{1}, E_{1}\right]^{T} + \left[E_{1}, E_{3}\right] Q_{4} \left[E_{1}, E_{3}\right]^{T} \\ &- \left[E_{1}, E_{4}\right] \left(Q_{2} + q_{4}\right) \left[E_{1}, E_{4}\right]^{T} + 2 \left[A_{c}, 0\right] \\ &\cdot Q_{4} \left[\left(\tau_{2} - \tau_{1}\right) E_{1}, E_{6} + E_{7}\right]^{T} + \tau_{2} \left[E_{1}, A_{c}\right] \\ &\cdot Q_{4} \left[\left(\tau_{2} - \tau_{1}\right) E_{1}, E_{6} + E_{7}\right]^{T} + \tau_{2} \left[E_{1}, A_{c}\right] \\ &\cdot Z_{t} \left[x^{T} + x^{T}\right] \left[x^{T} + x^{T} + x^{T}\right] \left[x^{T} + x^{T} + x^{T}\right] \left[x^{T} + x^{T}\right] \left[$$

where

$$\begin{aligned} V_{x} &= -\int_{t-\tau_{2}}^{t} \left\{ \begin{bmatrix} x^{T}(s) \ \dot{x}^{T}(s) \end{bmatrix} Q_{3} \begin{bmatrix} x^{T}(s) \ \dot{x}^{T}(s) \end{bmatrix}^{T} \\ &+ 2(\tau_{2} - t + s) \dot{x}^{T}(s) R_{1} \dot{x}(s) \\ &+ 3(\tau_{2} - t + s)^{2} \dot{x}^{T}(s) R_{2} \dot{x}(s) \right\} ds, \end{aligned}$$
(22)
$$V_{y} &= -\int_{t-\tau_{2}}^{t-\tau_{1}} \left\{ \begin{bmatrix} x^{T}(s) \ \dot{x}^{T}(s) \end{bmatrix} Q_{5} \begin{bmatrix} x^{T}(s) \ \dot{x}^{T}(s) \end{bmatrix}^{T} \\ &+ 2(\tau_{2} - t + s) \dot{x}^{T}(s) R_{3} \dot{x}(s) \\ &+ 3(\tau_{2} - t + s)^{2} \dot{x}^{T}(s) R_{4} \dot{x}(s) \right\} ds. \end{aligned}$$

It is easy to obtain the following identities:

$$\tau_{2} - t + s = (\tau(t) - t + s) + (\tau_{2} - \tau(t)),$$

$$(\tau_{2} - t + s)^{2} = (\tau(t) - t + s)^{2} + (\tau_{2}^{2} - \tau^{2}(t))$$
(23)

$$+ 2(\tau_{2} - \tau(t))(s - t).$$

Thus, we have

$$\begin{split} V_{x} &= -\int_{t-\tau_{2}}^{t-\tau(t)} \left\{ \left[x^{T}\left(s\right) \ \dot{x}^{T}\left(s\right) \right] Q_{3} \left[x^{T}\left(s\right) \ \dot{x}^{T}\left(s\right) \right]^{T} \right. \\ &+ 2 \left(\tau_{2} - t + s \right) \dot{x}^{T}\left(s\right) R_{1} \dot{x}\left(s\right) \\ &+ 3 \left(\tau_{2} - t + s \right)^{2} \dot{x}^{T}\left(s\right) R_{2} \dot{x}\left(s\right) \right\} ds \\ &- \int_{t-\tau(t)}^{t} \left\{ \left[x^{T}\left(s\right) \ \dot{x}^{T}\left(s\right) \right] Q_{3} \left[x^{T}\left(s\right) \ \dot{x}^{T}\left(s\right) \right]^{T} \\ &+ 2 \left(\tau_{2} - t + s \right) \dot{x}^{T}\left(s\right) R_{1} \dot{x}\left(s\right) \\ &+ 3 \left(\tau_{2} - t + s \right)^{2} \dot{x}^{T}\left(s\right) R_{2} \dot{x}\left(s\right) \right\} ds \\ &= - \int_{t-\tau_{2}}^{t-\tau(t)} \left\{ \left[x^{T}\left(s\right) \ \dot{x}^{T}\left(s\right) \right] Q_{3} \left[x^{T}\left(s\right) \ \dot{x}^{T}\left(s\right) \right]^{T} \\ &+ 2 \left(\tau_{2} - t + s \right) \dot{x}^{T}\left(s\right) R_{1} \dot{x}\left(s\right) \\ &+ 3 \left(\tau_{2} - t + s \right) \dot{x}^{T}\left(s\right) R_{1} \dot{x}\left(s\right) \\ &+ 3 \left(\tau_{2} - t + s \right) \dot{x}^{T}\left(s\right) R_{2} \dot{x}\left(s\right) \right\} ds \\ &- \int_{t-\tau(t)}^{t} \left\{ \left[x^{T}\left(s\right) \ \dot{x}^{T}\left(s\right) \right] Q_{3} \left[x^{T}\left(s\right) \ \dot{x}^{T}\left(s\right) \right]^{T} \\ &+ 2 \left(\tau\left(t\right) - t + s \right) \dot{x}^{T}\left(s\right) R_{1} \dot{x}\left(s\right) \\ &+ 3 \left(\tau\left(t\right) - t + s \right) \dot{x}^{T}\left(s\right) R_{2} \dot{x}\left(s\right) \right\} ds \\ &- \int_{t-\tau(t)}^{t} \dot{x}^{T}\left(s\right) \left[2 \left(\tau_{2} - \tau\left(t\right) \right) R_{1} \\ &+ 3 \left(\tau_{2}^{2} - \tau^{2}\left(t\right) \right) R_{2} + 6 \left(\tau_{2} - \tau\left(t\right) \right) \left(s - t\right) R_{2} \right] \\ &\cdot \dot{x}\left(s\right) ds \end{split}$$

$$\leq -\int_{t-\tau(t)}^{t-\tau(t)} \left\{ \left[x^{T}(s) \ \dot{x}^{T}(s) \right] Q_{3} \left[x^{T}(s) \ \dot{x}^{T}(s) \right]^{T} \right. \\ \left. + 2 \left(\tau_{2} - t + s \right) \dot{x}^{T}(s) R_{1} \dot{x}(s) \right. \\ \left. + 3 \left(\tau_{2} - t + s \right)^{2} \dot{x}^{T}(s) R_{2} \dot{x}(s) \right\} ds \\ \left. - \int_{t-\tau(t)}^{t} \left\{ \left[x^{T}(s) \ \dot{x}^{T}(s) \right] Q_{3} \left[x^{T}(s) \ \dot{x}^{T}(s) \right]^{T} \right. \\ \left. + 2 \left(\tau(t) - t + s \right) \dot{x}^{T}(s) R_{1} \dot{x}(s) \right. \\ \left. + 3 \left(\tau(t) - t + s \right)^{2} \dot{x}^{T}(s) R_{2} \dot{x}(s) \right\} ds \\ \left. - 2 \int_{t-\tau(t)}^{t} \dot{x}^{T}(s) \left(\tau_{2} - \tau(t) \right) R_{1} \dot{x}(s) ds \right. \\ \left. - 3 \int_{t-\tau(t)}^{t} \dot{x}^{T}(s) \left(\tau_{2} - \tau(t) \right)^{2} R_{2} \dot{x}(s) ds \\ \left. \leq - \int_{t-\tau_{2}}^{t-\tau(t)} \left\{ \left[x^{T}(s) \ \dot{x}^{T}(s) \right] Q_{3} \left[x^{T}(s) \ \dot{x}^{T}(s) \right]^{T} \\ \left. + 2 \left(\tau_{2} - t + s \right) \dot{x}^{T}(s) R_{1} \dot{x}(s) \\ \left. + 3 \left(\tau_{2} - t + s \right)^{2} \dot{x}^{T}(s) R_{2} \dot{x}(s) \right\} ds \\ \left. - \int_{t-\tau(t)}^{t} \left\{ \left[x^{T}(s) \ \dot{x}^{T}(s) \right] Q_{3} \left[x^{T}(s) \ \dot{x}^{T}(s) \right]^{T} \\ \left. + 2 \left(\tau(t) - t + s \right) \dot{x}^{T}(s) R_{1} \dot{x}(s) \\ \left. + 3 \left(\tau(t) - t + s \right) \dot{x}^{T}(s) R_{1} \dot{x}(s) \right\} ds. \right]$$
 (24)

Similarly, we have

$$\begin{split} V_{y} &\leq -\int_{t-\tau_{2}}^{t-\tau(t)} \left\{ \begin{bmatrix} x^{T}(s) & \dot{x}^{T}(s) \end{bmatrix} Q_{5} \begin{bmatrix} x^{T}(s) & \dot{x}^{T}(s) \end{bmatrix}^{T} \\ &+ 2 (\tau_{2} - t + s) \dot{x}^{T}(s) R_{3} \dot{x}(s) \\ &+ 3 (\tau_{2} - t + s)^{2} \dot{x}^{T}(s) R_{4} \dot{x}(s) \right\} ds \\ &- \int_{t-\tau(t)}^{t-\tau_{1}} \left\{ \begin{bmatrix} x^{T}(s) & \dot{x}^{T}(s) \end{bmatrix} Q_{5} \begin{bmatrix} x^{T}(s) & \dot{x}^{T}(s) \end{bmatrix}^{T} \\ &+ 2 (\tau(t) - t + s) \dot{x}^{T}(s) R_{3} \dot{x}(s) \\ &+ 3 (\tau(t) - t + s)^{2} \dot{x}^{T}(s) R_{4} \dot{x}(s) \right\} ds. \end{split}$$
(25)

 $\begin{aligned} V_{x} &\leq \xi_{t}^{T} \left\{ \left(\tau_{2} - \tau\left(t\right)\right) F_{1}^{T} Q_{3}^{-1} F_{1} + 2F_{1}^{T} \left[E_{7}, E_{2} - E_{4}\right]^{T} \right. \\ &+ \left(\tau_{2} - \tau\left(t\right)\right)^{2} F_{2}^{T} R_{1}^{-1} F_{2} \\ &+ 4F_{2}^{T} \left[\left(\tau_{2} - \tau\left(t\right)\right) E_{2} - E_{7} \right]^{T} \end{aligned}$

$$+ 3 (\tau_{2} - \tau (t)) F_{3}^{T} R_{2}^{-1} F_{3}$$

$$+ 6F_{3}^{T} [(\tau_{2} - \tau (t)) E_{2} - E_{7}]^{T} + \tau (t) F_{4}^{T} Q_{3}^{-1} F_{4}$$

$$+ 2F_{4}^{T} [E_{5} + E_{6}, E_{1} - E_{2}]^{T} + \tau^{2} (t) F_{5}^{T} R_{1}^{-1} F_{5}$$

$$+ 4F_{5}^{T} [\tau (t) E_{1} - (E_{5} + E_{6})]^{T} + 3\tau (t) F_{6}^{T} R_{2}^{-1} F_{6}$$

$$+ 6F_{6}^{T} [\tau (t) E_{1} - (E_{5} + E_{6})]^{T} \} \xi_{t},$$

$$V_{y} \leq \xi_{t}^{T} \{(\tau_{2} - \tau (t)) F_{7}^{T} Q_{5}^{-1} F_{7} + 2F_{7}^{T} [E_{7}, E_{2} - E_{4}]^{T}$$

$$+ (\tau_{2} - \tau (t))^{2} F_{8}^{T} R_{3}^{-1} F_{8}$$

$$+ 4F_{8}^{T} [(\tau_{2} - \tau (t)) E_{2} - E_{7}]^{T}$$

$$+ 3 (\tau_{2} - \tau (t)) F_{9}^{T} R_{4}^{-1} F_{9}$$

$$+ 6F_{9}^{T} [(\tau_{2} - \tau (t)) E_{2} - E_{7}]^{T}$$

$$+ (\tau (t) - \tau_{1}) F_{10}^{T} Q_{5}^{-1} F_{10} + 2F_{10}^{T} [E_{6}, E_{3} - E_{2}]^{T}$$

$$+ (\tau (t) - \tau_{1})^{2} F_{11}^{T} R_{3}^{-1} F_{11}$$

$$+ 4F_{11}^{T} [(\tau (t) - \tau_{1}) E_{3} - E_{6}]^{T}$$

$$+ 3 (\tau (t) - \tau_{1}) F_{12}^{T} R_{4}^{-1} F_{12}$$

$$+ 6F_{12}^{T} [(\tau (t) - \tau_{1}) E_{3} - E_{6}]^{T} \} \xi_{t}.$$

$$(26)$$

On the other hand, from assumption (H1), we have

$$-2(f_{i}(x_{i}(t)) - h_{i}^{-}x_{i}(t))K_{1}(f_{i}(x_{i}(t)) - h_{i}^{+}x_{i}(t))$$

$$\geq 0,$$

$$-2(f_{i}(x_{i}(t - \tau(t))) - h_{i}^{-}x_{i}(t - \tau(t)))$$

$$\cdot K_{1}(f_{i}(x_{i}(t - \tau(t))) - h_{i}^{+}x_{i}(t - \tau(t))) \geq 0,$$
(27)

where $K_1 = \text{diag}(k_{11}, k_{12}, \dots, k_{1n}) > 0$, $K_2 = \text{diag}(k_{21}, k_{22}, \dots, k_{2n}) > 0$, $i = 1, 2, \dots, n$. From (27), it is easy to get that

$$2\xi_{t}^{T} \left\{ -E_{8}K_{1}E_{8}^{T} + E_{8}K_{1}H^{+}E_{1}^{T} + E_{1}H^{-}K_{1}E_{8}^{T} - E_{1}H^{-}K_{1}H^{+}E_{1}^{T} - E_{9}K_{2}E_{9}^{T} + E_{9}K_{2}H^{+}E_{2}^{T} + E_{2}H^{-}K_{2}E_{9}^{T} - E_{2}H^{-}K_{2}H^{+}E_{2}^{T} \right\} \ge 0.$$

$$(28)$$

It follows from (19)-(26) and (28) that

$$\dot{V}(t) \leq \xi_t^T \left\{ 2 \left[E_1, E_5, E_6 + E_7 \right] \right.$$

$$\cdot P \left[A_c, E_1 - E_3, E_3 - E_4 \right]^T + 2E_8 \Gamma A_c^T + E_8 S E_8^T$$

$$- \left(1 - \mu \right) E_9 S E_9^T + \left[E_1, E_3 \right] Q_1 \left[E_1, E_3 \right]^T - \left(1 - \mu \right)$$

$$\cdot \left[E_1, E_2 \right] Q_1 \left[E_1, E_2 \right]^T + 2 \left[A_c, 0 \right]$$

$$\begin{split} \cdot Q_{1} \left[\left(\tau \left(t \right) - \tau_{1} \right) E_{1}, E_{6} \right]^{T} + \left[E_{1}, E_{1} \right] Q_{2} \left[E_{1}, E_{1} \right]^{T} \right]^{T} \\ + \left[E_{1}, E_{3} \right] Q_{4} \left[E_{1}, E_{3} \right]^{T} - \left[E_{1}, E_{4} \right] \left(Q_{2} + q_{4} \right) \\ \cdot \left[E_{1}, E_{4} \right]^{T} + 2 \left[A_{c}, 0 \right] Q_{2} \left[\tau_{2} E_{1}, E_{5} + E_{6} + E_{7} \right]^{T} \\ + 2 \left[A_{c}, 0 \right] Q_{4} \left[\tau_{21} E_{1}, E_{6} + E_{7} \right]^{T} + \tau_{2} \left[E_{1}, A_{c} \right] \\ \cdot Q_{3} \left[E_{1}, A_{c} \right]^{T} + \tau_{21} \left[E_{3}, E_{10} \right] Q_{5} \left[E_{3}, E_{10} \right]^{T} \\ + A_{c} \left(\tau_{2}^{2} R_{1} + \tau_{2}^{3} R_{2} \right) A_{c}^{T} + E_{10} \left(\tau_{21}^{2} R_{3} + \tau_{31}^{3} R_{4} \right) E_{10}^{T} \\ + \left(\tau_{2} - \tau \left(t \right) \right) F_{1}^{T} Q_{3}^{-1} F_{1} + 2F_{1}^{T} \left[E_{7}, E_{2} - E_{4} \right]^{T} \\ + \left(\tau_{2} - \tau \left(t \right) \right) E_{2}^{-} E_{7} \right]^{T} + 3 \left(\tau_{2} - \tau \left(t \right) \right) \\ \cdot F_{3}^{T} R_{2}^{-1} F_{3} + 6F_{3}^{T} \left[\left(\tau_{2} - \tau \left(t \right) \right) E_{2} - E_{7} \right]^{T} + \tau \left(t \right) \\ \cdot F_{4}^{T} Q_{3}^{-1} F_{4} + 2F_{4}^{T} \left[E_{5} + E_{6}, E_{1} - E_{2} \right]^{T} + \tau^{2} \left(t \right) \\ \cdot F_{5}^{T} R_{1}^{-1} F_{5} + 4F_{5}^{T} \left[\tau \left(t \right) E_{1} - \left(E_{5} + E_{6} \right) \right]^{T} \\ + \left(\tau_{2} - \tau \left(t \right) \right) F_{7}^{T} Q_{5}^{-1} F_{7} + 2F_{7}^{T} \left[E_{7}, E_{2} - E_{4} \right]^{T} \\ + \left(\tau_{2} - \tau \left(t \right) \right) F_{7}^{T} Q_{5}^{-1} F_{7} + 2F_{7}^{T} \left[E_{7}, E_{2} - E_{4} \right]^{T} \\ + \left(\tau_{2} - \tau \left(t \right) \right) F_{7}^{T} Q_{5}^{-1} F_{7} + 2F_{7}^{T} \left[E_{7}, E_{2} - E_{4} \right]^{T} \\ + \left(\tau_{2} - \tau \left(t \right) \right) F_{7}^{T} Q_{5}^{-1} F_{7} + 2F_{7}^{T} \left[E_{7}, E_{2} - E_{4} \right]^{T} \\ + \left(\tau \left(t - \tau_{1} \right) \right) F_{7}^{T} Q_{5}^{-1} F_{7} + 2F_{7}^{T} \left[E_{7}, E_{2} - \tau \left(t \right) \right) \\ \cdot F_{9}^{T} R_{4}^{-1} F_{9} + 6F_{9}^{T} \left[\left(\tau_{2} - \tau \left(t \right) \right) E_{2} - E_{7} \right]^{T} \\ + \left(\tau \left(t \right) - \tau_{1} \right) F_{10}^{T} Q_{5}^{-1} F_{10} + 2F_{10}^{T} \left[E_{6}, E_{3} - E_{2} \right]^{T} \\ + \left(\tau \left(t \right) - \tau_{1} \right) F_{10}^{T} Q_{5}^{-1} F_{10} + 2F_{10}^{T} \left[E_{6}, E_{3} - E_{2} \right]^{T} \\ + \left(\tau \left(t \right) - \tau_{1} \right) F_{10}^{T} Q_{5}^{-1} F_{10} + 2F_{10}^{T} \left[E_{6}, E_{3} - E_{2} \right]^{T} \\ + \left(\tau \left(t \right) - \tau_{1} \right) F_{10}^{T} R_{3}^{-1} F_{11} \\ + 4F_{11}$$

where Δ_0 is defined in the theorem context, and

$$\begin{split} \Delta_{1} &= 2 \left[A_{c}, 0 \right] Q_{1} \left[\left(\tau \left(t \right) - \tau_{1} \right) E_{1}, E_{6} \right]^{T} \\ &+ \left(\tau_{2} - \tau \left(t \right) \right) F_{1}^{T} Q_{3}^{-1} F_{1} + \left(\tau_{2} - \tau \left(t \right) \right)^{2} F_{2}^{T} R_{1}^{-1} F_{2} \\ &+ 4 F_{2}^{T} \left[\left(\tau_{2} - \tau \left(t \right) \right) E_{2} - E_{7} \right]^{T} \\ &+ 3 \left(\tau_{2} - \tau \left(t \right) \right) F_{3}^{T} R_{2}^{-1} F_{3} \end{split}$$

$$+ 6F_{3}^{T} [(\tau_{2} - \tau(t)) E_{2} - E_{7}]^{T} + \tau(t) F_{4}^{T} Q_{3}^{-1} F_{4} + \tau^{2}(t) F_{5}^{T} R_{1}^{-1} F_{5} + 4F_{5}^{T} [\tau(t) E_{1} - (E_{5} + E_{6})]^{T} + 3\tau(t) F_{6}^{T} R_{2}^{-1} F_{6} + 6F_{6}^{T} [\tau(t) E_{1} - (E_{5} + E_{6})]^{T} + (\tau_{2} - \tau(t)) F_{7}^{T} Q_{5}^{-1} F_{7} + (\tau_{2} - \tau(t))^{2} F_{8}^{T} R_{3}^{-1} F_{8} + 4F_{8}^{T} [(\tau_{2} - \tau(t)) E_{2} - E_{7}]^{T} + 3(\tau_{2} - \tau(t)) F_{9}^{T} R_{4}^{-1} F_{9} + 6F_{9}^{T} [(\tau_{2} - \tau(t)) E_{2} - E_{7}]^{T} + (\tau(t) - \tau_{1}) F_{10}^{T} Q_{5}^{-1} F_{10} + (\tau(t) - \tau_{1})^{2} F_{11}^{T} R_{3}^{-1} F_{11} + 4F_{11}^{T} [(\tau(t) - \tau_{1}) E_{3} - E_{6}]^{T} + 3(\tau(t) - \tau_{1}) F_{12}^{T} R_{4}^{-1} F_{12} + 6F_{12}^{T} [(\tau(t) - \tau_{1}) E_{3} - E_{6}]^{T} .$$

$$(30)$$

It is easy to see that $\Delta_0 + \Delta_1$ is a quadratic convex combination of matrices on $\tau(t) \in [\tau_1, \tau_2]$.

Applying Lemma 5 to (16), we have

$$\begin{split} (\Delta_{0} + \Delta_{1})|_{\tau(t)=\tau_{1}} &= \Delta_{0} + \Delta_{1}|_{\tau(t)=\tau_{1}} = \Delta_{0} + 2(\tau_{21}E_{2} \\ &- E_{7})(2F_{2} + 3F_{3} + 2F_{8} + 3F_{9}) + 2(\tau_{1}E_{1} \\ &- (E_{1} + E_{5}))(2F_{5} + 3F_{6}) + \tau_{1}(F_{4}^{T}Q_{3}^{-1}F_{4} \\ &+ 3F_{6}^{T}R_{2}^{-1}F_{6}) + \tau_{1}^{2}F_{5}^{T}R_{1}^{-1}F_{5} + \tau_{21}(F_{1}^{T}Q_{3}^{-1}F_{1} \\ &+ F_{7}^{T}Q_{5}^{-1}F_{7} + 3F_{3}^{T}R_{2}^{-1}F_{3} + 3F_{9}^{T}R_{4}^{-1}F_{9}) \\ &+ \tau_{21}^{2}(F_{2}^{T}R_{1}^{-1}F_{2} + F_{8}^{T}R_{3}^{-1}F_{8}) < 0, \end{split}$$
(31)
$$(\Delta_{0} + \Delta_{1})|_{\tau(t)=\tau_{2}} = \Delta_{0} + \Delta_{1}|_{\tau(t)=\tau_{2}} = \Delta_{0} + 2[A_{c}, 0] \\ &\cdot Q_{1}[\tau_{21}E_{1}, E_{6}]^{T} - 2E_{7}(2F_{2} + 3F_{3} + 2F_{8} + 3F_{9}) \\ &+ 2(\tau_{2}E_{1} - (E_{5} + E_{6}))(2F_{5} + 3F_{6}) + \tau_{2}(F_{4}^{T}Q_{3}^{-1}F_{4} \\ &+ 3F_{6}^{T}R_{2}^{-1}F_{6}) + \tau_{2}^{2}F_{5}^{T}R_{1}^{-1}F_{5} < 0. \end{split}$$

Since $\xi_t^T \{F_2^T R_1^{-1} F_2 + F_5^T R_1^{-1} F_5 + F_8^T R_3^{-1} F_8 + F_{11}^T R_3^{-1} F_{11}\} \xi_t \ge 0$, from Lemma 6, if LMIs (16) are true, then $\Delta_0 + \Delta_1 < 0$, $\forall \tau(t) \in [\tau_1, \tau_2]$. Then, we can see that the origin of system (1) is asymptotically stable.

The proof is completed.

Remark 9. In [16], stability analysis for neural networks with time-varying delay is studied by using quadratic convex combination. Our results have two advantages compared with the results of that paper. On the one hand, the information on the lower bound τ_1 of the time-varying delays is considered.

On the other hand, the augmented vector $\zeta(t)$ includes the distributed delay terms.

Remark 10. We use three inequalities in Lemma 7 combined with the quadratic convex combination implied by Lemma 6, rather than Jensen's inequality and the linear convex combination. In addition, our theoretical proof is not concerned with free-weighting matrix method.

Remark 11. To use the quadratic convex approach, we construct the Lyapunov-Krasovskii functional with the following term: $\int_{t-\tau_2}^t \sum_{j=1}^3 (\tau_2 + t - s)^j g(s) ds$. The degree increase of $\tau_2 - t + s$ by 1 means the number increase of the integral by 1 due to the fact that $\int_{\tau_2}^t \int_{t+\theta}^t g(s) ds d\theta = \int_{t-\tau_2}^t (\tau_2 - t + s)g(s) ds$. In the case $\tau_1 = 0$, we have the following result.

Corollary 12. Suppose assumptions (H1) and (H2) hold with $\tau_1 = 0$; the origin of system (1) is globally asymptotically stable if there exist matrices P > 0, S > 0, $Q_i > 0$ (i = 1, 2, 3), $R_j > 0$ (j = 1, 2), and $\Gamma = \text{diag}(\gamma_1, \gamma_2, ..., \gamma_n) > 0$, $K_1 = \text{diag}(k_{11}, k_{12}, ..., k_{1n}) > 0$, $K_2 = \text{diag}(k_{21}, k_{22}, ..., k_{2n}) > 0$, and F_i (i = 1, 2, ..., 6), such that the following two LMIs hold:

$$\Xi = \begin{bmatrix} \Delta_0 & \tau_2 F_1^T & \sqrt{3}\tau_2 F_3^T & \tau_2 F_2^T \\ * & -\tau_2 Q_3 & 0 & 0 \\ * & * & -\tau_2 R_2 & 0 \\ * & * & * & -R_1 \end{bmatrix} < 0,$$

$$\Pi = \begin{bmatrix} \Pi_{11} & \tau_2 F_4^T & \sqrt{3}\tau_2 F_6^T & \tau_2 F_5^T \\ * & -\tau_2 Q_3 & 0 & 0 \\ * & * & -\tau_2 R_2 & 0 \\ * & * & * & -R_1 \end{bmatrix} < 0,$$
(32)

where $\Pi_{11} = \Delta_0 + \tau_2 \text{SYM}([A_c, 0]Q_1[E_1, 0]^T) - \tau_2 \text{SYM}(E_2(2F_2 + 3F_3)) + \tau_2 \text{SYM}(E_1(2F_5 + 3F_6))$ with

$$\begin{split} &\Delta_{0} = \text{SYM}\left(\left[E_{1}, E_{4} + E_{5}\right] P\left[A_{c}, E_{1} - E_{3}\right]^{T} + E_{6}\Gamma A_{c}^{T}\right) \\ &+ E_{6}SE_{6}^{T} - (1 - \mu) E_{7}SE_{7}^{T} + \left[E_{1}, E_{1}\right] (Q_{1} + Q_{2}) \\ &\cdot \left[E_{1}, E_{1}\right]^{T} - (1 - \mu) \left[E_{1}, E_{2}\right] Q_{1} \left[E_{1}, E_{2}\right]^{T} \\ &- \left[E_{1}, E_{3}\right] Q_{2} \left[E_{1}, E_{3}\right]^{T} + \text{SYM}\left(\left[A_{c}, 0\right] Q_{1} \left[0, E_{5}\right]^{T} \right. \\ &+ \left[A_{c}, 0\right] Q_{2} \left[\tau_{2}E_{1}, E_{4} + E_{5}\right]^{T}\right) + \tau_{2} \left[E_{1}, A_{c}\right] \\ &\cdot Q_{3} \left[E_{1}, A_{c}\right]^{T} + A_{c} \left(\tau_{2}^{2}R_{1} + \tau_{2}^{3}R_{2}\right) A_{c}^{T} \\ &+ \text{SYM} \left(\left[E_{5}, E_{1} - E_{2}\right] F_{1} \\ &+ \left(\tau_{2}E_{2} - E_{5}\right) \left(2F_{2} + 3F_{3}\right) + \left[E_{4}, E_{1} - E_{2}\right] F_{4}\right) \\ &- \text{SYM} \left(E_{4} \left(2F_{5} + 3F_{6}\right)\right) + \text{SYM} \left(E_{8}K_{1}H^{+}E_{1}^{T} \\ &+ E_{1}H^{-}K_{1}E_{8}^{T} + E_{9}K_{2}H^{+}E_{2}^{T} + E_{2}H^{-}K_{2}E_{9}^{T}\right) \end{split}$$

$$- \operatorname{SYM} \left(E_8 K_1 E_8^T + E_1 H^- K_1 H^+ E_1^T + E_9 K_2 E_9^T + E_2 H^- K_2 H^+ E_2^T \right).$$
(33)

In addition, when the information of the time derivative of delays is unknown or the derivative of the delays does not exist, then we have the following result.

Corollary 13. Suppose assumption (H1) holds with $\tau_1 = 0$; the origin of system (1) is globally asymptotically stable if there exist matrices P > 0, $Q_i > 0$ (i = 2, 3), $R_j > 0$ (j = 1, 2), and $\Gamma = \text{diag}(\gamma_1, \gamma_2, \ldots, \gamma_n) > 0$, $K_1 = \text{diag}(k_{11}, k_{12}, \ldots, k_{1n}) > 0$, $K_2 = \text{diag}(k_{21}, k_{22}, \ldots, k_{2n}) > 0$, and F_i ($i = 1, 2, \ldots, 6$), such that the following two LMIs hold:

$$\Xi = \begin{bmatrix} \Delta_{0} & \tau_{2}F_{1}^{T} & \sqrt{3}\tau_{2}F_{3}^{T} & \tau_{2}F_{2}^{T} \\ * & -\tau_{2}Q_{3} & 0 & 0 \\ * & * & -\tau_{2}R_{2} & 0 \\ * & * & * & -R_{1} \end{bmatrix} < 0,$$

$$\Pi = \begin{bmatrix} \Pi_{11} & \tau_{2}F_{4}^{T} & \sqrt{3}\tau_{2}F_{6}^{T} & \tau_{2}F_{5}^{T} \\ * & -\tau_{2}Q_{3} & 0 & 0 \\ * & * & -\tau_{2}R_{2} & 0 \\ * & * & * & -R_{1} \end{bmatrix} < 0,$$
(34)

where $\Pi_{11} = \Delta_0 + \tau_2 \text{SYM}([A_c, 0]Q_1[E_1, 0]^T) - \tau_2 \text{SYM}(E_2(2F_2 + 3F_3)) + \tau_2 \text{SYM}(E_1(2F_5 + 3F_6))$ with

$$\Delta_{0} = \text{SYM}\left(\left[E_{1}, E_{4} + E_{5}\right] P\left[A_{c}, E_{1} - E_{3}\right]^{T} + E_{6}\Gamma A_{c}^{T}\right) \\ + \left[E_{1}, E_{1}\right] Q_{2}\left[E_{1}, E_{1}\right]^{T} - \left[E_{1}, E_{3}\right] Q_{2}\left[E_{1}, E_{3}\right]^{T} \\ + \text{SYM}\left(\left[A_{c}, 0\right] Q_{1}\left[0, E_{5}\right]^{T} \\ + \left[A_{c}, 0\right] Q_{2}\left[\tau_{2}E_{1}, E_{4} + E_{5}\right]^{T}\right) + \tau_{2}\left[E_{1}, A_{c}\right] \\ \cdot Q_{3}\left[E_{1}, A_{c}\right]^{T} + A_{c}\left(\tau_{2}^{2}R_{1} + \tau_{2}^{3}R_{2}\right) A_{c}^{T} \\ + \text{SYM}\left(\left[E_{5}, E_{1} - E_{2}\right]F_{1} \right] \\ \cdot \left(\tau_{2}E_{2} - E_{5}\right)\left(2F_{2} + 3F_{3}\right) + \left[E_{4}, E_{1} - E_{2}\right]F_{4}\right) \\ - \text{SYM}\left(E_{4}\left(2F_{5} + 3F_{6}\right)\right) + \text{SYM}\left(E_{8}K_{1}H^{+}E_{1}^{T} \\ + E_{1}H^{-}K_{1}E_{8}^{T} + E_{9}K_{2}H^{+}E_{2}^{T} + E_{2}H^{-}K_{2}E_{9}^{T}\right) \\ - \text{SYM}\left(E_{8}K_{1}E_{8}^{T} + E_{1}H^{-}K_{1}H^{+}E_{1}^{T} + E_{9}K_{2}E_{9}^{T} \\ + E_{2}H^{-}K_{2}H^{+}E_{2}^{T}\right).$$

Remark 14. It is worth noting that when we consider system (1) without switching, that is, $d_i^* = d_i^{**}$, $a_{ij}^* = a_{ij}^{**}$, and $b_{ij}^* = b_{ij}^{**}$, then Corollary 12 is the main Theorem 1 of [16].

Remark 15. Compared with the results on stability of neural networks with continuous right-hand side [6], our results

on stability of neural networks are with discontinuous righthand sides. So the results of this paper are less conservative and more general.

4. Numerical Example

In this section, an example is provided to verify the effectiveness of the results obtained in the previous section.

Example 1. Consider two-dimensional switched neural networks with time-varying delays

$$\dot{x}_{i}(t) = -d_{i}(x_{i}(t))x_{i}(t) + \sum_{j=1}^{2} a_{ij}(x_{i}(t))f_{j}(x_{j}(t))$$

$$+ \sum_{j=1}^{2} b_{ij}(x_{i}(t))f_{j}(x_{j}(t - \tau_{j}(t))),$$

$$t \ge 0, \ i = 1, 2,$$
(36)

where

$$\begin{aligned} d_1(x_1(t)) &= \begin{cases} 2, & x_1(t) \le 0, \\ 2.1, & x_1(t) > 0, \end{cases} \\ d_2(x_2(t)) &= \begin{cases} 2.1, & x_2(t) \le 0, \\ 2, & x_2(t) > 0, \end{cases} \\ a_{11}(x_1(t)) &= \begin{cases} 1, & x_1(t) \le 0, \\ 0.9, & x_1(t) > 0, \end{cases} \\ a_{12}(x_1(t)) &= \begin{cases} 1, & x_1(t) \le 0, \\ 1.2, & x_1(t) > 0, \end{cases} \\ a_{21}(x_2(t)) &= \begin{cases} -0.9, & x_2(t) \le 0, \\ -1, & x_2(t) > 0, \end{cases} \\ a_{22}(x_2(t)) &= \begin{cases} -1.2, & x_2(t) \le 0, \\ -1, & x_2(t) > 0, \end{cases} \\ a_{11}(x_1(t)) &= \begin{cases} -0.7, & x_1(t) \le 0, \\ -0.8, & x_1(t) > 0, \end{cases} \\ b_{12}(x_1(t)) &= \begin{cases} 1, & x_1(t) \le 0, \\ 1.2, & x_1(t) > 0, \end{cases} \\ b_{21}(x_2(t)) &= \begin{cases} 1, & x_2(t) \le 0, \\ 0.9, & x_2(t) > 0, \end{cases} \\ b_{22}(x_2(t)) &= \begin{cases} -1, & x_2(t) \le 0, \\ 0.9, & x_2(t) > 0, \end{cases} \end{aligned} \end{aligned}$$

(37)

with $\tau_j(t) = 0.5 \cos(1.6t) + 3$, j = 1, 2, and take the activation function as $f(x) = (\sin(x_1), \sin(x_2))^T$. We can obtain that $\tau_1 = 2.5, \tau_2 = 3.5, \tau_{21} = 1, \mu = 0.8$,

$$D = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix},$$

$$A = \begin{bmatrix} 1 & 1.2 \\ 1 & 1.2 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.8 & 1.2 \\ 1 & 1.2 \end{bmatrix}$$

$$H^{-} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix},$$

$$H^{+} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$
(38)

For written simplification, we use the Matlab LMI Control Toolbox and then a solution to LMIs (32) is obtained (F_i , i = 1, ..., 6, is omitted since the dimension is too big) as follows:

$$P = \begin{bmatrix} 28.5171 & -20.9241 & 0.5484 & -0.4280 \\ -20.9241 & 27.7701 & -0.4064 & 0.4984 \\ 0.5484 & -0.4064 & 10.8451 & -3.9493 \\ -0.4280 & 0.4984 & -3.9493 & 10.6288 \end{bmatrix}$$

$$Q_1 = \begin{bmatrix} 7.3594 & -1.0832 & 0.0588 & -0.0060 \\ -1.0832 & 7.4606 & -0.0074 & 0.0723 \\ 0.0588 & -0.0074 & 3.5541 & -0.4045 \\ -0.0060 & 0.0723 & -0.4045 & 3.6751 \end{bmatrix}$$

$$Q_2 = \begin{bmatrix} 67.4363 & -64.0296 & 3.2893 & -3.0414 \\ -64.0296 & 65.2276 & -3.0862 & 0.0723 \\ 3.2893 & -3.0862 & 35.1371 & -5.2829 \\ -3.0414 & 0.0723 & -5.2829 & 34.8395 \end{bmatrix}$$

$$Q_3 = \begin{bmatrix} 15.7736 & -2.2877 & 0.8863 & -0.8539 \\ -2.2877 & 15.6801 & -0.8085 & 0.4984 \\ 0.8863 & -0.8085 & 7.6796 & -6.7268 \\ -0.8539 & 0.4984 & -6.7268 & 7.3682 \end{bmatrix}$$

$$S = \begin{bmatrix} 2.4075 & 1.0722 \\ 1.0722 & 2.5766 \end{bmatrix}$$

$$G = \begin{bmatrix} 3.5681 & 0 \\ 0 & 3.7071 \end{bmatrix}$$



FIGURE 1: The state curves of system (36).

$$K_{1} = \begin{bmatrix} 57.4100 & 0 \\ 0 & 65.2979 \end{bmatrix},$$

$$K_{2} = \begin{bmatrix} 26.5401 & 0 \\ 0 & 29.9701 \end{bmatrix}.$$
(39)

Therefore, according to Corollary 12, we can see that the origin of system (36) is globally asymptotically stable. The state trajectories of variables $x_1(t)$ and $x_2(t)$ are shown in Figure 1.

Remark 16. Because the parameters of system (1) are discontinuous, the results obtained in [6] about neural networks with continuous right-hand sides cannot be used here. In addition, the lower bounds of the delays of system (36) are not zero, so the results obtained in [8, 9, 16] cannot be used here.

5. Conclusions

In this paper, the delay-dependent stability for a class of switched neural networks with time-varying delays has been studied by using the quadratic convex combination. Some delay-dependent criteria in terms of LMIs have been obtained. The lower bound τ_1 of the time-varying delays is considered to be nonzero so that the information of τ_1 can be used adequately. It is worth noting that we resort to neither Jensen's inequality with delay-dividing approach nor the free-weighting matrix method compared with previous results.

Conflict of Interests

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

Acknowledgments

The authors would like to thank the financial supports from the National Natural Science Foundation of China (nos. 61304068 and 61473334), Jiangsu Qing Lan Project, and PAPD.

References

- A. Cichocki and R. Unbehauen, Neural Networks for Optimization and Signal Processing, Wiley, New York, NY, USA, 1993.
- [2] L. O. Chua and L. Yang, "Cellular neural networks: applications," *IEEE Transactions on Circuits and Systems*, vol. 35, no. 10, pp. 1273–1290, 1988.
- [3] Z. Zeng and J. Wang, "Analysis and design of associative memories based on recurrent neural networks with linear saturation activation functions and time-varying delays," *Neural Computation*, vol. 19, no. 8, pp. 2149–2182, 2007.
- [4] X. Liao and S. Guo, "Delay-dependent asymptotic stability of Cohen-Grossberg models with multiple time-varying delays," *Discrete Dynamics in Nature and Society*, vol. 2007, Article ID 28960, 17 pages, 2007.
- [5] Q. Zhu, J. Cao, and R. Rakkiyappan, "Exponential input-to-state stability of stochastic Cohen-Grossberg neural networks with mixed delays," *Nonlinear Dynamics*, vol. 79, no. 2, pp. 1085–1098, 2015.
- [6] Q. Zhu, R. Rakkiyappan, and A. Chandrasekar, "Stochastic stability of Markovian jump BAM neural networks with leakage delays and impulse control," *Neurocomputing*, vol. 136, pp. 136– 151, 2014.
- [7] Y. Shen and J. Wang, "Almost sure exponential stability of recurrent neural networks with Markovian switching," *IEEE Transactions on Neural Networks*, vol. 20, no. 5, pp. 840–855, 2009.
- [8] O.-M. Kwon, M.-J. Park, S.-M. Lee, J. H. Park, and E.-J. Cha, "Stability for neural networks with time-varying delays via some new approaches," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 2, pp. 181–193, 2013.
- [9] H. Zhang, Z. Liu, G.-B. Huang, and Z. Wang, "Novel weighting delay-based stability criteria for recurrent neural networks with time varying delay," *IEEE Transactions on Neural Networks*, vol. 21, no. 1, pp. 91–106, 2010.
- [10] J. Sun, G. P. Liu, J. Chen, and D. Rees, "Improved delay-rangedependent stability criteria for linear systems with time-varying delays," *Automatica*, vol. 46, no. 2, pp. 466–470, 2010.
- E. Fridman, U. Shaked, and K. Liu, "New conditions for delayderivative-dependent stability," *Automatica*, vol. 45, no. 11, pp. 2723–2727, 2009.
- [12] H. Yu, X. Yang, C. Wu, and Q. Zeng, "Stability analysis for delayed neural networks: reciprocally convex approach," *Mathematical Problems in Engineering*, vol. 2013, Article ID 639219, 12 pages, 2013.
- [13] P. G. Park, J. W. Ko, and C. Jeong, "Reciprocally convex approach to stability of systems with time-varying delays," *Automatica*, vol. 47, no. 1, pp. 235–238, 2011.
- [14] T. Li, X. Yang, P. Yang, and S. Fei, "New delay-variationdependent stability for neural networks with time-varying delay," *Neurocomputing*, vol. 101, pp. 361–369, 2013.
- [15] J.-H. Kim, "Note on stability of linear systems with time-varying delay," *Automatica*, vol. 47, no. 9, pp. 2118–2121, 2011.

- [16] H. G. Zhang, F. S. Yang, X. D. Liu, and Q. L. Zhang, "Stability analysis for neural networks with time-varying delay based on quadratic convex combination," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 4, pp. 513–521, 2013.
- [17] X.-M. Zhang and Q.-L. Han, "Global asymptotic stability analysis for delayed neural networks using a matrix-based quadratic convex approach," *Neural Networks*, vol. 54, pp. 57–69, 2014.
- [18] A. F. Filippov, Differential Equations with Discontinuous Righthand Sides, vol. 18 of Mathematics and its Applications, Kluwer Academic, Dordrecht, The Netherlands, 1988.
- [19] F. H. Clarke, Y. S. Ledyaev, R. J. Stern, and P. R. Wolenski, Nonsmooth Analysis and Control Theory, vol. 178 of Graduate Texts in Mathematics, Springer, New York, NY, USA, 1998.
- [20] J.-P. Aubin and A. Cellina, *Differential Inclusions*, Springer, Berlin, Germany, 1984.
- [21] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, SIAM, Philadelphia, Pa, USA, 1994.

Research Article

A Swarm Intelligent Algorithm Based Route Maintaining Protocol for Mobile Sink Wireless Sensor Networks

Xiaoming Wu,^{1,2,3} Yinglong Wang,^{1,2,3} and Yifan Hu^{2,3}

¹ College of Information Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
 ²Shandong Computer Science Center (National Supercomputer Center in Jinan), Jinan 250014, China
 ³Shandong Provincial Key Laboratory of Computer Network, Jinan 250014, China

Correspondence should be addressed to Yinglong Wang; wangyl@sdas.org

Received 2 June 2015; Revised 5 September 2015; Accepted 30 September 2015

Academic Editor: Yuan Fan

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

Recent studies have shown that mobile sink can be a solution to solve the problem that energy consumption of sensor nodes is not balanced in wireless sensor networks (WSNs). Caused by the sink mobility, the paths between the sensor nodes and the sink change frequently and have profound influence on the lifetime of WSN. It is necessary to design a protocol that can find efficient routings between the mobile sink and nodes but does not consume too many network resources. In this paper, we propose a swarm intelligent algorithm based route maintaining protocol to resolve this issue. The protocol utilizes the concentric ring mechanism to guide the route researching direction and adopts the optimal routing selection to maintain the data delivery route in mobile sink WSN. Using the immune based artificial bee colony (IABC) algorithm to optimize the forwarding path, the routing maintaining protocol could find an alternative routing path quickly and efficiently when the coordinate of sink is changed in WSN. The results of our extensive experiments demonstrate that our proposed route maintaining protocol is able to balance the network traffic load and prolong the network lifetime.

1. Introduction and Related Work

Wireless sensor network (WSN) is an intelligent monitoring self-organized network consisting of many microsensor nodes with the capabilities of communication, sensing, and computing deployed inside or around the monitoring area. It has broad application prospects and great application value in industrial and agricultural control, urban management, environmental testing, hazardous area remote control, and other fields. WSN involves multiple frontier research fields; it is considered as one of the top-ten world-changing technologies in the future [1].

A typical WSN is composed of many sensor nodes and one or several sinks. The sink collects data from the sensing environment. However, if the sink involved in collecting data is static, the sensors connected directly to a sink deplete their energy much faster than the rest of the network since they carry all the data gathered by the sensors [2], which is called "the crowded center effect" [3] or "energy hole problem" [4, 5]. One method to avoid the formation of energy holes is to use sink mobility. The sensor nodes can take turns to become the neighbors of the sink due to the sink mobility in a controlled manner, so the energy is consumed evenly among the nodes. It has been demonstrated that a mobile sink can potentially prolong the network's lifetime as the mobile sink would cause the sensor nodes to consume less energy [6].

Mobile sink can be a solution to solve the problem that energy consumption of nodes is not balanced in WSN; we call it mobile sink WSN (MSWSN); the network architecture is illustrated in Figure 1; the source nodes constantly deliver data through multihops relay node path to the mobile sink. But caused by the sink mobility, the paths between the sensor nodes and the sink would change with time [7]. Routing protocol in MSWSN is a great challenge due to the following reasons. Firstly, it is not easy to grasp the whole network topology and it is hard to find a routing path. Secondly, sensor nodes are tightly constrained in terms of energy, processing, and storage capacities. The unpredictable and constant changes in the sink's location form the obstacle of designing the route maintaining protocols in the energy



FIGURE 1: Mobile sink wireless sensor network model.

constrained WSN [8]. It is necessary to design a protocol that can find efficient routes between the mobile sink and sensor nodes but does not consume too many network resources.

Various routing protocols have been proposed to address the routing problem of mobile sink. In the literature [9], the TTDD concentrates on efficient data delivery to mobile sinks by clustering nodes into cells. Mobile sinks broadcast request packets only in their local cell, and an overlay routing scheme keeps track of the current cells of the sinks for routing data to them. While effective in high mobility scenarios, the overhead to build and maintain the overlay is significant, especially in periodic reporting scenarios, which are more traffic intensive than event-based reporting. Thus, TTDD is better suited to event-detecting sensor networks with sporadic rather than continuous traffic.

SEAD [10] and its developed protocol DEED [11] attempt to optimize routes from a single source to mobile sinks by allowing each sink to select an access sensor node. A data delivery tree is built between the source and all access nodes based on a geographic location heuristic. When the sink moves, a path between its current nearest neighbor and the access node is maintained, eliminating the need to rebuild the tree. However, if the sink moves far away, a new access node would be selected and the tree is rebuilt.

An enhanced real time with load distribution (ERTLD) routing protocol for mobile WSN is proposed in literature [12]. ERTLD utilizes corona mechanism and optimal forwarding metrics to forward the data packet in mobile WSN. It ensures high packet delivery ratio and experiences minimum end-to-end delay and enhances the total performance, reliability, and flexibility of data forwarding mechanism in mobile WSN.

However, in these classical routing protocols, the route maintaining scheme for the large scale sensor nodes has rarely been considered. Moreover, as the routing optimization problem to find the optimal routing is a NP-hardness problem, the heuristic deterministic methods always fall into local optimum and only get the approximate optimal result. In this paper, we propose a swarm intelligent algorithm optimized route maintaining protocol to optimize the routing path of the MSWSN, and this optimization algorithm would converge to the optimal resolution of the path. Our proposed scheme is developed on the basis of the corona mechanism in ERTLD and differs from the above works. The main contributions of this paper are as follows:

- (1) We utilize the concentric ring mechanism to guide the route researching direction and adopt self-adapt feature of our swarm intelligent routing protocol to deal with the dynamic route maintaining problem of MSWSN, which results in saving energy and maximizing packet delivery ratio.
- (2) We analyze and compare our protocol with TTDD [9] and ERTLD [12] in terms of packet delivery and energy consumption. Simulation results show superiority of the protocol.



FIGURE 2: The route maintaining model of MSWSN. (a) The process of delivering route maintaining packet through possible paths. (b) The concentric ring number of each node on the path.

The contributions of this paper are different from the literatures [8, 13], though the network models of them are all the WSNs with mobile sink:

- (1) The routing mechanism is different, we provide concentric ring mechanism to detect the relative distance and direction of each node to the mobile sink, in order to efficiently guide the route researching direction from source node to the sink, comparing to the routing researching mechanism without direction guidance in [13] and another different routing mechanism in [8].
- (2) The kernel optimization algorithm of routing protocol is different; our protocol adopts the artificial bee colony (ABC) algorithm to optimize the forwarding path, comparing to the particle swarm optimization algorithm in literature [13] and machine learningbased approach in literature [8]. The tests for several standard functions by [14] have shown that the performance of ABC is better than the other populationbased algorithms with the advantage of employing

fewer control parameters, such as PSO and other algorithms.

The rest of this paper is organized as follows. Section 2 formulates our network model and method. Section 3 describes the proposed algorithm in detail, and the routing protocol is analyzed. Section 4 evaluates the performance of the IABC by comparing it with other routing protocols. Finally, the conclusion is presented in Section 5.

2. Description of Network Model

2.1. Concentric Ring Mechanism. The MSWSN is modeled as a time-dependent connected graph G(V, E), where V is a finite set of sensor nodes and E is the set of edges representing connection between these nodes. The mobile sink is a rendezvous point and moves in the network area for the purpose of collecting information of nodes. In our model, the regular moving track of the mobile sink is the rectangular polyline track, as shown in Figure 1, which can efficiently traverse the whole network. The similar traverse track method has been used in the literature [15]. Figure 2 represents a certain time T_0 when the mobile sink has moved to place A in the network model. v_i (0 < i < N – 2) is the relay sensor node, v_s is the source node, and v_{sink} is the mobile sink. Suppose there exist m (0 < m < N - 1) source nodes and one mobile sink; these source nodes can route the packets through multihops relay node path to the mobile sink, as illustrated in the right blue dash-line box (b) of Figure 2. After deploying in the middle area of the network, the mobile sink would broadcast the concentric ring (CR) packets to its one-hop neighbors, in order to determine the concentric ring number for all sensor nodes in the network. The main fields of CR are CR_N (the ring number; initial number is zero) and CR_ID (the packet ID sent by sink). Then, the nodes forward the packet to the next-hop neighbors. The sink would not rebroadcast the CR packet until it moves to a new coordinate which is more than D meters away from the previous coordinate. In our assumption, if the moving distance of sink is less than D meters, it is still within the communication range of the closest relay node.

As shown in Figure 2, all circles are concentric at the sink and can form the coordinate system, and the width of each concentric ring is assumed to be equal to the sensor node's transmission range *R*. Therefore, each sensor node would belong to exactly one concentric ring. The data would be delivered from node v_i with high value of CR_N to node v_j with low value of CR_N and at last to the mobile sink. If there is not any candidate with lower number of CR_N than v_i in the neighbor table of v_i , data would be forwarded to the node with the same number of CR_N. The right blue box (b) of Figure 2 shows the CR_N of each node on the multihop path.

When the mobile sink moves to the random position, the coordinate system is also changed. Once the sink moves up to R meters away, it would rebroadcast the CR_N to the network, and each node's CR_N would be changed. The packet CR would be delivered to all one-hop neighbors. If one relay node receives CR, it would check CR_ID; if it is the same as the node's previous CR_ID, the node would discard the packet; if not, the node would increase CR_N in CR and save this value as its new concentric ring number. Then, the node would broadcast CR to its neighbors, until the nodes of the whole network are upgraded. With this method, the network can automatically respond to the dynamic topology. Once the sink moves to a certain distance, the previous scenario would be repeated. Using the concentric ring mechanism, the relative distance and direction of each node to the mobile sink can be detected, and the following routing research from source node can be guided toward the direction of sink.

Our network routing model is based on the following assumptions: (1) The area is covered by a large number of homogeneous sensor nodes. Sensor nodes are stationary, but sink moves and changes position constantly with a relatively fixed speed. (2) Data is sensed and transmitted from each source node to the mobile sink every T time period.

2.2. Route Maintaining Algorithm. The concentric ring mechanism can calculate the sensor node's ring number according to its distance to the sink. The following routing algorithm

can calculate and decide the optimal forwarding path from the source nodes to the mobile sink, in order to update and maintain the dynamic route.

As assumed, the path generated from a given source node v_s^1 to the mobile sink v_{sink} is denoted by p_j $(j \in 1, 2, ..., n)$. The *k*th relay node on path p_j is denoted by v_j^k , $\{k \in 1, 2, ..., h_j\}$, in which h_j is the hop count on path p_j . e_j^m represents the *m*th direct edge between two neighbor nodes on p_j . Let $N(p_j) = \{v_j^S, v_j^1, v_j^2, ..., v_j^k, ..., v_j^n\} \subset N(v)$ be the set of the sensor nodes existing along path p_j , where *k* represents the distance from the sink to the node on a hop scale. As shown in the right blue box (b) of Figure 2, path p_1 connects source v_s^1 with the sink and contains 6 relay nodes $\{v_1^1, v_1^2, v_1^3, v_1^4, v_1^5, v_1^6\}$, and 7 relay edges $\{e_1^1, e_1^2, e_1^3, e_1^4, e_1^5, e_1^6, e_1^7\}$. Each node *v* owns its neighbor table, which stores its surrounding neighbor node's ID v_i and other information (node signal strength RSSI(v_i) between the node and v_i).

The movement of mobile sink would firstly upgrade all the nodes' CN_R. Once the sink has moved out of the communication distance of relay node v_j^n and the data packet cannot be delivered to the sink, the routing failed information packet would be sent backward to the source node v_j^s through path p_j ; then, the route maintaining algorithm would be implemented.

 v_i^s would broadcast the route maintaining (RM) packets to one-hop neighbors, and the packet contains the information of battery voltage and source's CR_N (called CR_N_S). Our rule is that if the relay node's CR_N is not higher than the source's CR_N_S, it can relay RM packet, so RM can be delivered forward of the mobile sink. If the relay node v_i receives RM, it will check its own CR_N; if its CR_N is higher than CR_N_S in RM, v_i will discard the packet; if not, v_i will decrease and save CR_N_S in RM and relay RM to its onehop neighbors; the packet would contain information of its $RSSI(v_i)$, $Delay(v_i)$, and its battery voltage $V_{bat}(v_i)$. Then, the node will relay RM to the other neighbors, until the data packet RM is delivered through any multihop path p_i (0 < j < n) to the mobile sink. Thus, the sink would receive several packets RM from these paths, and all the paths delivering RM to the sink are considered as the possible alternative paths set P_{all} . The process of delivering packet RM through multihop paths is illustrated in the left red dash-line box (a) of Figure 2.

With this method, the optimal path p_{op} with the suitable relay node sequence $\{v_{op}^{S}, v_{op}^{1}, v_{op}^{2}, \ldots, v_{op}^{k}, \ldots, v_{op}^{n}\}$ is among these possible alternative paths set P_{all} . The sink collects all the information and extracts $RSSI(v_i)$, $Delay(v_i)$, and $V_{bat}(v_i)$ of these nodes on the possible alternative paths P_{all} and calculates the fitness value of path and selects one optimal alternative path p_{op} with optimal fitness fitness(p_{op}) (e.g., path $\{v_s^{1}, v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ in left red box (a) of Figure 2). The nodes sequence of a possible alternative path p_j is $\{v_j^{S}, v_1^{1}, v_j^{2}, \ldots, v_j^{k}, \ldots, v_j^{n}\}$, and the factors affecting the choice of p_j include the

nodes' signal strength, end-to-end delay, and battery voltage on the path. These parameters can determine the fitness function of p_i , fitness (p_i) :

fitness
$$(p_j) = \max\left(\lambda_1 \sum_{i=1}^n \frac{\text{RSSI}_{\text{th}}}{\text{RSSI}(v_j^i)} + \lambda_2 \sum_{i=1}^n \frac{\text{Delay}_{\text{th}} - \text{Delay}(v_j^i)}{\text{Delay}_{\text{th}}} + \lambda_3 \sum_{i=1}^n \frac{V_{\text{bat}}(v_j^i)}{V_{\text{batth}}}\right),$$
 (1)

where $\text{RSSI}(v_j^i)$ is the signal strength of v_j^i from the nexthop node v_j^{i-1} and its value can indirectly reflect the distance between v_j^i and v_j^{i-1} . RSSI_{th} is the signal strength value at threshold point 1 m which is -45 dBm. $\text{Delay}(v_j^i)$ is the endto-end delay between the node and next-hop node; Delay_{th} is the end-to-end delay threshold which is set to 250 ms. $V_{\text{bat}}(v_j^i)$ is the node battery voltage; its value can indirectly reflect the remaining energy of (v_j^i) . V_{batth} is the node threshold battery voltage and equals $3.5 \text{ v. } \lambda_1, \lambda_2$, and λ_3 are the weights of remaining energy, delay, and distance constraints in the fitness function, respectively, and $\lambda_1 + \lambda_2 + \lambda_3 = 1$. We set $\lambda_1 = 0.4, \lambda_2 = 0.2$, and $\lambda_3 = 0.4$. The higher fitness value indicates the more suitable routing path; therefore, the best path p_b with the optimal fitness fitness(p_b) will be selected.

After that if the number of possible alternative paths in the set P_{all} is more than 5, the selection of calculated possible alternative paths would be so complex, and we would use the swarm intelligent algorithm to optimize the selection of optimal path, or else the optimal path would be calculated and selected directly. We consider each possible alternative path p_j from source v_s^i to v_{sink} as a solution, and the number of solutions in the population is *m*. The details of swarm intelligent algorithm are described in the next section.

3. Design of the Protocol in MSWSN

3.1. The Swarm Intelligent Algorithm of Route Maintaining Protocol. The kernel optimization algorithm of our proposed route maintaining protocol is a swarm intelligent optimization algorithm. This optimization algorithm would compute and select the optimized path from source node to the sink among all the possible paths, in which each possible path represents a solution. As new avenues in the field of optimization, swarm intelligence was defined by Bonabeau as the attempt to design algorithms or distributed problemsolving devices inspired by collective behavior of social insect colonies and other animal societies [16]. Common examples range from flock of birds to colony of bees. The main essence of swarm intelligence is including the functional behavior of these group agents into a practical computational model [13], such as artificial bee colony (ABC) algorithm.

The population-based artificial bee colony (ABC) algorithm proposed by Okdem et al. is based on the minimal foraging model of honey bees used in nectar collection from the adjoining environment of their honeycomb [17], with the behavior like self-organization, task allocation, and communication among the individuals. Its advantages of fewer parameter settings, faster convergence speed, and higher convergence precision have attracted the attention of many scholars since it was proposed. The algorithm has been successfully applied in the function optimization problem, WSN, and other areas [18, 19]. The tests for several standard functions by [14] have demonstrated that the performance of ABC is better than the mainstream optimization algorithms, such as GA, PSO, and DE algorithm.

In the ABC, bees are categorized into three groups: employed bees, onlooker bees, and scout bees. The employed bees search food sources and share the information to recruit the onlooker bees. The onlooker bees make decision to choose a food source from those found by the employed bees and search food around it. The food source that has more nectar amount (fitness value) would have a higher probability to be selected by onlooker bees. The scout bees are translated from a few employed bees, which discard their food sources and randomly search new ones. For a search problem in a Ddimensional space, the position of a food source represents a potential solution. The nectar amount of a food source is the fitness value of the associate solution. Each food source is exploited by only one employed bee. The number of employed or onlooker bees is equal to the number of solutions in the population.

The ABC can use the positive feedback mechanisms of optimized search between bees to effectively speed up the process of global optimization and set fewer parameters. But when searching in the near global optimal solution, the search speed would slow down and the diversity of population would be reduced. Therefore, we draw on outstanding diversity characteristic of immune mechanism and develop the IABC algorithm. Each solution would be considered as an antibody. After the antibody clone and selection step, the better one is reserved and the worse one is discarded, which would increase its diversity.

3.1.1. Artificial Bee Colony Mechanism. $X_i = x_{i1}, x_{i2}, ..., x_{iD}$ is the *i*th food source (solution) in the population, and *D* is the problem dimension size. Each employed bee generates a new food source V_i around the neighborhood of its previous food source position as follows:

$$v_{ij} = x_{ij} + \phi_{ij} \left(x_{ij} - x_{kj} \right),$$
(2)

where i = 1, 2, 3, ..., SN, SN is the population size, j = 1, 2, ..., D is a random index, and x_k is a randomly selected solution in the current population $(k \neq i)$. ϕ_{ij} is a random number in the range [-1, 1]. If the new V_i is better than its parent X_i , then V_i replaces X_i .

After employed bees phase, the probability value Pro_i is calculated according to the food sources fitness as the following:

$$Pro_{i} = \frac{fitness(x_{i})}{\sum_{i=1}^{SN} fitness(x_{i})},$$
(3)

where fitness(x_i) is the fitness value of the *i*th solution in the population and Pro_i is proportional to fitness(x_i). A better food source has higher probability to be selected.

$$x_{ij} = x_j^{\min} + \text{Rand}(0, 1) \left(x_j^{\max} - x_j^{\min} \right),$$
 (4)

where Rand(0, 1) is uniformly distributed in the range [0, 1] and $[x_j^{\min}, x_j^{\max}]$ is the boundary constraint for the *j*th variable. Then, if the fitness value of the solution is the optimal fitness or the number of iterations increases from zero to Gen, the optimal path p_{op} (solution) would be output, or else go to the immunization step.

3.1.2. Immunization Mechanism. In this step, each solution would be considered as an antibody. The sequence number SN of solution x_i arranged in the optimal solution set X_b is considered as the affinity of solution SA_i, and SA_i = SN. Then, the clone number CN_i is calculated as follows:

$$CN_i = SA_i \times \frac{N_p}{\left[\sum_{j=1}^m j\right]},$$
(5)

where N_p is the solution number, *m* is the size of optimal solution set X_b , and the total cloned solution number is Sum = $\sum_{j=1}^{m} CN_i$. Thus, the cloned solution number is proportional to the fitness. Then, the solution mutation is used in the clone populations and the mutation rule is as follows:

$$CS_i = x_i + \gamma Rand(0, 1), \qquad (6)$$

where x_i is the original antibody, γ represents mutation factors and $\gamma = 0.5$, and CS_i is the clone solution individual. In the solution restrain rule, we calculate the antigen stimulus degree of solution in $N(X_b)$ and the mutation solution. The Euclidean distance between solution $N(CS_t)$ and antigen

(fitness) fitness(x_t) is $D(i, j) = \sqrt{\sum_{i=1}^{n} (CS_{it} - fitness(x_{it}))^2}$.

Therefore, the stimulus degree of antibody (solution) is

$$SD(i, j) = \frac{1}{D(i, j)} = \frac{1}{\sqrt{\sum_{i=1}^{n} \left(CS_{it} - fitness(x_{it}) \right)^2}}.$$
 (7)

After comparing each solution with the stimulus threshold Th, the better solution (SD(i, j) > Th) would be reserved in the memory cell, and the worse one is discarded. The detailed process of the proposed IABC algorithm is shown in Figure 3.

The IABC optimization algorithm is the kernel algorithm of our route maintaining protocol of the MSWSN. To deal with the path maintaining problem due to the movement of sink, the proposed algorithm would optimize the path fitness function to provide the fast routing recovery mechanism with an alternative optimal-fitness path. Apparently, the more suitable alternative path selected from source to the mobile sink would contain nodes with stronger signal strength, less end-to-end delay, and higher battery voltage. 3.2. Assumption of Energy Model. The calculation of minimum energy consumption emphasizing the effect of distances will be as in (8) and (9) expressing sum of the energy consumptions of network [9]. The abbreviations j, Ene_j, Ene^{TX}, Ene^{RX}, Ene^{elec}, Ene^{amp}, k, and d in (9) are node index, energy consumption of the *j*th node, transmit energy, receive energy, radio electronics parameter, transmit amplifier parameter, number of bits of the transmitting data, and distance value between *j*th node and next-hop node, respectively:

$$\operatorname{Ene}_{j} \ge \left(\operatorname{Ene}^{\operatorname{RX}} + \operatorname{Ene}^{\operatorname{TX}}\right)$$
 (8)

$$Ene^{RX} = Ene^{elec} \cdot k,$$

$$Ene^{TX} = \left(Ene^{elec} + Ene^{amp} \cdot d_j^2\right) \cdot k.$$
(9)

Using this method, the total energy consumption of the data transmission and executing the proposed IABC per round can be calculated in the simulation.

4. Experimental Evaluation of the Route Maintaining Protocol

4.1. Model and Assumption. Our system uses MATLAB 2008a to simulate and evaluate the performance of the protocol. The experimental hardware environments are Intel i7-4600M, 2.90 GHz CPU, and 4 GB memory, and the operating system is MS windows 7. The whole MSWSN is simulated in the area of $3500 \text{ m} \times 3500 \text{ m}$. The field is static and 200~300 sensor nodes are deployed uniformly in which 10% sensor nodes are source nodes. The sensor nodes are homogeneous and have the same initial energy of 120 J. Their communication radius is 300 m. This experiment compares our protocol with TTDD, ERTLD routing protocols for MSWSN. The purpose of the simulation is to illustrate that our protocol could provide a more robust and efficient transmission environment.

The other network environment parameters are as follows: One mobile sink moves in the network and its speed is $3 \sim 6 \text{ m/s}$, the source node delivers packets at the rate of 20 data packets per round, with 10 KB of each packet size, and the simulation lasts for 600 rounds. In the energy model, Ene^{elec} = 40 nJ/bit and Ene^{amp} = 60 nJ/bit. The sink is assumed to provide sufficient energy to receive data from nodes and operate our protocol. The values of parameters used for the IABC are function dimension D = 20 and iterated generation Gen = 150.

A snapshot from the source node to the mobile sink during the network simulation is shown in Figure 4. We can see that when the sink moves from A to a new coordinate B, the source node immediately establishes an optimal alternative path (path 1-2-3-8-9-10-11-B in Figure 4) to reach the sink, so as to replace the previous broken path (path 1-2-3-4-5-6-7-A in Figure 4).

4.2. Evaluation of the Experimental Results. The performance metrics used for the comparison are packet delivery ratio



FIGURE 3: The flowchart of the IABC algorithm for route maintaining.



FIGURE 4: The snapshot of network routing simulation with our protocol.



FIGURE 5: Comparison of packet delivery ratio between the three protocols with different node number: (a) 200 sensor nodes and (b) 300 sensor nodes.

(the ratio between the successfully received data packets at the sink and the successfully sent data packets by the source node), energy expenditure ratio (the ratio between the consumed energy of nodes and initial energy of these nodes), and average end-to-end delay (the difference between the time a packet is received by the sink and the time it was originally sent by the source node). The results of packet delivery ratio and energy expenditure ratio are normalized, which is helpful to compare their performances.

In terms of packet delivery ratio in Figures 5(a) and 5(b), it can be clearly seen that the delivery ratio for all protocols drops as the number of simulation rounds is increased. This is because the mobility of sink affects the quality of the selected links of the path; the process of links repair and loss of packet would reduce the packet delivery ratio. Among the selected protocols, IABC has the highest packet delivery rate and ERTLD achieves the second one. This is due to several factors: data traffic is routed along shorter paths by using concentric ring mechanism and IABC optimizing algorithm, which would reduce the packet loss rate; the intelligent swarm optimization mechanism keeps the route update faster and more efficiently, with its fast responsiveness to the changing sink position.

In our second experiment presented in Figures 6(a) and 6(b), we vary the velocity of a mobile sink from 3 m/s to 6 m/s. The energy expenditure ratios of all protocols are increased with an increasing number of simulation rounds. This ratio also increases as the mobile sink moves faster, because the change of the frequent topology will incur heavier communication overhead. IABC has lower energy expenditure, followed by ERTLD and finally TTDD. IABC performs slightly better due to its concentric ring mechanism,

which can guide the route researching direction and reduce communication overhead of nodes, and also due to the intelligent routing optimization mechanism which enables faster recovery of routes and reduces the energy consumption of protocol. The delivery rate trend in the case of higher velocity of mobile sink is also as expected, dropping with higher velocities. This is due to the fact that nodes would consume more energy to search the route to the sink when they moves faster away from their transmission radius. Notably, the appropriate speed of the sink will be needed for all protocols, which can better reflect the performance of the routing protocols.

In the third experiment, the smaller average end-to-end delay means the faster data transmission. We can observe in Figures 7(a) and 7(b) that the IABC outperforms the ERTLD and TTDD in terms of average delay. It can be explained by the faster node communication routing and shorter alternative path selection of the proposed IABC for route maintaining, which can balance the network traffic load and prolong the network lifetime. The delay curve value of IABC almost has not changed as the number of simulation rounds increases, because our protocol would always choose the optimal path with the stable routing delay. Comparing with Figures 7(a) and 7(b), as the more sensor nodes number indicates the longer delivery length of the path, thus the advantage of our IABC with the better alternative path selection is demonstrated more obviously, which means that our routing protocol is more suitable for deploying in the large scale networks with mobile sink.

In summary, these experiments demonstrate clearly the routing optimization ability of IABC and its intelligent optimization mechanism to quickly identify routes to mobile



FIGURE 6: Comparison of energy expenditure between the three protocols as a function of different node speeds: (a) sink speed 3 m/s and (b) sink speed 6 m/s.



FIGURE 7: Comparison of end-to-end delay between the three protocols with different node number: (a) 200 sensor nodes and (b) 300 sensor nodes.

sink. Compared to the other routing protocols, it consumes significantly less energy, needs less end-to-end delay, and achieves considerably higher delivery rates.

5. Conclusion

This study presents a novel route maintaining protocol based on the IABC for the MSWSN. In the proposed protocol, the concentric ring mechanism is utilized to guide the route researching direction, and the optimal routing selection is adopted to preserve the data delivery route in the network. Using the immune based artificial bee colony (IABC) algorithm to optimize the forwarding path, the protocol could find an alternative routing path quickly and efficiently when the coordinate of sink is changed in MSWSN. More importantly, this paper demonstrated the applicability and the potential of IABC algorithm for solving routing optimization problems. The results of our extending performance are compared to the other aforementioned routing protocols in terms of energy, packet delivery, and delay. Our proposed route maintaining protocol could efficiently solve the energy hole problem, balance the network traffic load, and maintain the network robustness against topology changes.

In the future we will focus on improving the convergence performance, reducing the computational complexity of the IABC algorithm, and validating the proposed protocol on different scenarios with various movement trajectories of mobile sink, and the most important optimization objective is maximizing the network lifetime. In addition, the nodes would have GPS to locate themselves, and the sink may broadcast its position instead of CR packets to build the concentric ring mechanism.

Conflict of Interests

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

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (nos. 61401257 and 61501282), Shandong Province Young and Middle-Aged Scientists Research Awards Fund (nos. BS2013DX021 and BS2013DX019), Shandong Academy Young Scientists Fund Project (no. 2013QN037), Shandong Academy of Science Doctoral Fund (no. [2012] 58), Shandong Academy Basic Research Fund, and Shandong Academy of Sciences Pilot Project for Science and Technology.

References

- H. W. Ding, D. F. Zhao, Y. S. Gao, and N. S. Zhao, "Research progress of wireless sensor network," *Experiment Science & Technology*, vol. 9, no. 6, pp. 26–29, 2011.
- [2] Y. S. Jiang, W. R. Shi, X. G. Wang, and H. B. Li, "A distributed routing for wireless sensor networks with mobile sink based on the greedy embedding," *Ad Hoc Networks*, vol. 20, pp. 150–162, 2014.
- [3] A. Jawahar, S. Radha, and S. Vadivelan, "Connectivityguaranteed hybrid topology management scheme for improving the operational lifetime of 3-dimensional wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2010, Article ID 547368, 11 pages, 2010.
- [4] Y. Xue, X. M. Chang, S. M. Zhong, and Y. Zhuang, "An efficient energy hole alleviating algorithm for wireless sensor networks," *IEEE Transactions on Consumer Electronics*, vol. 60, no. 3, pp. 347–355, 2014.
- [5] J. Wang, X. Q. Yang, Z. Q. Zhang, B. Li, and J.-U. Kim, "A survey about routing protocols with mobile sink for wireless sensor network," *International Journal of Future Generation Communication and Networking*, vol. 7, no. 5, pp. 221–228, 2014.
- [6] K. Karenos and V. Kalogeraki, "Traffic management in sensor networks with a mobile sink," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 10, pp. 1515–1530, 2010.
- [7] Y.-F. Hu, Y.-S. Ding, L.-H. Ren, K.-R. Hao, and H. Han, "An endocrine cooperative particle swarm optimization algorithm

for routing recovery problem of wireless sensor networks with multiple mobile sinks," *Information Sciences*, vol. 300, pp. 100–113, 2015.

- [8] A. Förster and A. L. Murphy, "Froms: a failure tolerant and mobility enabled multicast routing paradigm with reinforcement learning for WSNs," *Ad Hoc Networks*, vol. 9, no. 5, pp. 940–965, 2011.
- [9] H. Luo, F. Ye, J. Cheng, S. Lu, and L. Zhang, "TTDD: twotier data dissemination in large-scale wireless sensor networks," *Wireless Networks*, vol. 11, no. 1-2, pp. 161–175, 2005.
- [10] S. Tian, S. M. Shatz, Y. Yu, and J. Li, "Querying sensor networks using Ad Hoc mobile devices: a two-layer networking approach," *Ad Hoc Networks*, vol. 7, no. 5, pp. 1014–1034, 2009.
- [11] H. S. Kim, T. F. Abdelzaher, and W. H. Kwon, "Dynamic delayconstrained minimum-energy dissemination in wireless sensor networks," ACM Transactions on Embedded Computing Systems, vol. 4, no. 3, pp. 679–706, 2005.
- [12] A. Ali Ahmed, "An enhanced real-time routing protocol with load distribution for mobile wireless sensor networks," *Computer Networks*, vol. 57, no. 6, pp. 1459–1473, 2013.
- [13] Y. F. Hu, Y. S. Ding, K. R. Hao, L. H. Ren, and H. Han, "An immune orthogonal learning particle swarm optimisation algorithm for routing recovery of wireless sensor networks with mobile sink," *International Journal of Systems Science*, vol. 45, no. 3, pp. 337–350, 2014.
- [14] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," *Applied Mathematics and Computation*, vol. 214, no. 1, pp. 108–132, 2009.
- [15] D.-T. Ho, J. Park, and S. Shimamoto, "Performance evaluation of the PFSC based MAC protocol for WSN employing UAV in rician fading," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '11)*, pp. 55–60, Cancún, Mexico, March 2011.
- [16] G. J. Zhang and J. W. Hu, "Adaptive distributed gateway discovery with swarm intelligence in hybrid wireless networks," *International Journal of Future Generation Communication and Networking*, vol. 7, no. 5, pp. 137–150, 2014.
- [17] S. Okdem, D. Karaboga, and C. Ozturk, "An application of wireless sensor network routing based on Artificial Bee Colony algorithm," in *Proceedings of the IEEE Congress of Evolutionary Computation (CEC '11)*, pp. 326–330, New Orleans, La, USA, June 2011.
- [18] R.-C. Chen, W.-L. Chang, C.-F. Shieh, and C. C. Zou, "Using hybrid artificial bee colony algorithm to extend wireless sensor network lifetime," in *Proceedings of the 3rd International Conference on Innovations in Bio-Inspired Computing and Applications* (*IBICA '12*), pp. 156–161, IEEE, Kaohsiung, Taiwan, September 2012.
- [19] M. Saleem, G. A. D. Caro, and M. Farooq, "Swarm intelligence based routing protocol for wireless sensor networks: survey and future directions," *Information Sciences*, vol. 181, no. 20, pp. 4597–4624, 2011.

Research Article Graph-Analytical Method of Determining Impedance in Electrical Transformers

Zorica Bogicevic,¹ Slobodan Bjelić,² Petar Spalević,² and Milan Mišić¹

¹Higher Technical Professional School in Zvečan, Nušićeva 6, 38227 Zvečan, Serbia
 ²Faculty of Technical Sciences, University of Priština, Kneza Miloša 7, 38220 Kosovska Mitrovica, Serbia

Correspondence should be addressed to Zorica Bogicevic; zocabogi@gmail.com and Slobodan Bjelić; slobodanbjelic49@yahoo.com

Received 11 June 2015; Accepted 6 September 2015

Academic Editor: Hou-Sheng Su

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

This paper presents a graph-analytical method for determining the electrical impedance of alternate energy sources, especially small power transformers and current transformers in electric networks. Unlike conventional short-circuit and idle tests, according to proposed method, in this paper, transformer parameters are determined in a new way, which is based on measurement of voltages and currents on the active and reactive load (inductive or capacitive). The effectiveness of the proposed model was verified using an adapted simulation in the software package MATLAB Simulink. The simulation was performed for three types of ABB transformers with a 100% load. Simulation results were obtained for power transformers: $S_{n1} = 1$ [MVA], $S_{n2} = 2$ [MVA], $S_{n3} = 3.15$ [MVA]. If we compare measurement result values of R_T , X_T , which are contained in a brochure for ABB transformers and those obtained through simulation, different tolerances are obtained. For reactance results, deviations are up to 20% for all three tested transformers. For results of active resistance tolerances are up to 5% for all three tested transformers. This method can be used not only to determine the active and inductive AC power source parameters but also to determine and analyze the impendence of electrical sources with high frequencies.

1. Introduction

In addition to classical methods, one of the possible methods for obtaining the values and parameters of the transformer is graph-analytical method based on the construction of phase diagrams for different types of transformer loads.

Three-phase two-coiled transformer (Figure 1(a)) has HV/LV coils in coupling Dy11 (or Yy11) [1]. In the delta or star coupling HV transformers' coils, phase voltages have a sinusoidal shape and the same form of voltage that has the network from which the transformer is supplied. In symmetrical regime, the analysis of three-phase transformers is reduced to the analysis of the schemes of equivalent single-phase transformers. Equivalent single-phase diagram gives a picture of the three-phase system only if the system in all three phases has equal impedances Z, through the same modules flow as the same voltage modules V act on impedance. Single-phase transformer with two windings

corresponds to equations where secondary values are reduced to the primary side (Figure 1(f)) [1]:

$$\dot{V}' = -\dot{E}' + Z'\dot{I}',$$

 $-\dot{E}' = -\dot{E}_1'' = Z_0\dot{I}_0 = Z_\mu\dot{I}_\mu.$
(1)

The electrical circuit secondary values must be expressed in a way that the primary voltage refers to the primary current. At the beginning the current \dot{I}' must be determined in relation to ems \dot{E}' and to the parameters of the circuit:

$$\dot{I}' = \dot{I}_0 - \dot{I}_1'' = -\frac{\dot{E}}{Z_0} + \frac{-\dot{E}'}{Z_1'' + Z_{1\,\text{load}}''},$$

$$-\dot{E}' = \frac{\dot{I}'}{1/Z_0 + 1/(Z_1'' + Z_{1\,\text{load}}'')}.$$
(2)



(g)

FIGURE 1: Measurement graph-analytical transformer model for (a) *Dy*11 grope, (b) no-load, (c) short-circuit, (d) load, (e) high frequencies, (f) single load transformer, (g) and equivalent circuit of transformer.

By replacing the previous dependencies into the voltage equation $\dot{V}' = -\dot{E}' + Z'\dot{I}'$ the following is obtained:

$$\dot{V}' = \dot{I} \left[Z' + \frac{1}{1/Z_0 + 1/(Z_1'' + Z_{1\,\text{load}}'')} \right] = \dot{I}' Z_{\text{eq}}, \quad (3)$$

$$Z_{\rm eq} = Z' + \frac{1}{1/Z_0 + 1/(Z_1'' + Z_{\rm 1\,load}'')}.$$
(4)

From relation (4) it is clear that equivalent circuit of the transformer, where the primary current I' passes, must have equivalent impedance Z_{eq} . This impedance is presented in the circuit and is described in Figure 1, where the series connection Z' with the parallel combination $Z_0 = Z_{\mu}$ and $(Z_1'' + Z_{1 \text{ load}}'')$ is shown and parameters of transformers, which are included with losses, are in classical-conventional manner measured in the experiment of idling (no-load)

and in the short-circuit test. For small power transformers, due to difficulties in measuring active losses this method is practically not applicable.

A classic method (Section 2) and the proposed-analytical method (Section 3) are shown in this paper. The model proposed in this paper is verified by simulation in MATLAB Simulink software package (Section 4). In the adapted simulation model for three different ABB transformers of low power, tabular and graphical results are obtained (Section 5).

2. Measurement of Transformer's Impedance

2.1. The Short-Circuit Test. The equations of transformer when it is not loaded (Figure 1(b)), for defined primary impedance Z', can be formulated on the basis of the general system of [1, 2]

$$\begin{split} \dot{V}' &= -\dot{E}' + Z'\dot{I}', \\ \dot{V}'' &= \dot{E}'' - Z''\dot{I}'' \\ &\implies \dot{V}_1'' &= \dot{E}'' - Z_1''\dot{I}_1'', \quad \dot{I}' &= \dot{I}_0 - \dot{I}_1''. \end{split}$$
(5)

If the impedance load is great $(Z_{\text{load}} \rightarrow \infty)$, the value of secondary current is equal to zero:

$$I'' = 0,$$

$$\dot{I}_{1}'' = 0,$$

$$\dot{V}' = -\dot{E}' + Z'\dot{I}' = \dot{I}'(Z' + Z_{0}),$$
 (6)

$$\dot{V}_{1}'' = -\dot{E}_{1}'' = -\dot{E}' = Z_{0}\dot{I}_{0},$$

$$\dot{I}' = \dot{I}_{0}.$$

The medium values of phase currents and medium voltage values are obtained by three-phase transformer in idling current:

$$\dot{I}_{0} = \frac{(I_{A} + I_{B} + I_{C})}{3},$$

$$\dot{V}' = \frac{(V_{A} + V_{B} + V_{C})}{3}.$$
(7)

Based on the values $I_0 i V' i$ researcher can determine idling power factor (cos φ_0). The sizes of the transformer which can be determined from experiments of idling at rated voltage are as follows.

Transmission ratio of the transformer, which is the ratio of secondary and primary rated voltage at idle, is as follows:

$$K_{n} = \frac{N''}{N'} = \frac{E_{R}''}{E_{R}'} \approx \frac{V_{R}''}{V_{R}'}.$$
(8)

Idle current $I_{0,oc}$, idle current and its relative value, as a part of rated primary current in p.u. system, are as follows:

$$\dot{I}_{0,\text{oc,pu}} = \frac{I_{0,\text{oc}}}{I'_R}.$$
 (9)

The mutual impedance, defined when $Z' \approx 0$, is as follows:

$$|Z_0| \approx \frac{V_R'}{I_{0,\text{oc}}} = |Z_{0,\text{oc}}|.$$
 (10)

Its active component is obtained as

$$R_0 = \frac{P_{0,\text{oc}}}{3I_{0,\text{oc}}^2} = |Z_0| \cos \varphi_0, \tag{11}$$

and reactive component-reactance is obtained as

$$X_0 = \sqrt{Z_0^2 - R_0^2} = |Z_0| \sin \varphi_0 \approx |Z_0|.$$
 (12)

2.2. Losses in Idling. When the voltage is equal to the nominal $V' = V'_R$ practically do not differ from losses in the magnetic circuit (core), $P_{\text{core,oc}}$, because the losses at primer's copper, $P'_{\text{Cu,oc}} = 3I^2_{0,oc}R'$, in those conditions are small, since the current $I_{0,oc}$ has low value. The magnetization characteristic of any magnetic circuit made by ferromagnetic material contains information about the useful flux saturation degree [3, 4]. In the paper magnetic has value of residual flux since there is no load (provided that it is V' predominantly unchanged) [5, 6]. However, at rated voltage, the losses in the core P_{core} are about the same as losses at idle, $P_{\text{core,oc}}$, that is, total idle losses: $P_{o,oc}$, $P_{\text{core}} = P_{\text{core,oc}} = P_0$. Dangerous transient currents of the transformer can occur if a short-circuit on the secondary side happened at rated voltage $V' = V'_R$ [7].

2.3. The Short-Circuit Test. Secondary side of transformer is short-circuited (Figure 1(c)) [2] and then the value of the impedance load ($Z_{11oad}'' \approx 0$) and secondary voltage $V_1'' \approx 0$ are also equal to zero (in a three-phase transformer, all secondary ends are short-circuited to obtain a balanced short circuit). The equations for the transformer for short-circuit experiment are derived from the general system of

$$\dot{V}' = -\dot{E}' + Z'\dot{I}',$$

$$-\dot{E}' = -\dot{E}''_{1} = -Z''_{1}\dot{I}''_{1} = Z_{0}\dot{I}_{0} = Z_{\mu}\dot{I}_{\mu},$$

$$\left(Z_{0} = \infty \longleftrightarrow \dot{I}_{0} = 0\right), \quad \dot{I}' = \dot{I}_{0} - \dot{I}''_{1}.$$

(13)

Application of that system of equations on the equivalent circuit in Figure 1(g) [1, 2] for short-circuit scheme can be determined by primary I' and secondary $-I''_1$ current, magnetizing current I_0 and the common ems short-circuit -E'. One has

$$\begin{split} \dot{I}' &= \frac{\dot{V}'}{Z' + Z_1'' Z_0 / (Z_1'' + Z_0'')} = \frac{\dot{V}'}{Z_{\rm sc}} \approx \frac{\dot{V}'}{Z' + Z_1''}, \\ -\dot{I}_1'' &= \frac{\dot{I}' Z_0}{Z_1'' + Z_0} \approx \dot{I}', \end{split}$$

$$\dot{I}_{0} = \dot{I}' \frac{Z_{1}''}{Z_{1}'' + Z_{0}} \approx \frac{\dot{V}'}{Z_{0}} \frac{Z_{1}''}{Z_{sc}} = \dot{I}_{0,oc} \frac{Z_{1}''}{Z_{sc}},$$

$$-\dot{E}' = Z_{0} \dot{I}_{0} = Z_{0} \dot{I}_{0,oc} \frac{Z_{1}''}{Z_{sc}} = \dot{V}' \frac{Z_{1}''}{Z_{sc}},$$

(14)

where

$$Z_{\rm sc} = Z' + \frac{Z_1'' Z_0}{Z_1'' + Z_0} \approx Z' + Z_1''.$$
(15)

If $Z_0 \gg Z_1''$ is impedance, its active resistance and reactance of the transformer are

$$Z_{\rm sc} \approx Z' + Z_1'' = R_{\rm sc} + jX_{\rm sc} = Z_T.$$
 (16)

Impedance of transformer of the short-circuited secondary is reduced on a side of supply-line network.

The corresponding phase diagram is shown in Figures 1(c), 1(d), and 1(e) [2]. As it is seen from this diagram, shortcircuit voltage $V' = V_{sc} = Z_{sc}I'$ is triangle's hypotenuse and catheters are active voltage $V_{sc,R} = R_{sc}I'$ and reactive voltage $V_{sc,R} = jX_{sc}I'$. The angle of voltage φ_{sc} (or impedance, called short-circuit angle) triangle that graphically displays [8, 9] the short-circuit triangle and is $\varphi_{sc} = \arctan(X_{sc}/R_{sc})$. By adjusting $Z' \approx Z_1'' = Z_{sc}/2$ we will get the simpler expressions for the current magnetizing and short-circuit ems:

$$\dot{I}' = -\dot{I}_{1}'' = \frac{\dot{V}'}{Z_{sc}},$$

$$\dot{I}_{0} = \frac{\dot{V}'}{2Z_{0}} = \frac{I_{0,oc}}{2},$$

$$\dot{I} = \frac{\dot{V}'}{2Z_{0}},$$
 (17)

$$-\dot{E}' = Z_0 \frac{I_{0,\text{oc}}}{2} = \frac{V}{2}.$$

Possible equivalent scheme and short-circuit diagram are presented in Figure 1(c). As with experiments of idling it is not necessary to have any particular active load or higher voltage source [10]. If the frequency has a nominal value of $f = f_{\text{RATED}}$, the same values are read as in the experiment of idling, primary current, and power such that a transformer takes P_{sc} .

By reading of values I', P_{sc} , and $\cos \varphi_{sc}$ which are entered into the diagram as phase voltage functions V a diagram is obtained which is used for graphic determination of V'_{sc} , P_{sc} , and $\cos \varphi_{sc}$ in relation to the primary current values $I' = I'_R$.

Power factor is determined from the values I' and V' as $\cos \varphi_{sc} = P_{sc}/3V'I'$.

Short-circuit test gives the following values of the transformer in relation to rated current values. The impedance of short-circuit and its active and reactive component from the expression is as follows:

$$|Z_{sc}| = \frac{V'_{sc}}{I'_R},$$

$$R_{sc} = \frac{P_{sc}}{3I'^2_R} = |Z_{sc}| \cos \varphi_{sc},$$

$$X_{sc} = \sqrt{Z^2_{sc} - R^2_{sc}} = |Z_{sc}| \sin \varphi_{sc}.$$
(18)

Active component is a sum of resistance of the two coils $R_{\rm sc} = R' + R''_1 = R_T$. During short-circuit test, it should be stressed at which temperature θ the measurement is performed, $R_{\rm sc}$, where the temperature is set at around 75°C, where $R_{\rm sc75} = R_{\rm sc}[1 + 0.004(75 - \theta)]$. Reactive component, $X_{\rm sc} = X' + X''_1 = X_T$, is calculated from the sum of inductive scattering, independent currents which are flowing through respective coils [11]. From the same reasons $X_{\rm sc} = X_T$ is an independent value in accordance with the tested currents [12]. Impedance and power factor in the short-circuit are set for the temperature of 75°C:

$$\begin{aligned} \left| Z_{\text{sc},75^{\circ}\text{C}} \right| &= \sqrt{R_{\text{sc},75}^{2} + X_{\text{sc},75}^{2}},\\ \cos \varphi_{\text{sc}} &= \frac{R_{\text{sc},75}}{\left| Z_{\text{sc},75} \right|}. \end{aligned} \tag{19}$$

2.4. Losses P_{sc} in the Short-Circuit Test. If the rated currents flow through the coils $I' = I'_R = I''_1$ practically there is no difference in the value of losses in copper on the primary and secondary side:

$$P_{\mathrm{Cu},R} = 3R'I_R'^2 + 3R_1''I_{1R}''^2 = 3R_{\mathrm{sc}}I_R'^2, \qquad (20)$$

where the copper losses are several times greater than the losses in the core of the transformer at short-circuit, $P_{\text{core,sc}}$.

Short-circuit voltage is defined as the voltage that must lead to single coil, when in the second short-circuited winding rated current I'_R flows, which corresponds to the nominal voltage of coils at a temperature of 75°C. If the voltage has been brought to the primary winding, short-circuit voltage is expressed in absolute units $V'_{sc} = |Z_{sc,75}|I'_R$.

Active and reactive components of short-circuit voltage are obtained by the following expression:

$$v_{a,sc} = \frac{R_{sc,75}I'_R}{V'_R} = R_{pu,sc} = v_{sc}\cos\varphi_{sc},$$

$$v_{r,sc} = \frac{X_{sc}I'_R}{V'_R} = X_{pu,sc} = v_{sc}\sin\varphi_{sc}.$$
(21)

3. The Proposed Measurement Method

To determine the two parameters of the impedance sources (in this case resistance r and inductive reactance x transformers), it is sufficient to examine only two modes of load. By graph-analytical method, through analysis of vector



(d) Diagram from $X_c[X_l]$ load

FIGURE 2: Scheme and vector diagrams: (a) load *R*-active, (b) vector diagram from *R* load, (c) reactive $X_c(X_l)$ load, and (d) vector diagram for $X_c(X_l)$ load.

diagrams of two different passive loads, active and reactive (capacitive/inductive), realistic results can be obtained. It should be emphasized that the application of inductive loads in measurement is not recommended because inductive loads have prominent and active component of resistance that cannot be simply determined (22). The active component of the inductive load also has an impact on the phase stance φ_{2l} . Therefore, it is better to perform the procedure with a capacitive load in which the influence of components of active resistance is several times lower. From the vector diagrams that correspond to schemes of measurements (active, Figures 2(a) and 2(b), and reactive capacitive/inductive loads, Figures 2(c) and 2(d) [7]) the equation can be written for the general case:

$$R: E_{1}^{2} = (V_{1} + r \cdot I_{1})^{2} + x^{2} \cdot I_{1}^{2},$$

$$\left[\text{sign}: +rI_{1} \longleftrightarrow R = R, \ V_{1} = RI_{1}; |R| = \left|\frac{V_{1}}{I_{1}}\right|\right],$$
(22)

$$\begin{aligned} X_{c} &: E_{2}^{2} = \left(V_{2} - x \cdot I_{2}\right)^{2} + r^{2}I_{2}^{2}, \\ \left[\text{sign} : -xI_{2} \longleftrightarrow X = X_{c} = \left|\frac{V_{2}}{I_{2}}\right|, \ V_{2} = -jX_{c}I_{2} = V_{2c} \end{aligned} \right] \\ &= -jX_{c}I_{2c}, \end{aligned}$$
(23)

$$X_{l}: E_{2}^{2} = (V_{2} + x \cdot I_{2})^{2} + r^{2} \cdot I_{2}^{2},$$

$$\left[\operatorname{sign}: +xI_{2} \longleftrightarrow X = X_{l} = \left|\frac{V_{2}}{I_{2}}\right|, V_{2} = jX_{l}I_{2} = V_{2l} \quad (24)$$

$$= jX_{l}I_{2l}\right],$$

where E_1 , E_2 -ems of transformers are in idling (measured or simulated on the secondary side of the transformer when the load is omitted). The solution of (1) and (2) is as follows:

$$\begin{aligned} x^{2} \cdot I_{1}^{2} + r^{2} \cdot I_{1}^{2} + r \cdot 2 \cdot V_{1}I_{1} &= E_{1}^{2} - V_{1}^{2}, \\ \left(x^{2} + r^{2} + r \cdot 2 \cdot \frac{V_{1}}{I_{1}}\right) \cdot I_{1}^{2} &= E_{1}^{2} - V_{1}^{2}, \text{ act. load,} \\ x^{2} \cdot I_{2}^{2} + r^{2} \cdot I_{2}^{2} - x \cdot 2 \cdot V_{2}I_{2} &= E_{2}^{2} - V_{2}^{2}, \\ \left(x^{2} + r^{2} - x \cdot 2 \cdot \frac{V_{2}}{I_{2}}\right) \cdot I_{2}^{2} &= E_{2}^{2} - V_{2}^{2}, \\ \left(x^{2} + r^{2} - x \cdot 2 \cdot \frac{V_{2}}{I_{2}}\right) = \frac{E_{2}^{2}}{I_{2}^{2}} - \frac{V_{2}^{2}}{I_{2}^{2}} = N, \text{ cap. load,} \\ x^{2} \cdot I_{2}^{2} + r^{2} \cdot I_{2}^{2} + x \cdot 2 \cdot V_{2}I_{2} &= E_{2}^{2} - V_{2}^{2}, \end{aligned}$$

$$\left(x^{2} + r^{2} + x \cdot 2 \cdot \frac{V_{2}}{I_{2}}\right) \cdot I_{2}^{2} = E_{2}^{2} - V_{2}^{2},$$

$$\left(x^{2} + r^{2} + x \cdot 2 \cdot \frac{V_{2}}{I_{2}}\right) = \frac{E_{2}^{2}}{I_{2}^{2}} - \frac{V_{2}^{2}}{I_{2}^{2}} = N, \text{ ind. load.}$$

$$(25)$$

By introducing replacements,

$$m = \left(\frac{E_1}{I_1}\right)^2,$$

$$n = \left(\frac{E_2}{I_2}\right)^2,$$

$$R^2 = \left(\frac{V_1}{I_1}\right)^2,$$

$$X_c^2 = \left(\frac{V_2}{I_2}\right)^2,$$

$$X_l^2 = \left(\frac{V_2}{I_2}\right)^2.$$
(26)

The following is obtained:

$$M = m - R^{2},$$

$$N = n - X_{c}^{2},$$

$$[N = n - X_{l}^{2}],$$

$$p_{c} = m - n - (R^{2} - X_{c}^{2}),$$

$$p_{l} = m - n - (R^{2} - X_{l}^{2}).$$
(27)

(1) One has *R* & *C*: $x^2 + r^2 + r \cdot 2 \cdot R = m - R^2$; consider

$$\frac{x^{2} + r^{2} - x \cdot 2 \cdot X_{c} = n - X_{c}^{2}}{2 \left(r \cdot R + x \cdot X_{c}\right) = M - N}$$

$$x = \frac{M - N}{2X_{c}} - \frac{R}{X_{c}}r,$$

$$x = \frac{p_{c}}{2X_{c}} - \frac{R}{X_{c}}r.$$
(28)

(2) One has *R* & *L*: $x^2 + r^2 + r \cdot 2 \cdot R = m - R^2$; consider

$$\frac{x^{2} + r^{2} + x \cdot 2 \cdot X_{c} = n - X_{l}^{2}; \ \ (-1)}{2 \left(r \cdot R - x \cdot X_{l}\right) = M - N}$$

$$x = -\frac{M - N}{2X_{l}} + \frac{R}{X_{l}}r,$$

$$x = -\frac{p_{l}}{2X_{l}} + \frac{R}{X_{l}}r.$$
(29)

To determine the value r in both combinations of R & C and R & L first replace the value of the reactance x into (26):

$$\left[\frac{p_c}{2X_c} - \frac{R}{X_c}r\right]^2 + r^2 + 2R \cdot r = M,$$

$$\left[\frac{p_c}{2X_c}\right]^2 - \frac{p_c}{X_c}\frac{R}{X_c}r + \left[\frac{R}{X_c}\right]^2r^2 + r^2 + 2R \cdot r = M,$$

$$\left[-\frac{p_l}{2X_l} + \frac{R}{X_l}r\right]^2 + r^2 + 2R \cdot r = M,$$

$$\left[\frac{p_l}{2X_l}\right]^2 - \frac{p_l}{X_l}\frac{R}{X_l}r + \left[\frac{R}{X_l}\right]^2r^2 + r^2 + 2R \cdot r = M.$$
(30)

Equitation has the same form for both combinations, so it can be written with parameters *X*, *p*:

$$\left[\frac{p}{2X}\right]^2 - \frac{p}{X}\frac{R}{X}r + \left[\frac{R}{X}\right]^2r^2 + r^2 + 2R\cdot r = M$$
(31)

(explicit relation) or

$$\left[1 + \left(\frac{R}{X}\right)^2\right] \cdot r^2 + \left[2R - \frac{pR}{X^2}\right] \cdot r + \left[\frac{p}{2X}\right]^2 - M = 0 \quad (32)$$

(implicit relation).

Basic type quadratic equation is obtained: $ar^2 + br + c = 0$. After replacing values R/X = q coefficients have following values:

$$a = 1 + q^{2},$$

$$b = R \left[2 - \frac{M - N}{X^{2}} \right],$$

$$c = \left[\frac{M - N}{2X} \right]^{2} - M \text{ or }$$

$$\left\{ b = R \left[2 - \frac{p}{X^{2}} \right], \ c = \left[\frac{p}{2X} \right]^{2} - M \right\}.$$
(33)

Second grade equitation (quadratic) with nominal form $ar^2 + br + c = 0$ or (after dividing with (*a*)): $r^2 + b_1r + c_1 = 0$, $(b_1 = b/a)$, $(c_1 = c/a)$.

The number of real solutions depends on the sign of the discriminant D,

$$D = (4ac - b^2) \leq 0 \text{ or } \left(c_1 - \frac{b_1^2}{4}\right) \leq 0,$$
 (34)

if one has the following:

(i) (D < 0), $(4ac - b^2) < 0$, has 2 solutions (2 real roots).

- (ii) (D = 0), $(4ac b^2) = 0$, has 1 solution (2 same roots) $r_{1,2} = -b/2a \Leftrightarrow -b/2a > 0$, za (b < 0).
- (iii) (D > 0), $(4ac b^2) > 0$, has 2 solutions (2 complex roots).

Solving of quadratic equitation, one has the following. (1) Method: one has

$$a \cdot r^{2} + b \cdot r + c = a (r - \alpha) (r - \beta) = 0 \text{ or}$$

$$r^{2} + b_{1} \cdot r + c_{1} = (r - \alpha) (r - \beta) = 0.$$
(35)



FIGURE 3: Demonstration of the adapted discrete 3-phase programmable voltage source 3-phase V-I measurement.



FIGURE 4: Diagram voltage V_{abc} (V) and current I_{abc} (A).
Power	HV/	'LV	No-load losses	Load losses (75°C)	$R(\Omega)$	X (0)	7 (0)
(kVA)	(kV)	(%)	(W)	(W)	$R_T(22)$	$X_T(22)$	Σ_T (32)
T1ABB 1000	10/0.4	6%	2000	8800	$1.40\cdot 10^{-3}$	$9.5 \cdot 10^{-3}$	$9.6e \cdot 10^{-3}$
T2ABB 2000	10/0.4	6%	3500	15500	$0.61 \cdot 10^{-3}$	$4.76\cdot 10^{-3}$	$4.8\cdot 10^{-3}$
T3ABB 3150	10/0.4	6%	5200	22600	$0.34\cdot 10^{-3}$	$3.03\cdot 10^{-3}$	$3.05\cdot 10^{-3}$

TABLE 1: Table of the transformers of brochure [13].



FIGURE 5: Diagram voltage V_{abc} (V) and current I_{abc} (A).

(2) Method: apply the formula

$$r_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \text{ or}$$

$$r_{1,2} = \frac{-b/2 \pm \sqrt{(b/2)^2 - ac}}{a}.$$
(36)

And for the form $r^2 + b_1 \cdot r + c_1 = 0 \Leftrightarrow r_{1,2} = b_1/2 \pm \sqrt{b_1^2/4 - c_1}$, root characteristic is

$$r_1 + r_2 = -\frac{b}{a} = -b_1$$
 and $r_1 \cdot r_2 = \frac{c}{a} = c_1$. (37)

Active electric resistance of electrical sources r is due to the physical nature of the real value greater than zero r > 0 so the

Parameters	<i>T</i> 1: 1 [MVA], $ R = X_c = 0.16 [\Omega]$	<i>T</i> 2: 2 [MVA], $ R = X_c = 0.08 [\Omega]$	<i>T</i> 2: 3.15 [MVA], $ R = X_c = 0.05 [\Omega]$
$E_{1m} = E_{2m} \left[\mathbf{V} \right]$	3264	3264	3264
$E_{1m}^2 = E_{2m}^2 [\mathrm{V}^2]$	106537	106537	106537
V_{1m} [V]	3231	3231	3232
$V_{1m}^2 \left[V^2 \right]$	104393.6	104329	104458
I_{1m} [A]	2000	4000	6400
$I_{1m}^2 \left[\mathbf{A}^2 \right]$	$4 * 10^{6}$	$16 * 10^6$	$40.96 * 10^6$
V_{2m} [V]	350	350	350
V_{2m}^2 [V ²]	122500	122500	122500
I_{2m} [A]	2200	4400	7000
$I_{2m}^2 \left[\mathbf{A}^2 \right]$	$4.84 * 10^6$	$19.36 * 10^6$	$49 * 10^{6}$
$M\left[\Omega^2 ight]$	$5.359 * 10^{-4}$	$1.340 * 10^{-4}$	$0.51 * 10^{-4}$
$N\left[\Omega^2 ight]$	$-3.298 * 10^{-3}$	$-0.825 * 10^{-3}$	$-0.353 * 10^{-3}$
$\frac{M-N}{2} \left[\Omega^2 \right]$	$1.914 * 10^{-3}$	$0.478 * 10^{-3}$	$0.202 * 10^{-3}$
$\frac{M-N}{2X_c} \left[\Omega \right]$	$11.96 * 10^{-3}$	$5.99 * 10^{-3}$	$4.03 * 10^{-3}$
$\left(\frac{M-N}{2X_c}\right)^2 \left[\Omega^2\right]$	$1.431 * 10^{-4}$	$0.358 * 10^{-4}$	$0.163 * 10^{-4}$
$a = 1 + \left \frac{R}{X_c} \right ^2 = 1 + 1 = 2$	2	2	2
$b = 2X_c \left(1 - \frac{M - N}{2X_c} \frac{1}{X_c}\right) \left[\Omega^2\right]$	0.2961	0.148	0.0919
$c = \left(\frac{M-N}{2\left X_{c}\right }\right)^{2} - M\left[\Omega^{2}\right]$	$-3.928 * 10^{-4}$	$-0.981 * 10^{-4}$	$-0.344 * 10^{-4}$
$r_2 < 0$ not realistic; $r_1 [\Omega]$	$1.35 - 10^{-3}$	$0.66 * 10^{-3}$	$0.37 * 10^{-3}$
$r_{1ABB} \left[\Omega^2 \right]$	$1.40 * 10^{-3}$	$0.61 * 10^{-3}$	$0.34 * 10^{-3}$
$x_2 < \text{not realistic}; x_1 [\Omega]$	$11.82 * 10^{-3}$	$5.31 * 10^{-3}$	$3.66 * 10^{-3}$
$x_{1ABB} \left[\Omega^2 \right]$	$9.5 * 10^{-3}$	$4.76 * 10^{-3}$	3.03 * 10 ⁻³

TABLE 2: Results of values and parameters calculated using the graph-analytical method.

characteristic roots of quadratic equations have to be $\text{Re}\{r_1\} > 0$; $\text{Re}\{r_2\} > 0$. This is possible if $(D \le 0)$, $(4ac - b^2) \le 0$, has two solutions (two real roots or two same roots) and if

(1)
$$0 > -\frac{b}{2a}$$
 and $0 > \left[1 \pm \sqrt{1 - \frac{4ac}{b^2}}\right]$ or
(2) $0 < -\frac{b}{2a}$ and $0 < \left[1 \pm \sqrt{1 - \frac{4ac}{b^2}}\right]$.
(38)

Additional condition is that root value must be $[1-4ac/b^2] \ge 0$, where the value for *r* can stay real value [r] = Re and because it cannot be imaginary value: $[r] \ne \text{Im}$. As it is

 $[a > 0], [b/2a]^2 > 0$, that condition depends on the value relation [c/a]:

$$\left[1 - \frac{4ac}{b^2}\right] = \left[1 - \frac{1}{\left(b/2a\right)^2}\frac{c}{a}\right] \ge 0,$$

$$\left(\frac{b}{2a}\right)^2 \ge \left[\frac{c}{a}\right].$$
(39)

If the values are $(b/2a)^2 = [c/a]$ the values of root's quantity are equal to zero and there is one solution for $r_1 = r =$ -b/2a = Re-, the value which in principle means that the value of b < 0; that is, $b = R[2 - p/X^2] < 0$. This is possible if $[2 - p/X^2] < 0$ or with precondition $[2 < p/X^2]$.



FIGURE 6: Diagram voltage V_{abc} (V) and current I_{abc} (A).

4. Simulation Results

In this case, the graph-analytical method for the verification of the simulation method [14], which replaces the measurement method, dry power transformer ABB of coupling Yy, power 1; 2; and 3.15 MVA, and the tests with three measurements for active, for capacitive, and for inductive loads are selected (adapted the program psb3phasesignalseq (Figure 3)) [14].

Data on the three-phase ABB transformers, obtained by experimental procedure in the laboratory, are taken from [13] and shown in Table 1, with the following values:

The paper presents the following obtained values: rated power [kVA], nominal voltage, connected voltage, or idle running voltage, on primary-higher voltage HV (kV) V'_A , V'_B , and V'_C and the lower secondary LV (kV) V''_a , V''_b , and V''_c ,

the impedance value of (%), idle running losses, and losses of loaded transformer at rated current and current at rated load.

Simulations in MATLAB program (Simulink-Power System, psb3phasesignalseq) on ABB transformers are derived from 100% load on T1, T2, and T3 with $S_n = 1, 2, 3.15$ [MVA].

Secondary currents are indicated on diagrams I''_a , I''_b , and I''_c and secondary voltages in different load tests with active, reactive capacitive (or inductive) load.

Gained changes of electrical quantities are shown in Figures 4, 5, and 6.

Diagrams (for harmonic size changes, between maximal and effective values applies relation $I = I_m/\sqrt{2}$, $E = E_m/\sqrt{2}$) of currents and voltage for a combined test with active and capacitive loads as well as Table 2 which entered all the important values from diagrams and corresponding values are required for the calculation of active resistance and reactance of the transformer. From Tables 1 and 2 the deviation can be seen between the results of measuring the value of R_T , X_T , which are stated in the ABB manual and those obtained through simulations and they are different to 20% for the results of a reactance and up to 5% for the results of active resistance for all three tested transformer.

Diagrams (for harmonic size changes, between maximal and effective values applies relation $I = I_m/\sqrt{2}$, $E = E_m/\sqrt{2}$) of currents and voltages for combined test with active and inductive load for all three transformers are shown in the paper, but because of the volume and significant values of diagrams for active resistance and reactance of the transformer the table is not displayed.

5. Conclusions

Graph-analytical method can be used not only to determine the active and inductive AC power source parameters but also to determine and analyze the impendence of electrical sources with high frequencies or in a transient process. It should be emphasized that it is better to determine parameters of transformer or source from measuring or simulation test combined with *R* and *C* loads that are close to the nominal load, where the errors are minimized. Larger deviations in the measurement or simulating values occur when calculating the reactance, which is natural, because the reactive loads are usually combined with a significant presence of the active component (lower value with capacitive and higher value with inductive loads).

Conflict of Interests

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

References

- [1] A. J. Pansini, *Electrical Transformers and Power Equipment*, The Farmont Press, Lilburn, Ga, USA, 3rd edition, 1999 (Indian).
- [2] L. Lawhead, R. Hamilton, and J. Horak, "Three phase transformer winding configurations and differential relay compensation," in *Proceedings of the 60th Annual Georgia Tech Protective Relay Conference*, Atlanta, Ga, USA, May 2006.
- [3] H. Su, G. Jia, and M. Z. Q. Chen, "Semi-global containment control of multi-agent systems with input saturation," *IET Control Theory & Applications*, vol. 8, no. 18, pp. 2229–2237, 2014.
- [4] H. Su and M. Z. Q. Chen, "Multi-agent containment control with input saturation on switching topologies," *IET Control Theory & Applications*, vol. 9, no. 3, pp. 399–409, 2015.
- [5] H. Dommel, "Digital computer solution of electromagnetic transients in single-and multiphase networks," *IEEE Transactions on Power Apparatus and Systems*, vol. 88, no. 4, pp. 388– 399, 1969.
- [6] P. Spalevic, B. Miric, D. Vuckovic, and S. Panic, "Magnetic field of inductive loop—theory and experimental results," *Elektronika Ir Elektrotechnika: Elektronika*, vol. 99, no. 3, pp. 83–88, 2010.

- [7] F. De Leon and A. Semlyen, "Complete transformer model for electro-magnetic transients," *IEEE Transactions on Power Delivery*, vol. 9, no. 1, pp. 231–239, 1994.
- [8] N. Markovic, S. Bjelic, U. Jaksic, and Z. Bogicevic, "Graphical zero-sequence cut-offs method of determining of fault to earth in electrical lines," in *Proceedings of the 9th Symposium on Neural Network Applications in Electrical Engineering (NEUREL* '08), pp. 73–76, Belgrade, Serbia, September 2008.
- [9] M. Fatima, R. Youcef, H. Assia, H. Habib, and B. Abdelber, "The non linear controller for terminal voltage and power angle of power systems," *ACTA Polytechnica Hungarica*, vol. 10, no. 8, pp. 205–218, 2013.
- [10] Y. Shibuya and S. Fujita, "High frequency model and transient response of transformer windings," in *Proceedings of the IEEE/PES Transmission and Distribution Conference and Exhibition: Asia Pacific*, vol. 3, pp. 1839–1844, IEEE, Yokohama, Japan, October 2002.
- [11] L. L. Grigzby, "Transient voltage response of coils and windings," in *Response of Coils and Windings*, vol. 10, CRC Press LLC; Auburn University, Auburn, Ala, USA, 2001.
- [12] E. P. Dick and C. C. Erven, "Transformer diagnostic testing by frequuency response analysis," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-97, no. 6, pp. 2144–2153, 1978.
- [13] ABB Transformers Power Technologies Division, 2015, http:// www.abb.com/transformers.
- [14] S. Bjelić and Z. Bogićević, "Computer simulation of theoretical model of electromagnetic transient processes in power transformers," *International Journal of Information Technology and Computer Science*, vol. 6, no. 1, pp. 1–12, 2014.

Research Article

Robust Leaderless Consensus of Uncertain Multiagent Systems with Fast Switching Topologies

Wei Liu, Shaolei Zhou, Shi Yan, and Gaoyang Yin

Department of Control Engineering, Naval Aeronautical and Astronautical University, Yantai 264001, China

Correspondence should be addressed to Wei Liu; weiliu.sd.china@hotmail.com

Received 18 June 2015; Revised 28 August 2015; Accepted 12 October 2015

Academic Editor: Yuan Fan

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

This paper investigates the robust leaderless consensus problem of uncertain multiagent systems with directed fast switching topologies. The topologies are assumed to jointly contain a directed spanning tree. Based on a special property of the graph Laplacian matrix, the consensus problem is converted into a stabilization problem by performing a proper variable transformation. Averaging method is employed for analysis. It is proved that if the topologies switch sufficiently fast and the controllers are properly designed, the robust leaderless consensus can still be achieved even when all the possible topologies are unconnected in the switching time intervals. Finally, a numerical simulation is provided to illustrate the effectiveness of the theoretical results.

1. Introduction

In the past few years, the consensus problem of multiagent systems has drawn great attention for its broad potential applications in many areas such as cooperative control of vehicle, unmanned air vehicle formation, and flocking control [1–3]. Consensus means that all agents will reach a common state in a cooperative fashion throughout distributed controllers. It is called leaderless consensus if there is no specified leader in the multiagent systems. Since agents in the multiagent systems are coupled through networks, the consensus achievement depends on not only the individual agent dynamic but also the structure of the interaction topology.

In many applications, the interaction topology among agents may change dynamically. This may happen when the communication links among agents may be unreliable due to disturbance or subject to communication range limitations [4]. Motivated by this observation, many efforts have been made to investigate this topic. A pioneer work was shown in [4], in which it was proved that the consensus of first order multiagent systems could be achieved if the topologies jointly contained a directed spanning tree. Some conclusions of the first and second order dynamics were obtained in [5–8]. The consensus problem of a class of heterogeneous multiagent systems including first and second order dynamics was solved in [9]. Note that the consensus problem of first or second order dynamics is mainly solved using the stochastic indecomposable aperiodic (SIA) based approach which is not applicable to the higher order dynamics case. Thus, several other approaches are employed to solve the consensus problem for higher order multiagent systems with different topology conditions. In [10], under undirected jointly connected communication topologies, the leaderless consensus problem of high-order linear multiagent systems was addressed using extended Barbalat's lemma. When each possible directed topology was strongly connected and balanced [11], leaderless consensus problem of linear multiagent systems was solved with common Lyapunov function approach. Using multiple Lyapunov function approach, the leader-following [12] and leaderless [13] consensus problems of linear multiagent systems were addressed with the assumption that each possible topology contained a directed spanning tree. Based on averaging method, the leader-following consensus problems of linear multiagent systems with jointly connected topologies were solved with and without parameter uncertainties in [14] and [15], respectively. Under undirected topologies [16], the leader-following consensus problem of nonlinear multiagent systems was investigated with jointly connected topologies. Then, the results were extended to the case with distributed

adaptive protocols [17]. In [18], the leaderless consensus problem of a certain type of nonlinear system was investigated with fast bidirectional switching topologies.

In contrast with the leader-following consensus problem, in the homogeneous multiagent systems, the leaderless consensus problem which can include the leader-following consensus problem as a special case is more complex and challenging especially with directed communication topologies. Since the leader-following consensus problem can be conveniently converted into a stabilization problem by constructing tracking error variables, the presence of the leader in the multiagent systems facilitates the derivation [13]. As for the leaderless consensus problem, there is no specified leader. Furthermore, when the topology is assumed to be undirected, the graph Laplacian matrix is positive semidefinite. This property facilitates the construction of the Lyapunov function greatly in the derivation. As for directed topologies, the graph Laplacian matrices are not necessarily symmetric. This increases the analysis difficulty especially when there are other specified constraints such as uncertainties in systems dynamics or switching communication topologies.

Motivated by this, the objective of this paper is to solve the robust leaderless consensus problem of uncertain multiagent systems with directed fast switching topologies. It is assumed that the switching topologies jointly contain a directed spanning tree. The analysis process is of two steps. Firstly, based on the property that the graph Laplacian matrix can be factored into the product of two specific matrices, the consensus problem with switching topologies is converted into a stabilization problem of a switched system by constructing a proper disagreement vector. Secondly, by using averaging method which is widely used for stability analysis of fast switching systems, sufficient conditions for achieving the leaderless consensus are obtained. It is shown that if the topologies switch sufficiently fast and the feedback gain matrices in the consensus controllers are properly designed, consensus can still be achieved even when the topologies are not connected in the switching time intervals. The main contribution of our paper is that the leaderless consensus problem of high-order dimension multiagent systems is solved under fast switching topologies which jointly contain a directed spanning tree.

The remainder of this paper is organized as follows. In Section 2, some preliminaries are provided. In Section 3, the robust leaderless consensus problem is solved with fast switching topologies. In Section 4, a simulation example is presented. Section 5 is the conclusion.

2. Preliminaries

2.1. Notations. Throughout this paper, the following notations will be used. $\mathbb{R}^{n \times n}$ and $\mathbb{C}^{n \times n}$ denote the set of $n \times n$ real and complex matrices, respectively. \otimes denotes the Kronecker product. For $\mu \in \mathbb{C}$, the real part is $\operatorname{Re}(\mu)$. I_n is the $n \times n$ identity matrix. For a square matrix A, $\lambda(A)$ denotes the eigenvalues of matrix A; A > B ($A \ge B$) means that A - B is positive definite (resp., positive semidefinite). For a symmetrical matrix A, max{ $\lambda(A)$ } (min{ $\lambda(A)$ }) denotes the largest (smallest) eigenvalue of A. 2.2. Graph Theory. A directed graph $\mathscr{G} = (\mathscr{V}, \mathscr{C}, \mathscr{A})$ contains the vertex set $\mathscr{V} = \{1, 2, ..., N\}$, the directed edges set $\mathscr{C} \subseteq \mathscr{V} \times \mathscr{V}$, and the weighted adjacency matrix $\mathscr{A} = [a_{ij}]_{N \times N}$ with nonnegative elements a_{ij} . $a_{ij} > 0$ if there is a directed edge between vertices *i* and *j*; $a_{ij} = 0$ otherwise. The set of neighbors of *i* is defined as $\mathscr{N}_i := \{j \in \mathscr{V} : a_{ij} = 1\}$. A directed path is a sequence of ordered edges of the form $(i_1, i_2), (i_2, i_3), \ldots$, where $i_j \in \mathscr{V}$. The Laplacian matrix of the topology \mathscr{G} is defined as $\mathscr{L} = [\mathscr{L}_{ij}]_{N \times N}$, where $\mathscr{L}_{ii} = \sum_{j \neq i} a_{ij}$ and $\mathscr{L}_{ij} = -a_{ij}$. Then, 0 is an eigenvalue of \mathscr{L} with 1_N as the eigenvector. A directed graph is said to have a spanning tree if there is a vertex called the root such that there is a directed path from this vertex to every other vertex.

In this paper, the communication topology is molded by a directed graph and we assume that the communication topology is time-varying. Denote $\widehat{\mathscr{G}} = \{\mathscr{G}_1, \mathscr{G}_2, \dots, \mathscr{G}_m\}, m \ge 1$, to be the set of all possible directed topologies. We define the switching signal $\sigma(t)$, where $\sigma(t) : [0, +\infty) \rightarrow \mathscr{P} = \{1, 2, \dots, p\}$. $0 = t_0 < t_1 < t_2 < \cdots$ denote the switching instants of $\sigma(t)$. Let $\mathscr{G}_{\sigma(t)} \in \widehat{\mathscr{G}}$ be the communication topology at time *t*. Across each time interval $[t_j, t_{j+1}), j \in \mathbb{Z}$, the graph $\mathscr{G}_{\sigma(t)}$ is fixed.

A union graph of a collection of graphs $\mathscr{G}_1, \ldots, \mathscr{G}_m$ is defined as a graph, denoted by $\overline{\mathscr{G}}_{\sigma(t)}$ with the same vertex set \mathscr{V} and the edge set equaling the union of the edge sets of all the graphs in the collection, and connection weight between edges *i* and *j* is the sum of a_{ij} of $\mathscr{G}_1, \ldots, \mathscr{G}_m$. The collection $\mathscr{G}_1, \ldots, \mathscr{G}_m$ jointly contains a directed spanning tree if its union graph $\overline{\mathscr{G}}_{\sigma(t)}$ contains a directed spanning tree. Let $\overline{\mathscr{G}}_{\sigma(t)}$ be the Laplacian matrix of $\overline{\mathscr{G}}_{\sigma(t)}$.

Lemma 1 (see [4]). Zero is a simple eigenvalue of \mathcal{L} and all the other nonzero eigenvalues have positive real parts if and only if the graph \mathcal{G} has a directed spanning tree.

3. Problem Formulation and Main Results

Consider a multiagent system composed of *N* agents with the following uncertain dynamics:

$$\dot{x}_i(t) = (A + \Delta A) x_i(t) + B u_i(t), \quad i = 1, 2, \dots, N,$$
 (1)

where $x_i(t) \in \mathbb{R}^n$, $u_i(t) \in \mathbb{R}^p$ are the state and the control input of the *i*th agent, respectively. A and B are constant system matrices with compatible dimensions. ΔA represents the time-varying parameter uncertainty and is assumed to be in the form of $\Delta A = DF(t)H$, where D and H are known matrices of appropriate dimensions which characterize the structure of the uncertainty. F(t) is an uncertain matrix satisfying $F(t)^T F(t) \le \rho^2 I$ and $\rho > 0$ is a given constant.

Definition 2. The consensus of system (1) is said to be achieved with any finite initial value $x_i(0)$, if there exists a controller $u_i(t)$ such that

$$\lim_{t \to \infty} \left\| x_i(t) - x_j(t) \right\| = 0, \quad \forall i, j = 1, 2, \dots, N.$$
 (2)

In order to achieve consensus, the following distributed consensus controller based on local relative states information of neighbor agents is proposed:

$$u_{i}(t) = cK \sum_{j=1}^{N} a_{ij}^{\sigma(t)}(t) \left(x_{j}(t) - x_{i}(t) \right), \quad i = 1, \dots, N, \quad (3)$$

where $K \in \mathbb{R}^{p \times n}$ is the feedback matrix to be designed, *c* is the coupling strength to be selected, and $a_{ij}^{\sigma(t)}(t)$ is the element of the adjacency matrix $\mathscr{A}_{\sigma(t)}$ of the graph $\mathscr{G}_{\sigma(t)}$. The closed-loop system dynamics of (1) with the con-

The closed-loop system dynamics of (1) with the controller (3) is

$$\dot{x}(t) = (I_N \otimes (A + \Delta A) - c \mathscr{L}_{\sigma(t)} \otimes BK) x(t), \qquad (4)$$

where $x(t) = [x_1(t)^T, x_2(t)^T, \dots, x_N(t)^T]^T$ and $\mathscr{L}_{\sigma(t)} \in \mathbb{R}^{N \times N}$ is the Laplacian matrix of the graph $\mathscr{G}_{\sigma(t)}$.

Without loss of generality, consider an infinite sequence of nonempty, bounded, nonoverlapping, and contiguous time intervals $[t_k, t_{k+1})$, k = 0, 1, ..., with $t_0 = 0$, $t_{k+1} - t_k \le T$ for some constant T > 0. Suppose that, in each interval $[t_k, t_{k+1})$, there is a sequence of m_k nonoverlapping subintervals

$$\begin{bmatrix} t_k^1, t_k^2 \end{bmatrix}, \dots, \begin{bmatrix} t_k^j, t_k^{j+1} \end{bmatrix}, \dots, \begin{bmatrix} t_k^{m_k}, t_k^{m_k+1} \end{bmatrix},$$

$$t_k = t_k^1, \ t_{k+1} = t_k^{m_k+1},$$
(5)

satisfying $t_k^{j+1} - t_k^j \ge \tau$, $1 \le j \le m_k$ for a given constant $\tau > 0$, such that, during each of such subintervals, the communication topology is fixed. The graph of communication topology is denoted by \mathscr{G}_{k_j} . In each time interval $[t_k^j, t_k^{j+1})$, the topology is not necessarily connected.

Assumption 3. In this paper, we assume that across each time interval $[t_k, t_{k+1}), k = 0, 1, ...,$ the union graph $\overline{\mathscr{G}}$ contains a directed spanning tree and remains the same.

Lemma 4 (see [19]). For any given $x, y \in \mathbb{R}^n$ and matrices P > 0, D, and S of appropriate dimensions, one has

$$2x^{T}DSy \le x^{T}DPD^{T}x + y^{T}S^{T}P^{-1}Sy.$$
(6)

Lemma 5 (see [20]). For a Laplacian matrix $\mathscr{L} \in \mathbb{R}^{N \times N}$ of graph \mathscr{G} and a full row rank matrix $E \in \mathbb{R}^{(N-1) \times N}$ defined as

$$E = \begin{bmatrix} 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 1 & -1 \end{bmatrix},$$
 (7)

there exists a matrix $M \in \mathbb{R}^{N \times (N-1)}$ such that $\mathscr{L} = ME$. Furthermore, if the graph has a directed spanning tree, M is full column rank and $\operatorname{Re}(\lambda(EM)) > 0$.

Remark 6. From the definition of union graph $\overline{\mathscr{G}}$ of a collection of graphs $\{\mathscr{G}_1, \ldots, \mathscr{G}_m\}$, we can obtain that the

Laplacian matrix $\overline{\mathscr{D}}$ of $\overline{\mathscr{G}}$ is the sum of the Laplacian matrices $\{\mathscr{L}_1, \ldots, \mathscr{L}_m\}$ of $\{\mathscr{G}_1, \ldots, \mathscr{G}_m\}$; that is, $\overline{\mathscr{D}} = \mathscr{L}_1 + \cdots + \mathscr{L}_m$. Since $\overline{\mathscr{G}}$ contains a directed spanning tree, $\overline{\mathscr{D}}$ has the property shown in Lemma 5 and so is $\overline{\mathscr{D}}^{av} = \sum_{i=1}^N \omega_i \mathscr{L}_i$, where $\omega_i > 0$ and $\sum_{i=1}^N \omega_i = 1$.

Lemma 7 (see [20]). *If the graph fulfills Assumption 3, there exist a positive definite matrix Q and a positive scalar* α *such that*

$$(EM)^T Q + QEM > \alpha Q, \tag{8}$$

where $0 < \alpha < 2\min{\text{Re}(\lambda(EM))}$, *E* is defined in (7), and *M* is a matrix satisfying $\mathcal{L} = ME$.

Let $\xi(t) = (E \otimes I_n)x(t)$, where $\xi_i(t) = x_i(t) - x_{i+1}(t)$, i = 1, 2, ..., N-1. Using Lemma 5, closed-loop system dynamics (4) can be rewritten as

$$\dot{\xi}(t) = \left(I_{N-1} \otimes (A + \Delta A) - cEM_{\sigma(t)} \otimes BK\right) \xi(t), \quad (9)$$

where $M_{\sigma(t)}$ is a matrix satisfying $\mathscr{L}_{\sigma(t)} = M_{\sigma(t)}E$.

Note that $\xi(t) = 0$ if and only if $x_1(t) = x_2(t) = \cdots = x_N(t)$. Therefore, we will design appropriate feedback gain matrix and the topology switching law such that system (9) is global asymptotically stable. The averaging method will be used.

Lemma 8 (see [21]). Suppose there exists a constant T for which the matrix-valued function B(t) is such that $(1/T) \int_{t}^{t+T} B(\tau) d\tau = \overline{B}$ for all t and

$$\dot{x}(t) = \left(A(t) + \overline{B}\right)x(t), \quad x(t_0) = x_0, \ t \ge t_0 \tag{10}$$

is uniformly exponentially stable. Then, there exists $\varepsilon^* > 0$ such that, for all fixed $\varepsilon \in (0, \varepsilon^*)$,

$$\dot{x}(t) = \left(A(t) + B\left(\frac{t}{\varepsilon}\right)\right)x(t), \quad t \ge t_0$$
(11)

is uniformly exponentially stable.

Let $\overline{M} = (1/T) \int_{t}^{t+T} M_{\sigma(s)} ds = \sum_{s=t}^{t+T} \tau_{\sigma(s)} M_{\sigma(s)}$, where $\tau_{\sigma(s)} = (t_{k}^{j+1} - t_{k}^{j})/T$, $j = k_{1}, \dots, k_{m_{k}}$, $t_{k}^{k_{1}} = t$, and $t_{k}^{k_{m_{k}}} = t + T$. Considering Lemma 5, one has $\overline{\mathscr{D}}^{av} = (1/T) \int_{t}^{t+T} M_{\sigma(\tau)} E d\tau = \overline{M}E$. Since Assumption 3 holds, $\overline{\mathscr{D}}^{av}$ has the property that $\operatorname{Re}(\lambda(E\overline{M})) > 0$. Thus, from (9), one can obtain the time-average system

$$\dot{\xi}(t) = \left(I_{N-1} \otimes (A + \Delta A) - c\overline{M} \otimes BK\right) \xi(t) .$$
(12)

Theorem 9. Suppose that Assumption 3 holds; there exist a real scalar c > 0 and a positive definite matrix P > 0 such that

$$\begin{bmatrix} A^{T}P + PA - c\alpha PBB^{T}P & H^{T} & PD \\ H & -\frac{1}{\rho^{2}}I & 0 \\ D^{T}P & 0 & -I \end{bmatrix} < 0, \quad (13)$$

where α is defined in Lemma 7 satisfying $0 < \alpha < 2\min\{\operatorname{Re}(\lambda(E\overline{M}))\}\)$. Then, there exists a positive constant $\varepsilon^* > 0$ such that the robust consensus of agents with the closed-loop dynamics

$$\dot{x}(t) = \left(I_N \otimes (A + \Delta A) - c\mathcal{L}_{(t/\varepsilon)} \otimes B_2 K\right) x(t)$$
(14)

can be achieved for any $\varepsilon \in (0, \varepsilon^*)$, where the parameter ε characterizes the speed of variation in $\mathscr{G}_{(t/\varepsilon)}$ and the feedback gain matrix can be designed as $K = B^T P$.

Proof. According to Schur complement lemma [22], linear matrix inequality (LMI) (13) holds if and only if the following inequality holds:

$$A^{T}P + PA - c\alpha PBB^{T}P + \rho^{2}H^{T}H + PDD^{T}P < 0.$$
 (15)

Consider the following Lyapunov candidate of timeaverage system (12):

$$V(t) = \xi(t)^{T} (Q \otimes P) \xi(t), \qquad (16)$$

where P is a solution of inequality (13) and Q is a feasible solution of (8). Then, the derivation of this Lyapunov candidate along the trajectory of system (12) is

$$\dot{V}(t) = 2\xi(t)^{T} \left(I_{N-1} \otimes (A + \Delta A) - cE\overline{M} \otimes BK \right)^{T}$$

$$\cdot (Q \otimes P) \xi(t) .$$
(17)

Substituting $K = B^T P$ into (17) yields

$$\dot{V}(t) = \xi(t)^{T} \left(Q \otimes \left(A^{T} P + P A \right) - c \left(\left(E \overline{M} \right)^{T} Q + Q E \overline{M} \right) \otimes P B B^{T} P \right) \xi(t) + 2\xi(t)^{T}$$
(18)

$$\cdot \left(Q \otimes \left(\Delta A \right)^{T} P \right) \xi(t) .$$

In light of the fact that $(E\overline{M})^T Q + QE\overline{M} > \alpha Q$, where $0 < \alpha < 2\min\{\operatorname{Re}(\lambda(E\overline{M}))\}\)$, one has

$$\dot{V}(t) \leq \xi(t)^{T} \left(Q \otimes \left(A^{T}P + PA - c\alpha PBB^{T}P \right) \right) \xi(t)$$

$$+ 2\xi(t)^{T} \left(Q \otimes \left(\Delta A \right)^{T}P \right) \xi(t) .$$
(19)

Using Lemma 4, one has

$$2\xi(t)^{T} \left(Q \otimes (\Delta A)^{T} P \right) \xi(t)$$

$$\leq \xi(t)^{T} \left(Q \otimes H^{T} F^{T} \right) \left(Q^{-1} \otimes I \right) \left(Q \otimes FH \right) \xi(t)^{T}$$

$$+ \xi(t)^{T} \left(I_{N-1} \otimes PD \right) \left(Q \otimes I \right) \left(I_{N-1} \otimes D^{T} P \right) \xi(t)$$

$$\leq \xi(t)^{T} \left(Q \otimes H^{T} F^{T} FH \right) \xi(t)^{T}$$

$$+ \xi(t)^{T} \left(Q \otimes PDD^{T} P \right) \xi(t)$$

$$\leq \xi(t)^{T} \left(Q \otimes \left(\rho^{2} H^{T} H + PDD^{T} P \right) \right) \xi(t).$$
(20)

In light of (20), it then follows from (19) that

$$\dot{V}(t) \leq \xi(t)^{T} \left(Q \\ \otimes \left(A^{T} P + PA - c \alpha P B B^{T} P + \rho^{2} H^{T} H + P D D^{T} P \right) \right) \quad (21)$$
$$\cdot \xi(t) .$$

Then, (15) implies that $\dot{V}(t) < 0$. Thus, time-average system (12) is asymptotically stable.

According to Lemma 8, this means that there exists a positive constant $\varepsilon^* > 0$ such that for all fixed $\varepsilon \in (0, \varepsilon^*)$

$$\dot{\xi}(t) = \left(I_{N-1} \otimes (A + \Delta A) - cEM_{(t/\varepsilon)} \otimes BK\right)\xi(t)$$
(22)

is asymptotically stable. That implies that if the topologies switch sufficiently fast, the consensus of system (14) will be achieved. $\hfill \Box$

Remark 10. Commonly, the leader-following consensus problem can be conveniently converted into a stabilization problem by constructing the tracking error variables [14–17]. In contrast, relying on Lemma 5 and the variable transformation, the leaderless consensus problem of system (4) with directed fast switching topologies is successfully converted into the stabilization problem of fast switching system (9) here. Then, the averaging method is used to obtain the sufficient conditions for achieving consensus.

Remark 11. Most of existing works about the consensus problem of higher order systems with jointly connected topologies are restricted to be undirected topologies [10, 14-18]. In this case, the graph Laplacian matrices are positive semidefinite. This property facilitated the construction of the Lyapunov function greatly in the derivation in both the leaderless and leader-following consensus problems. However, the Laplacian matrices of the directed graphs are usually not symmetric. Thus, the leaderless consensus problem with directed switching topologies is more challenging. Here, sufficient conditions for achieving consensus are obtained under directed switching topologies. It is just required that the union graph jointly contains a directed spanning tree, which means that the topologies are allowed to be unconnected in the switching time intervals. Although the topology condition in Assumption 3 is a bit strong, in which the union graph $\overline{\mathscr{G}}$ is assumed to be the same in each time interval $[t_k, t_{k+1}), k = 0, 1, \dots, a$ forward step has been made here towards a weaker communication topology condition that the union graph $\overline{\mathscr{G}}$ can be different in such time intervals.

4. Examples

In this section, we provide an example to illustrate the effectiveness of the above theoretical results. Consider a



FIGURE 1: Communication topologies $\{\mathcal{G}^1, \mathcal{G}^2, \mathcal{G}^3\}$ and the union graph $\overline{\mathcal{G}}$.



FIGURE 2: Switching signal.

multiagent system consisting of four agents in the form of position-speed model of moving plant with uncertainties

$$x_{i} = \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix},$$
$$B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad D = H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$
(23)

 $F(t) = 0.5 \sin(t)$. Then, $\rho = 0.5$.

The directed switching communication topologies $\{\mathscr{G}^1, \mathscr{G}^2, \mathscr{G}^3\}$ are given in Figure 1. Clearly, the topologies are not connected and the union graph $\overline{\mathscr{G}}$ of $\{\mathscr{G}^1, \mathscr{G}^2, \mathscr{G}^3\}$ contains a directed spanning tree. The topologies are switching as $\mathscr{G}^1 \to \mathscr{G}^2 \to \mathscr{G}^3 \to \mathscr{G}^1 \to \cdots$ and each graph is active for 0.2 seconds. We can see that, for each T = 0.6 seconds, $\overline{\mathscr{G}}$ contains a directed spanning tree.



FIGURE 3: Trajectories of x_{i1} , $i = 1, \ldots, 4$.



FIGURE 4: Trajectories of x_{i2} , $i = 1, \ldots, 4$.

Solving LMI (13) with c = 1.2, $\alpha = 1.5$, and $\rho = 0.5$, one has

$$P = \begin{bmatrix} 5.4872 & 7.1563\\ 7.1563 & 10.5219 \end{bmatrix}.$$
 (24)

Then, the feedback matrix can be chosen as $K = [7.1563 \ 10.5219]$. Figures 3 and 4 show the states trajectories of all agents with the switching signal being shown in Figure 2. It is shown that the consensus is achieved.

5. Conclusions

This paper has used the averaging method to solve the robust leaderless consensus problem of uncertain multiagent systems with fast switching topologies. The communication topologies are assumed to jointly contain a directed spanning tree. It has been proved that if the topologies switch sufficiently fast and the feedback matrix is properly designed, the consensus can be achieved even when the topologies are not connected in the switching time intervals.

Conflict of Interests

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

References

- J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.
- [2] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.
- [3] H. Chen, C. Wang, and Y. J. Lou, "Flocking multiple microparticles with automatically controlled optical tweezers: solutions and experiments," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 6, pp. 1518–1527, 2013.
- [4] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655– 661, 2005.
- [5] Y. Fan, L. Liu, G. Feng, C. Song, and Y. Wang, "Virtual neighbor based connectivity preserving of multi-agent systems with bounded control inputs in the presence of unreliable communication links," *Automatica*, vol. 49, no. 5, pp. 1261–1267, 2013.
- [6] K. Chen, J. Wang, Y. Zhang, and Z. Liu, "Second-order consensus of nonlinear multi-agent systems with restricted switching topology and time delay," *Nonlinear Dynamics*, vol. 78, no. 2, pp. 881–887, 2014.
- [7] H. Li, X. Liao, X. Lei, T. Huang, and W. Zhu, "Secondorder consensus seeking in multi-agent systems with nonlinear dynamics over random switching directed networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, no. 6, pp. 1595–1607, 2013.
- [8] M. J. Park, O. M. Kwon, J. H. Park, S. M. Lee, and E. J. Cha, "Leader following consensus criteria for multi-agent systems with time-varying delays and switching interconnection topologies," *Chinese Physics B*, vol. 21, no. 11, Article ID 110508, 2012.
- [9] Y.-J. Sun, G.-L. Zhang, S.-X. Zhang, and J. Zeng, "Consensus analysis for a class of heterogeneous multi-agent systems in fixed and switching topology," *Acta Physica Sinica*, vol. 63, no. 22, Article ID 220201, 2014.
- [10] Y. Su and J. Huang, "Stability of a class of linear switching systems with applications to two consensus problems," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1420–1430, 2012.

- [11] G. Wen, W. Yu, J. Cao, G. Hu, and G. Chen, "Consensus control of switching directed networks with general linear node dynamics," in *Proceedings of the 9th Asian Control Conference* (ASCC '13), pp. 1–6, IEEE, Istanbul, Turkey, June 2013.
- [12] G. Wen, G. Hu, W. Yu, J. Cao, and G. Chen, "Consensus tracking for higher-order multi-agent systems with switching directed topologies and occasionally missing control inputs," *Systems & Control Letters*, vol. 62, no. 12, pp. 1151–1158, 2013.
- [13] G. Wen and V. Ugrinovskii, "Distributed consensus of linear multi-agent systems with switching directedtopologies," http://arxiv.org/abs/1409.5519.
- [14] W. Ni, X. Wang, and C. Xiong, "Consensus controllability, observability and robust design for leader-following linear multi-agent systems," *Automatica*, vol. 49, no. 7, pp. 2199–2205, 2013.
- [15] W. Ni, X. Wang, and C. Xiong, "Leader-following consensus of multiple linear systems under switching topologies: an averaging method," *Kybernetika*, vol. 48, no. 6, pp. 1194–1210, 2012.
- [16] W. Xu, J. Cao, W. Yu, and J. Lu, "Leader-following consensus of non-linear multi-agent systems with jointly connected topology," *IET Control Theory & Applications*, vol. 8, no. 6, pp. 432– 440, 2014.
- [17] X. Mu, X. Xiao, K. Liu, and J. Zhang, "Leader-following consensus of multi-agent systems with jointly connected topology using distributed adaptive protocols," *Journal of the Franklin Institute*, vol. 351, no. 12, pp. 5399–5410, 2014.
- [18] Q. Jia and W. K. S. Tang, "Averaging method for consensus of fast switching multi-agent networks with communication delay," in *Proceedings of the 26th Chinese Control and Decision Conference* (*CCDC '14*), pp. 258–262, Changsha, China, June 2014.
- [19] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, New York, NY, USA, 1985.
- [20] S. Zhou, W. Liu, Q. Wu, and G. Yin, "Leaderless consensus of linear multi-agent systems: matrix decomposition approach," in *Proceedings of the 7th International Conference on Intelligent Human-Machine Systems and Cybernetics*, pp. 327–332, Hangzhou, China, August 2015.
- [21] D. J. Stilwell, E. M. Bollt, and D. G. Roberson, "Sufficient conditions for fast switching synchronization in time-varying network topologies," *SIAM Journal on Applied Dynamical Systems*, vol. 5, no. 1, pp. 140–156, 2006.
- [22] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in Systems and Control Theory*, SIAM, Philadelphia, Pa, USA, 1994.

Research Article

Networked Convergence of Fractional-Order Multiagent Systems with a Leader and Delay

Yuntao Shi¹ and Junjun Zhang²

¹*Key Laboratory of Beijing for Field-Bus Technology & Automation, North China University of Technology, Beijing 100144, China* ²*College of Science, North China University of Technology, Beijing 100144, China*

Correspondence should be addressed to Yuntao Shi; shiyuntao@ncut.edu.cn

Received 27 August 2015; Accepted 5 October 2015

Academic Editor: Michael Z. Q. Chen

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

This paper investigates the convergence of fractional-order discrete-time multiagent systems with a leader and sampling delay by using Hermite-Biehler theorem and the change of bilinearity. It is shown that such system can achieve convergence depending on the sampling interval h, the fractional-order α , and the sampling delay τ and its interconnection topology. Finally, some numerical simulations are given to illustrate the results.

1. Introduction

Recently, more and more scholars focus on the coordinated control [1, 2] of multiagent systems such as the consensus [3-5] and the controllability [6-8]. However, most of the practical distribution systems are fractional order [9-12]. Recently, with the development of society, fractional-order calculus theory [13-16] is widely used to study the signal processing and control, picture processing and artificial intelligence, and so on. The consensus of multiagent systems refers to the fact that agents in the system can transfer information and influence each other according to a certain protocol or algorithm, and eventually agents will tend to the consensus behavior with the evolution of the time in [17]. In fact, for most of multiagent systems, there widely exist time delays as in [18]. So the property of multiagent systems with time delays has always been the hot problem. In [19], the authors studied consensus of multiagent systems with heterogeneous delays and leader-following with integer-order and continuous time. In [20], the paper considered the consensus of fractionalorder multiagent systems with sampling delays without the leader.

However, for a complex environment, multiagent systems with fractional-order can be better to describe some real natural phenomena. Some basic issues of fractional-order multiagent systems with time delay, such as the convergence, are still lacking in studying. Specially, for a fractional-order multiagent system, which depends crucially on sampling interval h, the fractional-order α , and its interconnection topology, therefore, it is more difficult to study the convergence of the fractional-order multiagent system.

In this paper, we consider the convergence of fractionalorder discrete-time multiagent systems with a leader and sampling delay. The leader plays the role of an external input or signal to followers, and the followers update their states based on the information available from their neighbors and the leader. We will establish convergence conditions and discuss relations among sampling interval *h*, the fractionalorder α , its sampling delay τ , and its interconnection topology of such network.

The remainder of this paper is organized as follows. Section 2 gives the model and some preliminaries. Section 3 presents the main results, and some simulations are given in Section 4. Finally, Section 5 gives the conclusion.

2. Preliminaries and Problem Statement

In this section, we introduce some useful concepts and notations about the definition of fractional derivative [21], graph theory, and convergence of the multiagent systems. Denote a directed graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ consisting of a nonempty set of vertices \mathcal{V} and $\mathcal{E} = \{(i, j) : i, j \in \mathcal{V}\}$ is a set of edges, where (i, j) means an arc starts from *i* and ends by *j*. If $i, j \in \mathcal{V}$ and $(i, j) \in \mathcal{E}$, then we say that *i* and *j* are adjacent or *j* is a neighbor of *i*. We make $\mathcal{N}_i = \{i \in \mathcal{V} : (i, j) \in \mathcal{E}\}$ be the neighborhood set of node *i*. $A = [a_{ij}]$ is an adjacency matrix of graph \mathcal{G} , where $a_{ij} \geq 0$ is the coupling weight between any two agents. $D = \text{diag}\{d_1, d_2, \dots, d_n\} \in \mathbb{R}^{n \times n}$ is a degree matrix of \mathcal{G} ; its diagonal elements $d_i = \sum_{j \in \mathcal{N}_i} a_{ij}, i =$ $1, 2, \dots, n$, for the graph. Then the Laplacian of the weighted graph \mathcal{G} is defined as

$$L = D - A \in \mathbb{R}^{n \times n}.$$
 (1)

The agent *i* is a globally reachable agent if it has paths to all of other agents.

Definition 1 (see [17]). Assume that, for arbitrary given initial values, if

$$\lim_{t \to \infty} \left(x_i(t) - s_i x_0(t) \right) = 0, \tag{2}$$

 $i \in \mathbb{N}$, where $x_i(t) \in \mathbb{R}^n$ is the state value of agent of the multiagent system $i \ (i \in \mathbb{N}, \mathbb{N})$ presents an index set

(1, 2, ..., N), $x_0(t) \in \mathbb{R}^n$, and s_i is a constant which is changed with different *i*. Then we have that the multiagent system is convergence.

Definition 2 (see [21] (Grunwald-Letnikov)). For any real number α , the integer part written for α is [α]. If the function f(t) has continuous (m + 1)-order derivative in the interval [α , t] and m equals [α] at last when $\alpha > 0$, then let α -order derivative be

$$f^{(\alpha)}(t) = \lim_{h \to 0} h^{-\alpha} \sum_{i=0}^{(t-\alpha)/h} (-1)^i \binom{-\alpha}{i} f(t-ih).$$
(3)

Consider a multiagent system is composed of N+1 agents, where the first N (labeled from 1 to N) are followers and the remainder agent N + 1 (labeled 0) is leader. The fractionalorder discrete-time multiagent system with a leader and sampling time is described by

$$x_{i}(k+1) = \alpha x_{i}(k) + h^{\alpha} u_{i}(k),$$

$$x_{0}(k+1) = x_{0}(k),$$
(4)

where

$$u_{i}(k) = \begin{cases} \sum_{j \in \mathcal{N}_{i}} a_{ij} \left(x_{j} \left(k - 1 \right) - x_{i} \left(k - 1 \right) \right) + b_{i0} \left(x_{0} \left(k - 1 \right) - x_{i} \left(k - 1 \right) \right), & t \in [kh, kh + \tau), \\ \sum_{j \in \mathcal{N}_{i}} a_{ij} \left(x_{j} \left(k \right) - x_{i} \left(k \right) \right) + b_{i0} \left(x_{0} \left(k \right) - x_{i} \left(k \right) \right), & t \in [kh + \tau \left(k + 1 \right) h + \tau), \end{cases}$$

$$(5)$$

 $\alpha \in (0, 1), x_i \in \mathbb{R}^n$ is the state of follower $i \ (i \in \mathbb{N}, \mathbb{N})$ presents an index set (1, 2, ..., N), and $x_0 \in \mathbb{R}^n$ is the state of the leader. \mathcal{N}_i is the neighbor set of agent $i. a_{ij} \ge 0$, $b_{i0} \ge 0$ represent the coupling information between followers and from the leader to the followers, respectively; otherwise, $a_{ij} = 0$ and $b_{i0} = 0$; h > 0 is the sampling interval and the sampling interval is h and the sampling delay is $0 < \tau < h$.

3. Main Results

Let $X(k) = (x_1(k), x_2(k), ..., x_N(k))^T$ and $X_0(k) = X_0(k + 1)$ be the state vectors of all the followers and the leader, respectively. Then such system can be rewritten as

$$\binom{X(k+1)}{X(k)} = U\binom{X(k)}{X(k-1)} + \binom{h^{\alpha}B1X_0(k)}{0}, \quad (6)$$

where

$$U = \begin{pmatrix} \alpha I_n - h^{\alpha} \left(1 - \tau\right) \left(L + B\right) & -h^{\alpha} \tau \left(L + B\right) \\ I_n & 0 \end{pmatrix}, \quad (7)$$

 I_N is the $N \times N$ identity matrix, $B = \text{diag}\{b_{10}, b_{20}, \dots, b_{N0}\} \in \mathbb{R}^{N \times N}$, $\mathbf{1} = (1, 1, \dots, 1)^T$ is the $N \times 1$ identity matrix, and $L = [l_{ij}] \in \mathbb{R}^{N \times N}$ is Laplacian matrix with

$$l_{ij} = \begin{cases} -a_{ij}, & i \neq j, \ j \in \mathcal{N}_i, \\ \sum_{j \in \mathcal{N}_i} a_{ij}, & i = j, \\ 0, & \text{otherwise.} \end{cases}$$
(8)

Lemma 3 ((Hermite-Biehler theorem) [22]). Assume the polynomial $q(s) = \rho_0 + \rho_1 s + \cdots + \rho_n s^n$, marking $q(j\omega) = m(\omega) + jn(\omega)$. So q(s) is Hurwitz stable if and only if the roots of $m(\omega) = 0$, $m_1 < m_2 < \cdots$, and $n(\omega) = 0$, $n_1 < n_2 < \cdots$ satisfy

(1)
$$m(0)n'(0) - m'(0)n(0) > 0;$$

(2) $m_1 < n_1 < m_2 < n_2 < \cdots$ or $n_1 < m_1 < n_2 < m_2 < \cdots$

Lemma 4 (see [23]). $||M||_2 = \rho(M)$, if $M \in \mathbb{R}^{N \times N}$ is a symmetrical matrix.

Theorem 5. Suppose system (4) is a symmetrical and directly weighted network and the leader is a globally reachable agent;



FIGURE 1: The trajectories of nine agents in the dynamical network.

then the state of each agent can converge to the spanning space of the leader's state, if and only if $\tau < \min\{1/2, h\}$ and

$$h < \min\left\{ \left(\frac{1+\alpha}{\lambda_n \left(1-2\tau\right)}\right)^{1/\alpha}, \frac{1}{\sqrt[\alpha]{\lambda_n \tau}} \right\}, \tag{9}$$

where λ_n is the biggest eigenvalue of matrix (L + B).

Proof. The fractional-order multiagent systems with sampling delay can be convergence if and only if ||U|| < 1 which means the eigenvalues of matrix U are less than 1. Because the symmetrically directed weighted network at least has a globally reachable agent, (L + B) can be orthogonal similar to a diagonal matrix. There exists an orthogonal matrix P which makes

$$L + B = P\Lambda P^{-1},\tag{10}$$

where $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_n\}$ and the eigenvalues of matrix (L + B) satisfy $0 < \lambda_1 \le \lambda_2 \le \dots \le \lambda_n$. Assuming that *z* is the eigenvalue of matrix *U*, then the characteristic polynomial of *U* is

$$det (zI_{2n\times 2n} - U)$$

$$= det \begin{pmatrix} zI_n - \alpha I_n + h^{\alpha} (1 - \tau) (L + B) & h^{\alpha} \tau (L + B) \\ -I_n & zI_n \end{pmatrix}$$

$$= det (z^2 I_n - z (\alpha I_n - h^{\alpha} (1 - \tau) (L + B))$$

$$+ h^{\alpha} \tau (L + B))$$

$$= det (z^2 I_n - z\alpha I_n + (zh^{\alpha} (1 - \tau) + h^{\alpha} \tau) (L + B))$$

$$= \prod_{i \in N} (z^2 - z\alpha + (zh^{\alpha} (1 - \tau) + h^{\alpha} \tau) \lambda_i)$$
(11)

 $\triangleq a(z)$.



FIGURE 2: The trajectories of nine agents in the dynamical network.

Agent 9

Agent 4

Agent 5



FIGURE 3: The trajectories of nine agents in the dynamical network.

Let a(z) = 0; applying the double linear change z = (s + 1)/(s - 1), we can have

$$b(s) = (1 - \alpha + h^{\alpha}\lambda_i)s^2 + (2 - 2h^{\alpha}\tau\lambda_i)s + 1 + \alpha$$

- h^{\alpha}\lambda_i + 2h^{\alpha}\tau\lambda_i. (12)

Since the network is symmetrical and directed, (L + B) can be orthogonally similar to a diagonal matrix, whose eigenvalues are all positive. Let $s = j\omega$; then

$$b(j\omega) = -(1 - \alpha + h^{\alpha}\lambda_{i})\omega^{2} + j(2 - 2h^{\alpha}\tau\lambda_{i})\omega + 1$$

+ $\alpha - h^{\alpha}\lambda_{i} + 2h^{\alpha}\tau\lambda_{i}.$ (13)

Denote $b(\omega) = m(\omega) + jn(\omega)$, where

$$m(\omega) = -(1 - \alpha + h^{\alpha}\lambda_{i})\omega^{2} + 1 + \alpha - h^{\alpha}\lambda_{i} + 2h^{\alpha}\tau\lambda_{i},$$

$$n(\omega) = (2 - 2h^{\alpha}\tau\lambda_{i})\omega.$$
(14)





FIGURE 4: The trajectories of nine agents in the dynamical network.



FIGURE 5: The trajectories of nine agents in the dynamical network.

Using Lemma 3, system (4) can be asymptotic convergence if and only if we have the following:

(1) if m(0)n'(0) - m'(0)n(0) > 0, then $1 + \alpha - h^{\alpha}\lambda_i + 2h^{\alpha}\tau\lambda_i > 0$, $1 - h^{\alpha}\tau\lambda_i > 0$, and then

$$h^{\alpha} < \frac{1+\alpha}{\lambda_i (1-2\tau)}, \quad h^{\alpha} < \frac{1}{(\tau\lambda_i)};$$
 (15)

(2) if $1 - h^{\alpha} \tau \lambda_i \neq 0$, then the roots of $n(\omega)$ and $m(\omega)$ are the same, where the roots of $m(\omega)$ satisfy

$$m^{2}(\omega) = \frac{\left(1 + \alpha - h^{\alpha}\lambda_{i} + 2h^{\alpha}\tau\lambda_{i}\right)}{\left(1 - \alpha + h^{\alpha}\lambda_{i}\right)}$$
(16)

with $1 + \alpha - h^{\alpha}\lambda_i + 2h^{\alpha}\tau\lambda_i > 0$. Therefore, system (4) is asymptotic convergence.

Theorem 6. Suppose system (4) is a symmetrical and directly weighted network and the leader is a globally reachable agent; then the state of each agent can converge to the spanning space of the leader's state, if $\rho(A) < 1$, where $A = h^{\alpha} \tau(L + B)$ and $\rho(A)$ is spectral radius of matrix A.

Proof. The fractional-order multiagent systems with sampling delay can be convergence if and only if ||U|| < 1. Using Lemma 4,

 $\|U\|$

$$= \left\| \begin{pmatrix} \alpha I_n - h^{\alpha} (1-\tau) (L+B) & I_n \\ I_n & 0 \end{pmatrix} \begin{pmatrix} I_n & 0 \\ 0 & -h^{\alpha} \tau (L+B) \end{pmatrix} \right\|$$

$$\leq \left\| \begin{pmatrix} \alpha I_n - h^{\alpha} (1-\tau) (L+B) & I_n \\ I_n & 0 \end{pmatrix} \right\| \left\| \begin{pmatrix} I_n & 0 \\ 0 & -h^{\alpha} \tau (L+B) \end{pmatrix} \right\|$$
(17)
$$= \rho \begin{pmatrix} \alpha I_n - h^{\alpha} (1-\tau) (L+B) & I_n \\ I_n & 0 \end{pmatrix} \rho \begin{pmatrix} I_n & 0 \\ 0 & -h^{\alpha} \tau (L+B) \end{pmatrix}$$
$$\rho (A) \leq 1.$$

Remark 7. Notice from Theorem 6 that $\|\cdot\|$ is $\|\cdot\|_2$ and $\|\cdot\|_2$ is 2-norm.

Remark 8. Theorem 5 describes the relation of the convergence of such system and time delay, while Theorem 6 describes the relation of the convergence of such system and spectral radius of matrix *A*.

4. Simulations

In this section, we will present numerical simulations to illustrate the theoretical results.

Consider a multiagent system with nine agents and a leader, in which agent 0 is the leader and the rest are followers. The coupling matrix is defined as follows:

By computing, the eigenvalues of matrix L + B are

 $\{0.1277, 0.6451, 0.9012, 1.0373, 1.5498, 1.8522, 2.1573, 2.3954, 2.7340\},\$

pp. 2842-2850, 2014.

- [5] H. Su, M. Z. Q. Chen, J. Lam, and Z. Lin, "Semi-global leaderfollowing consensus of linear multi-agent systems with input saturation via low gain feedback," *IEEE Transactions on Circuits* and Systems I: Regular Papers, vol. 60, no. 7, pp. 1881–1889, 2013.
- [6] H. Hu and Y. Zhu, "Controllability of impulsive fractional dynamical systems with distributed delays," *Electronica Sinica*, vol. 15, no. 4, pp. 34–37, 2012.
- [7] H. Su and M. Z. Chen, "Multi-agent containment with input saturation on switching topologies," *IET Control Theory and Applications*, vol. 9, no. 3, pp. 399–409, 2015.
- [8] B. Liu, H. Su, R. Li, D. Sun, and W. Hu, "Switching controllability of discrete-time multi-agent systems with multiple leaders and time-delays," *Applied Mathematics and Computation*, vol. 228, pp. 571–588, 2014.
- [9] P. J. Torvik and R. L. Bagley, "On the appearance of the fractional derivative in the behavior of real material," *Journal of Applied Mechanics*, vol. 51, no. 2, pp. 294–298, 1984.
- [10] C.-X. Zhu and Y. Zou, "Summary of research on fractionalorder control," *Control and Decision*, vol. 24, no. 2, pp. 161–169, 2009.
- [11] W. Zhang, H. Su, and F. Zhu, "Full-order and reduced-order observer design for one-sided Lipschitz nonlinear systems with unknown inputs," *Nonlinear Dynamics*, vol. 79, no. 2, pp. 1469– 1479, 2015.
- [12] Y. Zhang, Study on Fractionl-Order Calculus and Its Applications, Northeastern University, Shenyang, China, 2008.
- [13] Y. Cao, Y. Li, W. Ren, and Y. Chen, "Distributed coordination of networked fractional-order systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 40, no. 2, pp. 362–370, 2010.
- [14] Y. Cao and W. Ren, "Distributed formation control for fractional-order systems: dynamic interaction and absolute/relative damping," *Systems and Control Letters*, vol. 59, no. 3-4, pp. 233–240, 2010.
- [15] Q. S. Zeng, D. Q. Feng, and G. Y. Cao, "The controllability and observability criteria of systems described by fractional differential equations," *Journal of Zhengzhou University*, vol. 25, no. 1, pp. 66–69, 2004.
- [16] Z. Yanzhu and X. Dingyu, "Modeling and simulating transmission lines using fractional calculus," in *Proceedings of the International Conference on Wireless Communications Networking and Mobile Computing (WiCom '07)*, pp. 3115–3118, IEEE, Shanghai, China, 2007.
- [17] B. Liu, X. Han, J. Zhang, and D. Sun, "Convergence of fractionalorder discrete-time multi-agent systems with a leader," in *Proceedings of the 34th Chinese Control Conference*, Hangzhou, China, July 2015.
- [18] F. Yu, N. Yu, and J. Wang, "Research on controllability for fractional order linear time-invariant systems," *Journal of Chong University of Posts Telecommunications (Natural Science)*, vol. 19, no. 5, pp. 647–648, 2007.
- [19] H.-Y. Yang, S.-W. Tian, and S.-Y. Zhang, "Consensus of multiagent systems with heterogeneous delays and leader-following," *Acta Electronica Sinica*, vol. 39, no. 4, pp. 872–876, 2011.

systems with random initial states and $\alpha = 0.2$, h = 0.4, $\tau = 0.3$, $\alpha = 0.1$, h = 0.4, $\tau = 0.3$, $\alpha = 0.1$, h = 0.4, $\tau = 0.3$, $\alpha = 0.1$, h = 0.4, $\tau = 0.2$, $\alpha = 0.5$, h = 0.6, $\tau = 0.3$, and $\alpha = 0.8$, h = 0.9, $\tau = 0.6$, respectively. Under Theorem 5, Figures 1 and 2 are the convergence situation, and Figure 3 is case of divergence. Under Theorem 6, Figures 4 and 5 are the results of simulations under the conditions of convergence and divergence.

respectively. Figures 1-5 show the trajectories of multiagent

5. Conclusion

In this paper, we have investigated the convergence problem of the fractional-order discrete-time multiagent system with a leader and sampling delay. We have obtained the convergence results depending on the sampling interval h, the fractionalorder α , and the sampling delay.

Conflict of Interests

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

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant no. 61473002, Beijing Natural Science Foundation Program no. 4132021, "The-Great-Wall-Scholar" Candidate Training-Plan of North China University of Technology, Construction Plan for Innovative Research Team of North China University of Technology, the Plan Training Project of Excellent Young Teacher of North China University of Technology, and the Special Project of North China University of Technology (no. XN085).

References

- W. Zhang, H. Su, X. Cai, and H. Guo, "A control lyapunov function approach to stabilization of affine nonlinear systems with bounded uncertain parameters," *Circuits, Systems, and Signal Processing*, vol. 34, no. 1, pp. 341–352, 2015.
- [2] H. Su, G. Jia, and M. Z. Q. Chen, "Semi-global containment control of multi-agent systems with input saturation," *IET Control Theory and Applications*, vol. 8, no. 18, pp. 2229–2237, 2014.
- [3] H.-Y. Yang, L. Guo, Y.-L. Zhang, and X.-M. Yao, "Movement consensus of complex fractional-order multi-agent systems," *Acta Automatica Sinica*, vol. 40, no. 3, pp. 489–496, 2014.
- [4] H. Su, M. Z. Q. Chen, X. Wang, and J. Lam, "Semiglobal observer-based leader-following consensus with input saturation," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 6,

(19)

- [20] H.-Y. Yang, L. Guo, Y.-L. Zhang, and X.-M. Yao, "Delay consensus of fractional-order multi-agent systems with sampling delays," *Acta Automatica Sinica*, vol. 40, no. 9, pp. 2022–2028, 2014.
- [21] K. Lin, "Analysis and comparision of different definition about fractional integrals and derivatives," *Journal of MinJiang University*, vol. 24, no. 5, 2003.
- [22] K. Ogata, Discrete-Time Control System, Prentice-Hall, Englewood Cliffs, NJ, USA, 1995.
- [23] Q. Li, N. Wang, and D. Yi, *Numerical Analysis*, Tsinghua University Press, 5th edition, 2008.

Research Article

Exploring the Impact of Network Structure and Demand Collaboration on the Dynamics of a Supply Chain Network Using a Robust Control Approach

Yongchang Wei,¹ Hongwei Wang,² and Fangyu Chen¹

¹School of Business Administration and the Center of Industrial Economy Research, Zhongnan University of Economics and Law, Wuhan 430073, China

²Institute of Systems Engineering, The Key Laboratory of Image Processing and Intelligent Control of Education Ministry, Huazhong University of Science and Technology, Wuhan 430074, China

Correspondence should be addressed to Hongwei Wang; hwwang@mail.hust.edu.cn

Received 21 August 2015; Accepted 4 October 2015

Academic Editor: Michael Z. Q. Chen

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

A supply chain network might exhibit complex dynamics in the face of increasingly volatile and uncertain environment. The impact of network structure and collaboration on the dynamics and robustness of supply chain network, however, remains to be explored. In this paper, a unified state space model for a two-layer supply chain network composed of multiple distributors and multiple retailers is developed. A robust control algorithm is advocated to reduce both order and demand fluctuations for unknown demand. Numerical simulations demonstrate that the robust control approach has the advantage to reduce both inventory and order fluctuations. In the simulation experiment, it is interesting to notice that complex network structure and collaborations might contribute to the reduction of inventory and order oscillations. This paper yields new insights into the overestimated bullwhip effect problem and helps us understand the complexities of supply chain networks.

1. Introduction

A supply chain network is referred to as a complex network of organizations that synchronizes a series of interrelated business processes, such as procurement, manufacturing, distribution, and retailing, to create values to final customers in the form of products and services [1]. In reality, supply chain network is facing more and more fluctuating market environment and the customer demand is usually badly uncertain due to a lot of factors, such as technology revolution, short product cycles, and promotions. These uncertainties might propagate and further lead to unexpected dynamic behaviors, for example, instability [2–5], bullwhip effect [6-9], and even chaos [10-12], which can further cause huge extra costs due to order and inventory fluctuations [13]. Therefore, it is increasingly important to understand the dynamics and improve the robustness before designing a supply chain network.

It has been found that supply chain network has scalefree and small-world features [14]. Note that the small-world feature and scaleless feature have been widely explored in complex network theory [15-17]. These features indicate that a handful of nodes might dominate the whole network and potentially enhance the capability to respond to uncertainties. In addition, determining the number of upstream suppliers and extending distribution channels to gain more market shares are also important topics in supply chain management. All these facts demonstrate that the topology of supply chain network has substantial impacts on overall performance. In fact, the supply chain network studied in this paper is also a networked control system. Similar to the literature stream of networked control system [18-20], the network structure is a significant factor that affects the control performance. In addition, the central task for the supply chain network or the networks studied in the previous literature is how to design an appropriate controller to ensure stability and satisfy specific requirements. It is worthwhile to notice that a supply chain network is usually composed of heterogeneous nodes, for example, retailers, distributors, manufacturers, and raw material suppliers. In a supply chain network, distributors usually exchange information, for example, placing and receiving orders, with nodes inside the network. Unlike distributors, retailers, however, must fulfill customer demands from external volatile environment. For this reason, the dynamics of heterogeneous nodes should be treated or described differently. Furthermore, there exist a lot of interaction activities among nodes in supply chain network, for example, cooperations and competitions. As a result, all these features, complex topologies caused by complex supply relationships, heterogeneous nodes, and interactions, make supply chain network increasingly complex, hard to describe, and challenging for exploring dynamics and robustness.

It is not surprising that the dynamics and robustness of supply chain system have received considerable attention over the past decade. Control engineering technologies are frequently used to analyze stability [2, 21, 22] and study the bullwhip effect [7, 8, 23–25], which refers to the amplification of order fluctuations as one moves up a supply chain from downstream to upstream [6, 26]. In particular, robust control methods are well used in analyzing the bullwhip effect for known inventory rules [9, 27], or designing inventory rules for unknown demand using feedback control method [28]. Some "fluid models" and simulation models are also proposed to study the dynamic properties of supply chain systems [3, 10, 29, 30].

The majority of the previous studies focus on serial supply chains. Most supply chain systems, however, have a network topology in which each entity's decision may be influenced directly or indirectly by other nodes. More importantly, demand may arise from many customers at different locations of the supply chain network, and the interactions among nodes should be addressed. Unfortunately, few studies are undertaken to address the dynamics, robustness, and the impact of collaborations on dynamics in the context of supply chain network. For example, Helbing [31] proposes a rather general model of supply chain network and connected it to queueing theory and macroeconomics. Scholz-Reiter et al. [32] presented an approach to calculate the stability condition by mathematical system theory to guarantee stability for production networks, to identify a stability region, and to refine the region by discrete event simulations. Ouyang and Li [33] study the bullwhip effect propagation in a supply chain network with linear and time-invariant inventory management policies using a frequency domain control method. They obtained conditions to predict the existence of bullwhip effect for unknown demand. Recently, Dominguez et al. [34] study the bullwhip effect of a divergent supply chain system with simulation experiment. They begin to notice that complex structure of supply chain might yield different bullwhip effect conclusions.

From the above, the dynamics of supply chain network has attracted the attention in the very recent years. The impact of network structure and collaboration on supply chain dynamics still remain unclear. The contributions of this paper are as follows. A unified state space model is developed to model the dynamic for a supply chain network composed of heterogeneous nodes. To improve the robustness for unknown demand, a robust H_{∞} control approach is advocated to obtain optimal inventory rules. The time domain robust control method facilitates us to obtain the optimal replenishment policy under well-defined performance measures, in order to reduce both order and inventory fluctuations. There is a trade-off between reducing inventory variations and mitigating the bullwhip effect [23]. The effectiveness and superiority of the proposed method are numerically validated for several demand processes. Furthermore, we incorporate the network topology and demand collaboration between retailers into the dynamic model, which enables us to numerically study the impacts of network structure and demand collaboration on dynamics improvement. We show that both demand collaboration and complex supply relationship might contribute to dynamics improvement. The results obtained in this research provide new insights into supply chain management, especially for the overestimated bullwhip effect problem [35, 36], via considering complex supply relationship and demand collaboration.

The structure of this paper is arranged as follows. Section 2 describes the model and Section 3 introduces a robust metric and the optimization algorithm to calculate the minimal value of the robust metric. In Section 4, we numerically validate the effectiveness of the robust control method and also study the influences of network structure and demand collaboration on dynamics and robustness.

2. Modeling the Supply Chain Network

Consider a two-layer supply chain network constituted with m distributors and n retailers. The structure of supply chain network is shown in Figure 1. Without loss of generality, we impose the following assumptions on the supply chain network:

- (i) All the facilities, including retailers and distributors, have the same events order in each period, in order to construct a discrete-time state space model. They place replenishment orders at the beginning of each period and fulfill downstream orders or demands during the remaining time.
- (ii) Each retailer might choose to place orders from many distributors and each distributor might satisfy the order from multiple retailers. All the distributors place orders from external suppliers.
- (iii) The customer demands of retailers that come from external market are uncertain. All the nodes in the supply chain network follow the same way to deal with shortages; namely, unfulfilled orders or demands will be backlogged.
- (iv) Both the distributors and retailers have fixed replenishment lead times.

Denote symbols *i*, $1 \le i \le m$, and *j*, $1 \le j \le n$, as the *i*th distributor and the *j*th retailer, respectively. Then the supply relationship between retailers and distributors could



FIGURE 1: The structure of a general two-layer supply chain network.

be defined by a *supply relationship matrix* $L = (l_{ij})_{m \times n} \in \mathbb{R}^{m \times n}$. If $l_{ij} = 1$, the distributor *i* and the retailer *j* establish the supply relationship between them. Otherwise, if $l_{ij} = 0$, there is no relationship between them. In fact, the supply relationship matrix reflects the network topology. For example, if each column of the matrix *L* has only one element with value 1, then each retailer has only one immediate supplier. In contrast, if $l_{ij} = 1$ ($\forall i, j$), then each distributor supplies goods to every retailer, which represents the most complex network structure.

Suppose that the retailers in the same market can collaborate with each other, and consider the collaboration mechanism via demand diversion in [33]. Let $d_i(t)$ be the demand received by retailer j. Then the amount $\delta_{j_1j_2}$ × $[d_{j_1}(t) - d_{j_2}(t)]$ of the received demand of retailer j_1 will be satisfied by retailer j_2 if $d_{j_1}(t) > d_{j_2}(t)$. Note that the parameter $\delta_{j_1 j_2}$ can be negotiated by each pair of retailers by considering some factors such as geographic location and demand characteristics. As a result, we can define a *collaboration matrix* $H \triangleq (\delta_{j_1 j_2})_{n \times n} \in \mathbb{R}^{n \times n}$ to exactly describe the demand diversion among retailers in the supply chain network. Assume that the demand of retailer j_1 is $d_{j_1}(t)$ and the demand of retailer j_2 is $d_{j_2}(t)$. The essential demand of retailer j_1 after demand collaboration becomes $d_{j_1}(t)$ + $\delta_{j_1j_2}[d_{j_2}(t) - d_{j_1}(t)]$. The demand of retailer j_2 is changed to $d_{j_2}(t) + \delta_{j_2j_1}[d_{j_1}(t) - d_{j_2}(t)]$. Then we must have $\delta_{j_1j_2} = \delta_{j_2j_1}$ without losing any customer demand. In other words, the collaboration matrix is symmetric in order to balance the shifted demand between each pair of retailers.

As the supply chain network is composed of heterogeneous nodes, the dynamic equations for the retailers and distributors should be described separately. We will now first consider the inventory dynamics for distributors. Let $w_i(t)$, $1 \le i \le m$, denote the inventory position of the distributor *i*, let $u_i(t)$ denote the placed order by distributor *i* from external suppliers, and let $u_{ij}(t)$ denote the received order of distributor *i* from retailer *j* ($1 \le j \le n$). Then, the resulting dynamics of the distributor *i* can be represented by

$$w_{i}(t+1) = w_{i}(t) + u_{i}(t) - \sum_{j=1}^{n} l_{ij}u_{ij}(t).$$
(1)

All the retailers not only place orders from their upstream distributors but also fulfill external customer demand. In addition, a retailer might divert its received demand to its collaboration partner. Let $r_j(t)$, $1 \le j \le n$, denote the inventory position of retailer j; then its balanced inventory equation could be obtained as

$$r_{j}(t+1) = r_{j}(t) + \sum_{i=1}^{m} l_{ij}u_{ij}(t) - d_{j}(t) - \sum_{i_{j}=1}^{n} \delta_{jj_{1}} \left[d_{j_{1}}(t) - d_{j}(t) \right].$$
(2)

It is worthwhile to note that as we select inventory position as the state variable, the lead time disappeared in the dynamics equation. In fact, inventory position is a comprehensive variable that incorporates on-hand inventory, backorder, and on-road inventory simultaneously.

In order to develop a unified state space model, we will now transform the supply relationship matrix L and the collaboration matrix H. By doing this, we can incorporate the network structure and collaboration among retailers into the dynamic model. Let $L_j = [l_{1j}, l_{2j}, \ldots, l_{mj}]'$, $1 \le j \le n$, define the *j*th column of the matrix L. Then we could define the following two matrixes:

$$M_1 \triangleq \operatorname{diag}\left(L'_1, L'_2, \dots, L'_n\right),$$

$$M_2 \triangleq \begin{bmatrix}G_1 & G_2 & \cdots & G_n\end{bmatrix},$$
(3)

where $G_j = \text{diag}(l_{1j}, l_{2j}, \dots, l_{mj}), 1 \le j \le n$. We see that $M_1 \in \mathbb{R}^{n \times mn}$ and $M_2 \in \mathbb{R}^{m \times mn}$. Based on the collaboration matrix H, we could obtain the following matrix:

$$H_{L} \triangleq \begin{bmatrix} 1 - \sum_{j \neq 1} \delta_{1j} & \delta_{12} & \cdots & \delta_{1n} \\ \delta_{21} & 1 - \sum_{j \neq 2} \delta_{2j} & \cdots & \delta_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ \delta_{n1} & \delta_{n2} & \cdots & 1 - \sum_{j \neq n} \delta_{nj} \end{bmatrix}.$$
 (4)

The demand rate of retailer *j* is assumed to take the form

$$d_j(t) = d_j^e + \xi_j(t), \qquad (5)$$

where d_j^e is a known constant and $\xi_j(t)$ denotes the energy bounded disturbance with mean 0. The demand in Ouyang and Daganzo [9] and Ouyang and Li [33] also has the characteristics of the above demand pattern. If the supply chain network is stable and $d_j(t) = d_j^e$, j = 1, 2, ..., n, then the inventory positions and order quantities of retailers and distributors will return to constant or steady values after finite periods. Define w_i^e and r_j^e as the steady values of inventory position of distributor *i* and retailer *j*, respectively. Let u_i^e and u_{ij}^e denote the steady values of the amount of order of the distributor *i* placed from external suppliers and the amount of order of the retailer *j* placed from distributor *i*, respectively. In such case, we can have $\lim_{t\to\infty} w_i(t) = w_i^e$, $\lim_{t\to\infty} r_j(t) =$ r_j^e , $\lim_{t\to\infty} u_i(t) = u_i^e$, and $\lim_{t\to\infty} u_{ij}(t) = u_{ij}^e$.

It is well-known that inventory rules play significant roles in determining supply chain dynamics. In fact, inventory rules are essentially feedback controllers using state information, for example, inventory position. Define the following state vector: $\tilde{x}(t) = [\tilde{w}(t) \ \tilde{r}(t)]'$, where

$$\widetilde{w}(t) = [\widetilde{w}_{1}(t), \widetilde{w}_{2}(t), \dots, \widetilde{w}_{m}(t)]$$

$$= [w_{1}(t) - w_{1}^{e}, w_{2}(t) - w_{2}^{e}, \dots, w_{m}(t) - w_{m}^{e}],$$
(6)
$$\widetilde{r}(t) = [\widetilde{r}_{1}(t), \widetilde{r}_{2}(t), \dots, \widetilde{r}_{n}(t)]$$

$$= [r_{1}(t) - r_{1}^{e}, r_{2}(t) - r_{2}^{e}, \dots, r_{n}(t) - r_{n}^{e}].$$

The input vector can be represented by

$$\widetilde{d}(t) = \left[\widetilde{d}_{1}(t), \widetilde{d}_{1}(t), \dots, \widetilde{d}_{n}(t)\right]'
= \left[d_{1}(t) - d_{1}^{e}, d_{2}(t) - d_{2}^{e}, \dots, d_{n}(t) - d_{n}^{e}\right]' \in \mathbb{R}^{n}.$$
(7)

Before defining the control vector, we will now introduce the operator Vec(A).

Definition 1. Assume that $A \in \mathbb{R}^{m \times n}$ is a matrix with *m* rows and *n* columns and $A = (A_1, A_2, \dots, A_n)$, where A_i is the *i*th column of *A*. Then we can produce a single column vector using the following operator, which is given by

Vec
$$(A) = [A'_1, A'_2, \dots, A'_n]' \in \mathbb{R}^{mn}.$$
 (8)

For the supply chain network, the control vector $\tilde{u}(t)$ can be represented as $\tilde{u}(t) = \begin{bmatrix} \tilde{u}_w(t) & \tilde{u}_r(t) \end{bmatrix}' \in \mathbb{R}^{m+mn}$, where

$$\widetilde{u}_{w}(t) = [\widetilde{u}_{1}(t), \widetilde{u}_{2}(t), \dots, \widetilde{u}_{m}(t)]
= [u_{1}(t) - u_{1}^{e}, u_{2}(t) - u_{2}^{e}, \dots, u_{m}(t) - u_{m}^{e}], \quad (9)
\widetilde{u}_{r}(t) = \operatorname{Vec}(M_{R}(t))',$$

where

$$M_{R}(t) = \begin{bmatrix} \tilde{u}_{11}(t) & \tilde{u}_{12}(t) & \cdots & \tilde{u}_{1n}(t) \\ \tilde{u}_{21}(t) & \tilde{u}_{22}(t) & \cdots & \tilde{u}_{2n}(t) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \tilde{u}_{m1}(t) & \tilde{u}_{m2}(t) & \cdots & \tilde{u}_{mn}(t) \end{bmatrix}$$

$$= \begin{bmatrix} u_{11}(t) - u_{11}^{e} & u_{12}(t) - u_{12}^{e} & \cdots & u_{1n}(t) - u_{1n}^{e} \\ u_{21}(t) - u_{21}^{e} & u_{22}(t) - u_{22}^{e} & \cdots & u_{2n}(t) - u_{2n}^{e} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ u_{m1}(t) - u_{m1}^{e} & u_{m2}(t) - u_{m2}^{e} & \cdots & u_{mn}(t) - u_{mn}^{e} \end{bmatrix}$$

$$(10)$$

$$\in R^{m \times n}.$$

Define $\tilde{y}(t) \in \mathbb{R}^{2m+n+mn}$ as system output and let I_x denote the unit matrix with x dimensions. We then can develop the following unified state space model:

$$\widetilde{x}(t+1) = \widetilde{x}(t) + B\widetilde{u}(t) + B_w \widetilde{d}(t),$$

$$\widetilde{y}(t) = C\widetilde{x}(t) + D\widetilde{u}(t),$$
(11)

where

$$B = \begin{bmatrix} I_m & -M_2 \\ 0 & M_1 \end{bmatrix} \in R^{(m+n)\times(m+mn)},$$

$$B_w = \begin{bmatrix} 0 \\ -H_L \end{bmatrix} \in R^{(m+n)\times n},$$

$$C = \begin{bmatrix} \chi_i I_{m+n} & 0 \\ 0 & 0 \end{bmatrix} \in R^{(2m+n+mn)\times(m+n)},$$

$$D = \begin{bmatrix} 0 & 0 \\ 0 & \chi_o I_{m(n+1)} \end{bmatrix} \in R^{(2m+n+mn)\times(m+mn)},$$
(12)

in which χ_i and χ_o are variables with either value 1 or value 0 to define system output according to practical situations. For example, if managers only care for inventory fluctuations, they can set $\chi_i = 1$ and $\chi_o = 0$. Otherwise, the setting of $\chi_i = 0$ and $\chi_o = 1$ can be used to mitigate bullwhip effect or order fluctuations. Both the inventory and order dynamics will be considered if $\chi_i = 1$ and $\chi_o = 1$. Based on the state space model, we can design inventory rules to improve the dynamics and robustness of the supply chain network for unknown demand with robust H_{∞} control method.

3. The Robust Performance Metric and H_{∞} Control Method

The bullwhip effect, one kind of typical dynamic behavior, is of great significance and has been investigated extensively. The robustness of bullwhip effect, however, has been addressed recently only in a few literatures [9, 27, 28]. Meanwhile, most of the existing studies on the bullwhip effect pay major attention to order fluctuations but inventory fluctuations are neglected. To this end, this research focuses

Mathematical Problems in Engineering

on improving the dynamics and robustness of supply chain system for unknown demand with a robust H_{∞} control approach with the aim to reduce both inventory and order fluctuations.

Based on the state space model (11), we will now consider the following robust metric:

$$W_{I} \triangleq \sup_{\forall \tilde{d}(t) \neq 0} \left[\frac{\sum_{k=0}^{\infty} \tilde{y}(t)' \tilde{y}(t)}{\sum_{k=0}^{\infty} \tilde{d}(t)' \tilde{d}(t)} \right]^{1/2}.$$
 (13)

We see that the metric W_I is the amplification metric of the worst-case root mean square errors (RMSE) when the customer demand is unknown. Therefore, minimizing the value of W_I can effectively improve the dynamics and robustness of supply chain system. It is worthwhile to mention that this metric, also called L_2 gain or H_{∞} norm, has been widely used in the field of control engineering. Although it is not

an economic metric, the metric W_I is intimately related to fluctuations of order and inventory in supply chain systems. For example, Boukas [28] pointed out that reducing the value of the metric W_I increases the speed of a simple production control system to respond to customer demand. Ouyang and Daganzo [9] directly used this metric to quantify the bullwhip effect for unknown demand.

From the form of formula (13), we see that reducing the value of metric W_I reduces the fluctuation of system output for unknown input. Thus, we can reduce order oscillations if we select placed orders as system output by setting $\chi_i = 0$ and $\chi_o = 1$. However, in most cases, inventory fluctuations and order fluctuations should be balanced. Unlike the traditional bullwhip effect metric, which mainly focuses on dampening order fluctuations, the metric W_I can take both order fluctuations and inventory fluctuations into account by setting $\chi_i = 1$ and $\chi_o = 1$. In this case, the exact formula of W_I for our supply chain network model becomes

$$W_{I} = \sup_{\forall \tilde{d}(t) \neq 0} \left\{ \frac{\sum_{i=1}^{\infty} \left\{ \sum_{i=1}^{m} \left[\widetilde{w}_{i}(t)^{2} + \widetilde{u}_{i}(t)^{2} \right] + \sum_{j=1}^{n} \left[\widetilde{r}_{j}(t)^{2} + \sum_{i=1}^{m} \widetilde{u}_{ij}(t)^{2} \right] \right\}}{\sum_{i=1}^{\infty} \sum_{j=1}^{n} \widetilde{d}_{j}(t)^{2}} \right\}^{1/2}.$$
(14)

As emphasized, minimizing the value of the metric W_I helps improving dynamics for unknown demand. Here we focus on designing appropriate inventory rules to achieve this goal. In this research, the inventory rule for the supply chain network can be represented in the form of state feedback controller, which is given by $\tilde{u}(t) = K\tilde{x}(t)$. Then the robust H_{∞} control problem of our supply chain network model is to seek a feedback controller $\tilde{u}(t) = K\tilde{x}(t)$ that satisfies the following optimization problem:

$$\min_{\forall \tilde{u}(t)=K\tilde{x}(t)} W_I.$$
(15)

The inventory rule to minimize the value of the robust metric W_I for the state space model (11) can be obtained by Theorem 2. The proof process refers to Kim and Park [37].

Theorem 2. Consider the discrete-time system (11). If there exists a positive-definite matrix Q and a matrix M satisfies the following optimization problem:

$$\min \gamma^{1/2} \tag{16}$$

subject to

$$\begin{bmatrix} -Q & Q + BM & B_{w} & 0 \\ Q + M'B' & -Q & 0 & QC' + M'D' \\ B'_{w} & 0 & -\gamma I_{n} & 0 \\ 0 & CQ + DM & 0 & -I_{(2m+n+mn)} \end{bmatrix} < 0,$$
(17)
 $\gamma > 0,$

then system (11) is quadratically stable and the minimal value of the robust metric W_I can be approximated with the minimized $\gamma^{1/2}$, with the state-feedback controller: $\tilde{u}(t) = MQ^{-1}\tilde{x}(t)$.

The importance of Theorem 2 lies in two aspects. Firstly, the stability of the whole supply chain network is warranted if the optimization problem in Theorem 2 is solvable. Secondly, it produces an inventory rule that minimizes the metric W_I , or the H_{∞} norm, for unknown demand to reduce output oscillations. It also should be noted that the optimization problem can be readily solved using the standard numerical software, MATLAB, which enables us to numerically study the impacts of demand collaborations and network structures on dynamics and robustness.

4. Numerical Analysis

As introduced, the proposed robust H_{∞} control method is expected to reduce both order and inventory fluctuations, which will be numerically validated in the following section. Meanwhile, as the network topology and collaboration among retailers have been incorporated into the unified state space model, it enables us to study the influences of network structures and collaborations on the improvement of dynamics and robustness.

Traditionally, inventory and order fluctuations haven been measured with variance or standard deviation. For example, the bullwhip effect can be quantified with the ratio of variance between demand and order [26]. Here we consider the following four metrics in the numerical analysis:

$$W_{b1} = \sum_{i=1}^{m} \operatorname{Var}\left(\widetilde{u}_{i}(t)\right),$$
$$W_{b2} = \sum_{i=1}^{m} \sum_{j=1}^{n} \operatorname{Var}\left(\widetilde{u}_{ij}(t)\right),$$

$$W_{s1} = \sum_{i=1}^{m} \operatorname{Var}\left(\widetilde{w}_{i}\left(t\right)\right),$$
$$W_{s2} = \sum_{j=1}^{n} \operatorname{Var}\left(\widetilde{r}_{j}\left(t\right)\right),$$
(18)

in which $Var(\cdot)$ is used to calculate the variances of time sequences. It is quite apparent to see that the metrics W_{b1} and W_{b2} , respectively, represent the order fluctuations at distributor layer and retailer layer. The latter two metrics W_{s1} and W_{s2} are used to measure the inventory oscillations for retailers and distributors.

4.1. Effectiveness Validation for the Robust Control Method. Consider a supply chain network composed of 3 retailers and 3 distributors. Without loss of generality, we assume that there exists no collaboration among retailers and each retailer places order from all the three distributors. The supply relationship matrix and collaboration matrix can be obtained as

$$L = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},$$

$$H = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$
(19)

Assume that the demand of each retailer follows the firstorder autoregressive process:

$$\tilde{d}_{j}(t+1) = \rho \tilde{d}_{j}(t) + (1-\rho)\varepsilon_{t}, \quad j \in \{1, 2, 3\},$$
(20)

in which ρ is the autoregressive parameter and ε_t is the white noise. Setting different values for χ_i and χ_o , we can consider three different control ways:

- (i) control way 1: to reduce inventory fluctuation ($\chi_i = 1$, $\chi_o = 0$);
- (ii) control way 2: to reduce order fluctuation ($\chi_i = 0$, $\chi_o = 1$);
- (iii) control way 3: to reduce both order and inventory fluctuations (χ_i = 1, χ_o = 1).

Using the optimization algorithm in Theorem 2, we can obtain three different inventory rules for three control ways. Take the third control way as an example. In such a parameter setting, we have $\chi_i = 1$, $\chi_o = 1$. As we consider a supply chain network composed of 3 retailers and 3 distributors, we could have n = m = 3. Then, the matrices M_1 and M_2 can be obtained as

$$M_{1} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix},$$

$$M_{2} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}.$$
(21)

Because we do not consider demand collaboration in this section, the matrix H_L will be calculated as $H_L = \text{diag}\{1, 1, 1\}$. From the value of M_1 , M_2 , and H_L , we can further easily calculate the matrices B, B_w , C, and D. The inventory rule, a state-feedback controller, can be solved using Theorem 2 with LMI Toolbox in MATLAB, which is represented as

$$\tilde{u}(t) = \begin{bmatrix} -0.4140 & -0.1911 & -0.1911 & -0.2656 & -0.2656 & -0.2656 \\ -0.1911 & -0.4140 & -0.1911 & -0.2656 & -0.2656 & -0.2656 \\ -0.1911 & -0.1911 & -0.4140 & -0.2656 & -0.2656 & -0.2656 \\ 0.1450 & -0.0779 & -0.0779 & -0.3201 & -0.0058 & -0.0058 \\ -0.0779 & 0.1450 & -0.0779 & -0.3201 & -0.0058 & -0.0058 \\ 0.1450 & -0.0779 & 0.1450 & -0.3201 & -0.0058 & -0.0058 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 & -0.0058 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 & -0.0058 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 & -0.0058 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 & -0.0058 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 & -0.0058 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 & -0.0058 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 \\ -0.0779 & 0.1450 & -0.0779 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.1450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.01450 & -0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.0779 & 0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.00779 & 0.0058 & -0.0058 & -0.3201 \\ -0.0779 & 0.0779 & 0.0779 & 0.0058 & -0.00$$

The inventory and order dynamics of the supply chain network with $\rho = 0.6$ are shown in Figures 2 and 3. With

a simulation length of 15000 periods and by using a standard numerical software, we can calculate the values of the four



FIGURE 2: Dynamic curves of inventory under three different control ways.

metrics, W_{b1} , W_{b2} , W_{s1} , and W_{s2} , which are summarized in Table 1.

From Table 1 and Figures 2 and 3, we see that the first control way has an obvious effect on reducing inventory fluctuations. However, it inevitably results in huge order fluctuations. The second control way, in contrast, behaves well in reducing order fluctuations at the expense of high inventory fluctuations. We see that there is an obvious tradeoff between improving inventory dynamics and damping order fluctuations. In fact, balancing inventory fluctuations and order fluctuation has been recognized as an important problem in the literature [23].



FIGURE 3: Dynamic curves of order under three different control ways.

TABLE 1: System	performance	under different	control	ways.
-----------------	-------------	-----------------	---------	-------

	W_{b1}	W_{b2}	W_{s1}	W _{s2}
Control way 1	3.5782×10^{3}	6.2078×10^{3}	1.9027×10^{-27}	0.7570
Control way 2	9.8163×10^{-5}	2.4236×10^{-4}	0.4375×10^{3}	1.1454×10^3
Control way 3	0.2147	0.0810	0.0141	0.7890



FIGURE 4: Four types of supply chain network structures.

In practice, both inventory and order fluctuations greatly enhance difficulties for supply chain management. For example, order fluctuations bring difficulties for arranging labor force and making full use of production capacity. Meanwhile, inventory fluctuations should be rigidly controlled for distributors because high costs will incur if items cannot be sold out timely. It will be better if we can find inventory rules that reduce both inventory and order fluctuations. From the third row in Table 1, we see that the third control way performs well in reducing both inventory and order fluctuations which have been decreased. Figures 2(c) and 3(c) further confirm the advantage of the third control way with the demand process (20). In this sense, it reveals that the robust H_{∞} control method is highly effective in improving dynamics and robustness.

4.2. Impact of Network Structure on Dynamics and Robust Performance. The structure of supply chain network can be altered as a result of a lot of activities, such as adding new suppliers and extending distribution channels. These activities certainly impact the dynamics and robustness of supply chain network. For this purpose, the following study will be conducted with two concerns. We firstly consider the supply relationship, which is described by the supply relationship matrix L, and then consider the change of the number of nodes. Assume that $\chi_i = 1$ and $\chi_o = 1$ to consider both inventory and order fluctuations. To concentrate our attention on network structure, we will not take demand collaboration into account.

4.2.1. The Impact of Supply Relationship on System Performance. It is assumed that each retailer has the flexibility to choose its suppliers. Each retailer can place orders from multiple distributors; each retailer can also choose a single distributor as its supplier. Here we consider a supply chain network composed of 3 retailers and 6 distributors. Four types of supply are considered as shown in Figure 4. The first structure is the simplest one, in which each retailer has only one supplier. The second structure is a distribution system with a major supplier. The latter two structures represent supply chain network with more complex supply relationships. The fourth structure, in which each retailer places orders from all the distributors, is the most complex structure in this study. The supply relationship matrix Lcan be varied to represent different structures. The supply relationship matrixes for the four types of structures are given by

where L_1 , L_2 , L_3 , and L_4 are the supply relationship matrices for structure 1, structure 2, structure 3, and structure 4, respectively.

Using the optimization method introduced in Theorem 2, we could accordingly compute the minimal values of the robust metric W_I for the supply chain network under four different structures; they are 2.2396, 2.2691, 1.9519, and 1.8361. It is obvious that the structure of supply chain network has significant influences on dynamics and robustness.

We use the demand process (20) with $\rho = 0.6$ to generate demand sequences for retailers to observe the dynamics of order and inventory. The order dynamics of the three distributors are illustrated in Figure 5. Figure 5 shows that, compared with structure 1 and structure 2, structure 3 and structure 4 improve the dynamics for the three distributors. It means that complex supply chain network structure might help reduce order dynamics, namely, reducing the bullwhip effect. However, in the practice, we should consider other factors, such as distance between two nodes and information sharing costs. As the distance is increased, the transportation cost might be greatly increased. The second structure, which reflects a multiple-retailer system, results in high order fluctuations. In fact, too many downstream nodes can increase the risk of shortage, which has been recognized as one of



FIGURE 5: Order dynamics under different structures of supply chain network.

the main causes of bullwhip effect [6], in the face of limited supply resources. Furthermore, since all the retailers should satisfy uncertain demands from external environment, too many retailers largely increase the uncertainties for the whole network and further lead to poor performance.

4.2.2. The Impact of the Number of Nodes on Robust Performance. As the supply chain network is hierarchical, we can change the number of upstream nodes or downstream nodes. Here we consider two cases:

- (i) Case 1: increasing the number of distributors.
- (ii) Case 2: increasing the number of retailers.

We assume that there exist two retailers in the first case and two distributors in the second case. The first case characterizes the supply chain system with multiple suppliers as the number of distributors increases. The second case represents distribution system with multiple retailers. It is obvious that increasing the number of distributors, as shown in Figure 6(a), has a positive effect on robustness improvement. This result can be explained where increasing the number of suppliers provides new purchasing channels or opportunities and thus the possibility of shortages will be lowered. From Figure 6(b), we see that increasing the number of retailers results in poor robust performance. This result confirms the result obtained before that a distribution system usually behaves poor on robustness performance.

From the above, it reveals that the structure of supply chain network is closely related to dynamics and robustness and thus seeking for optimal structures is essential for supply chain networks to achieve high economic benefits. Most importantly, our study adds new insights into the overestimated bullwhip effect problem [35, 36]. A recent study of US industry level data found that demand volatility does not increase as one moves up the supply chain [35]. The author observed that, in general, manufacturers do not have substantially greater demand volatility than retailers. However, traditional researches on the bullwhip effect believe that order fluctuations will be amplified stage by stage [6, 26]. In other words, manufacturers, probably the most upstream nodes in a supply chain network, should experience the worst order fluctuations. It seems that the findings in [35] contradict the results in traditional researches. In this research, we found that a supply chain network with complex structures



FIGURE 6: Impact of the number of nodes on robust performance.

behaves well in dynamics and robustness compared to simple networks. We also found that increasing the number of distributors contributes to the bullwhip mitigation. For this reason, we argue that the existing approaches that aim at quantifying the bullwhip effect neglect the network structure of supply chains. We believe that if we begin to study the bullwhip effect of supply chain network with the consideration of complex supply relationships and collaborations, not just serial supply chains, new results about the bullwhip effect might emerge.

4.3. Impact of Demand Collaboration on Robust Performance. As introduced, the collaboration mechanism among retailers is characterized by the collaboration matrix H. The retailer j_1 and retailer j_2 form their collaboration relationship with the parameter δ_{j_1,j_2} , $0 \le \delta_{j_1,j_2} \le 1$, which reflects the strength of collaboration. We will now consider a supply chain network with 3 retailers and 6 retailers. In many circumstances, one retailer tends to collaborate with a single retailer geographically close to it. To facilitate analysis, we assume that each retailer has only one collaboration partner. Then the corresponding collaboration matrix is represented as

$$H = \begin{bmatrix} 0 & \delta_{12} & 0 & 0 & 0 & 0 \\ \delta_{21} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \delta_{34} & 0 & 0 \\ 0 & 0 & \delta_{43} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \delta_{56} \\ 0 & 0 & 0 & 0 & \delta_{65} & 0 \end{bmatrix},$$
(24)

in which δ_{12} , δ_{34} , and δ_{56} are the collaboration parameters established by each pair of retailers. In addition, we assume that each retailer places order from all the suppliers. In other words, the values of all the elements of matrix *L* are 1. We also assume that $\chi_i = 1$ and $\chi_o = 1$ in order to consider overall performance. In this study, we will discuss three different collaborating scenarios:

- (i) scenario 1: $\delta_{12} = \delta_{34} = \delta_{56} = 0$ (without cooperation);
- (ii) scenario 2: $\delta_{12} = \delta_{34} = \delta_{56} = 0.25$ (mild cooperation);
- (iii) scenario 3: $\delta_{12} = \delta_{34} = \delta_{56} = 0.5$ (close cooperation).

Assume that retailer j_1 collaborates with retailer j_2 , $\forall (j_1, j_2) \in \{(1, 2), (3, 4), (5, 6)\}$; then their demand processes are assumed to be

$$\begin{split} \tilde{d}_{j_{1}}(t+1) &= \rho \tilde{d}_{j_{1}}(t) + (1-\rho) \varepsilon_{j_{1}}(t) + \eta_{j_{1}} \varepsilon(t) ,\\ \tilde{d}_{j_{2}}(t+1) &= \rho \tilde{d}_{j_{2}}(t) + (1-\rho) \varepsilon_{j_{2}}(t) + \eta_{j_{2}} \varepsilon(t) , \end{split}$$
(25)

where $\rho = 0.2$, η_{j_1} , $\eta_{j_2} \in \{-1, 0, 1\}$, with white noises $\varepsilon_{j_1}(t)$ and $\varepsilon_{j_2}(t)$ with mean 0 and variance 1. A variety of auto- and cross-correlations among the demands can be represented by the selection of parameters. For example, the two demands may be independent of each other ($\eta_{j_1} = \eta_{j_2} = 0$), positively correlated ($\eta_{j_1} = \eta_{j_2} = 1$), or negatively correlated ($\eta_{j_1} = -\eta_{j_2} = 1$). The minimal values of the robust metric W_I under

The minimal values of the robust metric W_I under the above three different collaboration scenarios can be computed by Theorem 2, which are 1.8361, 1.8686, and 1.8751. It shows that the collaboration behaviors have little impacts on the robustness improvement. However, we will see that the collaboration behaviors among retailers may mitigate

TABLE 2: System performance under different collaboration scenarios.

		W_{b1}	W_{b2}	W_{s1}	W_{s2}
	$\eta_{j1} = \eta_{j2} = 0$	1.0654	0.3993	0.0432	4.1398
Scenario 1	$\eta_{j1}=\eta_{j2}=1$	4.3605	1.0635	0.1767	10.5216
	$\eta_{j1} = -\eta_{j2} = 1$	1.0423	1.0057	0.0422	10.5494
Scenario 2	$\eta_{j1} = \eta_{j2} = 0$	1.0352	0.2482	0.0436	2.5674
	$\eta_{j1}=\eta_{j2}=1$	4.3898	0.9023	0.1849	9.0700
	$\eta_{j1}=-\eta_{j2}=1$	1.0253	0.3944	0.0431	4.2528
Scenario 3	$\eta_{j1}=\eta_{j2}=0$	1.0089	0.2084	0.0445	2.0619
	$\eta_{j1}=\eta_{j2}=1$	4.1481	0.8425	0.1827	8.4610
	$\eta_{j1} = -\eta_{j2} = 1$	1.0093	0.2055	0.0445	2.0680

the bullwhip effect for some special demand processes, as demonstrated in the following simulations. Under the demand processes formulated in (25), we can obtain the values of the four metrics, W_{b1} , W_{b2} , W_{s1} , and W_{s2} , via simulation with a length of 15000 periods, which are shown in Table 2. From Table 2, we see that the demand collaboration between retailers, or demand collaboration, has obviously reduced order and inventory fluctuations. Interestingly, when the values of collaboration parameters increase, the fluctuations of inventory and order decrease, and the reduction is the most significant for the case of negative cross-correlation. In addition, the reduction of inventory and order fluctuations for retailers is more obvious than distributors. This can be explained by the fact that the collaborations are established among retailers, not distributors.

In summary, collaborations among retailers might not contribute to robustness improvement; however, collaborations can reduce inventory and order fluctuations for some specific demand processes, especially for retailers. When [33] discussed the collaboration among retailers, they obtained similar results, in which they focus on order fluctuations in term of bullwhip effect. In this sense, our results confirm and also enrich the results in [33] using the robust H_{∞} control.

5. Conclusions

This paper has investigated the dynamics and robustness of a two-layer supply chain network constituted with multiple retailers and multiple distributors. A unified state space model is developed to characterize some basic characteristics of the supply chain network. Specifically, we describe different supply relationship with supply relationship matrix and consider one kind of demand collaboration mechanism among retailers, namely, demand diversion. As the nodes of supply chain network are heterogeneous, the dynamic equations for retailers and distributors are developed separately. Considering these characteristics enables us to investigate the influences of network structure and collaborations on dynamics and robustness.

It is well known that reducing order fluctuations usually increases inventory fluctuations [23], which tends to result in shortages and high levels of safety stock. This research, however, numerically showed that the robust H_{∞} control

method based on the state space model has the advantage to reduce both inventory and order fluctuations for supply chain network. The simulation results show that complex supply relationships and demand collaborations among retailers contribute to the reduction of inventory and order fluctuations for unknown demand or specific demand. Collaborations among retailers might not improve the robustness of a supply chain network. However, collaborations can reduce inventory and order oscillations for some specific demand processes, especially for retailers. We argue that if we begin to study the bullwhip effect of supply chain network with the consideration of complex supply relationships and collaborations, new results about the bullwhip effect might emerge.

Finally, it is worthwhile to point out that the investigation of the dynamics and robustness of supply chain network, as we believe, is at the beginning stage. A lot of problems deserve future research. For example, if we consider more complex inventory rules, time delay problem will emerge. The relationship between system costs and dynamics is also an important topic because a supply chain network with complex supply relationships might incur high transportation costs.

Conflict of Interests

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

Acknowledgments

This work was supported by the National Natural Science Foundation of China (no. 71401181 and no. 71502076) and the National Natural Science Foundation for Distinguished Scholars of China (no. 71125001). This work was also supported by MOE (Ministry of Education in China) Project of Humanities and Social Sciences (no. 14YJC630136 and no. 15YJC630008).

References

- G. Li, P. Ji, L. Y. Sun, and W. B. Lee, "Modeling and simulation of supply network evolution based on complex adaptive system and fitness landscape," *Computers and Industrial Engineering*, vol. 56, no. 3, pp. 839–853, 2009.
- [2] S. M. Disney and D. R. Towill, "A discrete transfer function model to determine the dynamic stability of a vendor managed inventory supply chain," *International Journal of Production Research*, vol. 40, no. 1, pp. 179–204, 2002.
- [3] C. F. Daganzo, "On the stability of supply chains," *Operations Research*, vol. 52, no. 6, pp. 909–921, 2004.
- [4] Y. C. Wei, H. W. Wang, and C. Qi, "On the stability and bullwhip effect of a production and inventory control system," *International Journal of Production Research*, vol. 51, no. 1, pp. 154–171, 2013.
- [5] Y. Wei, H. Wang, and C. Qi, "The impact of stock-dependent demand on supply chain dynamics," *Applied Mathematical Modelling*, vol. 37, no. 18-19, pp. 8348–8362, 2013.
- [6] H. L. Lee, V. Padmanabhan, and S. Whang, "Information distortion in a supply chain: the bullwhip effect," *Management Science*, vol. 43, no. 4, pp. 546–558, 1997.

- [7] J. Dejonckheere, S. M. Disney, M. R. Lambrecht, and D. R. Towill, "Measuring and avoiding the bullwhip effect: a control theoretic approach," *European Journal of Operational Research*, vol. 147, no. 3, pp. 567–590, 2003.
- [8] J. Dejonckheere, S. M. Disney, M. R. Lambrecht, and D. R. Towill, "The impact of information enrichment on the bullwhip effect in supply chains: a control engineering perspective," *European Journal of Operational Research*, vol. 153, no. 3, pp. 727–750, 2004.
- [9] Y. Ouyang and C. Daganzo, "Characterization of the bullwhip effect in linear, time-invariant supply chains: some formulae and tests," *Management Science*, vol. 52, no. 10, pp. 1544–1556, 2006.
- [10] J. D. Sterman, "Modeling managerial behavior: misperceptions of feedback in a dynamic decision making experiment," *Management Science*, vol. 35, no. 3, pp. 321–339, 1989.
- [11] H. B. Hwarng and N. Xie, "Understanding supply chain dynamics: a chaos perspective," *European Journal of Operational Research*, vol. 184, no. 3, pp. 1163–1178, 2008.
- [12] H. W. Wang, H. X. Liu, and J.-B. Yang, "Dynamic analysis of a two-stage supply chain—a switched system theory approach," *International Journal of Advanced Manufacturing Technology*, vol. 43, no. 1-2, pp. 200–210, 2009.
- [13] S. C. Graves, "A single-item inventory model for a nonstationary demand process," *Manufacturing & Service Operations Management*, vol. 1, no. 1, pp. 50–61, 1999.
- [14] H. Sun and J. Wu, "Scale-free characteristics of supply chain distribution networks," *Modern Physics Letters B*, vol. 19, no. 17, pp. 841–848, 2005.
- [15] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, pp. 47–97, 2002.
- [16] A.-L. Barabási, R. Albert, and H. Jeong, "Mean-field theory for scale-free random networks," *Physica A*, vol. 272, no. 1-2, pp. 173–187, 1999.
- [17] S. H. Strogatz, "Exploring complex networks," *Nature*, vol. 410, no. 6825, pp. 268–276, 2001.
- [18] H. Su, G. Jia, and M. Z. Q. Chen, "Semi-global containment control of multi-agent systems with intermittent input saturation," *Journal of the Franklin Institute*, vol. 352, no. 9, pp. 3504–3525, 2015.
- [19] H. S. Su, M. Z. Q. Chen, and G. R. Chen, "Robust semiglobal coordinated tracking of linear multi-agent systems with input saturation," *International Journal of Robust and Nonlinear Control*, vol. 25, no. 14, pp. 2375–2390, 2015.
- [20] H.-S. Su and M. Z. Q. Chen, "Multi-agent containment control with input saturation on switching topologies," *IET Control Theory & Applications*, vol. 9, no. 3, pp. 399–409, 2015.
- [21] C. S. Lalwani, S. M. Disney, and D. R. Towill, "Controllable, observable and stable state space representations of a generalized order-up-to policy," *International Journal of Production Economics*, vol. 101, no. 1, pp. 172–184, 2006.
- [22] R. Sipahi and I. I. Delice, "Stability of inventory dynamics in supply chains with three delays," *International Journal of Production Economics*, vol. 123, no. 1, pp. 107–117, 2010.
- [23] S. M. Disney, D. R. Towill, and W. Van de Velde, "Variance amplification and the golden ratio in production and inventory control," *International Journal of Production Economics*, vol. 90, no. 3, pp. 295–309, 2004.
- [24] K. Hoberg, J. R. Bradley, and U. W. Thonemann, "Analyzing the effect of the inventory policy on order and inventory variability

with linear control theory," *European Journal of Operational Research*, vol. 176, no. 3, pp. 1620–1642, 2007.

- [25] K. Hoberg, U. W. Thonemann, and J. R. Bradley, "Analyzing the effect of inventory policies on the nonstationary performance with transfer functions," *IIE Transactions*, vol. 39, no. 9, pp. 911– 924, 2007.
- [26] F. Chen, Z. Drezner, J. K. Ryan, and D. Simchi-Levi, "Quantifying the bullwhip effect in a simple supply chain: the impact of forecasting, lead times, and information," *Management Science*, vol. 46, no. 3, pp. 436–443, 2000.
- [27] Y. Ouyang, "The effect of information sharing on supply chain stability and the bullwhip effect," *European Journal of Operational Research*, vol. 182, no. 3, pp. 1107–1121, 2007.
- [28] E. K. Boukas, "Manufacturing systems: LMI approach," *IEEE Transactions on Automatic Control*, vol. 51, no. 6, pp. 1014–1018, 2006.
- [29] T. Nagatani and D. Helbing, "Stability analysis and stabilization strategies for linear supply chains," *Physica A: Statistical Mechanics and its Applications*, vol. 335, no. 3-4, pp. 644–660, 2004.
- [30] S. Cannella and E. Ciancimino, "On the bullwhip avoidance phase: supply chain collaboration and order smoothing," *International Journal of Production Research*, vol. 48, no. 22, pp. 6739–6776, 2010.
- [31] D. Helbing, "Modelling supply networks and business cycles as unstable transport phenomena," *New Journal of Physics*, vol. 5, pp. 90.1–90.28, 2003.
- [32] B. Scholz-Reiter, S. Dashkovskiy, M. Görges, and L. Naujok, "Stability analysis of autonomously controlled production networks," *International Journal of Production Research*, vol. 49, no. 16, pp. 4857–4877, 2011.
- [33] Y. Ouyang and X. Li, "The bullwhip effect in supply chain networks," *European Journal of Operational Research*, vol. 201, no. 3, pp. 799–810, 2010.
- [34] R. Dominguez, S. Cannella, and J. M. Framinan, "On bullwhiplimiting strategies in divergent supply chain networks," *Computers & Industrial Engineering*, vol. 73, no. 1, pp. 85–95, 2014.
- [35] G. P. Cachon, T. Randall, and G. M. Schmidt, "In search of the bullwhip effect," *Manufacturing & Service Operations Management*, vol. 9, no. 4, pp. 457–479, 2007.
- [36] E. Sucky, "The bullwhip effect in supply chains—an overestimated problem?" *International Journal of Production Economics*, vol. 118, no. 1, pp. 311–322, 2009.
- [37] J. H. Kim and H. B. Park, "H_∞ state feedback control for generalized continuous/discrete time-delay system," *Automatica*, vol. 35, no. 8, pp. 1443–1451, 1999.

Research Article

Adaptive Second-Order Synchronization of Two Heterogeneous Nonlinear Coupled Networks

Bo Liu, Shuang Li, and Xuhui Tang

College of Science, North China University of Technology, Beijing 100144, China

Correspondence should be addressed to Bo Liu; boliu@ncut.edu.cn

Received 12 June 2015; Accepted 16 July 2015

Academic Editor: Michael Z. Q. Chen

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

This paper investigates the second-order synchronization of two heterogeneous nonlinear coupled networks by introducing controller and adaptive laws. Based on Lyapunov stability properties and LaSalle invariance principle, it is proved that the position and the velocity of two heterogeneous nonlinear coupled networks are asymptotically stable. Finally, some numerical simulations are presented to verify the analytical results.

1. Introduction

In recent years, people have paid more attention to the synchronization problem of complex networks due to their broad applications, such as biology, physics, communication, computer, and the traffic [1–3].

Various models and algorithms about complex networks have been investigated based on different tasks or interests. To achieve the synchronization of complex networks, the adaptive strategy is one of the most interesting topics on the synchronization problem of complex networks. In [4], the authors introduced an adaptive synchronization scheme in complex networks which was linked through nonlinearly coupling. [5] considered the consensus problems of multiagent systems with second-order nonlinear dynamics by introducing the distributed control gains. In [6], the agents of second-order multiagent systems were governed by both position and velocity consensus protocol with time-varying velocity. In [7], the authors studied the group-consensus problem of second-order nonlinear multiagent systems. There have been many studies about the second-order networks [8-11]. However, due to the limit of outside influences and communication conditions, the dynamics of the coupling nodes can be different; so the heterogeneous networks models were proposed in [12-15]. In [16], the authors investigated the consensus problem of heterogeneous multiagent systems. In [17], the authors discussed the adaptive consensus of second-order multiagent systems with heterogeneous nonlinear dynamics and time-varying delays. In [18], the authors studied the finite-time consensus problem of heterogeneous multiagent systems consisting of first-order and secondorder integrator agents. In [19], the authors investigated the containment control problem of heterogeneous multiagent systems. The recent papers focus on the synchronization of single network [20, 21]. However, the synchronization can also occur in two or more networks [22, 23], such as the inside doors and the outside doors of city subways. In [24], the authors investigated the synchronization between two coupled complex networks. In [25], the authors further solved the synchronization problem of two nonlinear coupled networks. In [26], the number of nodes, dynamics, and topological structures of the two complex networks were different. However, the second-order synchronization of two heterogeneous nonlinear coupled networks has not been investigated.

Motivated by this, in this paper, we focus on the problem of adaptive second-order synchronization of two heterogeneous nonlinear coupled networks. The main contributions of this paper are threefold: (1) the nonlinear intrinsic dynamics of each node is heterogeneous; (2) the synchronization occurs in the two heterogeneous nonlinear coupled networks; (3) controller and adaptive laws are introduced to solve the second-order synchronization of the two heterogeneous nonlinear coupled networks. Particularly, even if the topological structure is unknown, the two heterogeneous nonlinear networks can achieve synchronization by introducing the suitable controller and adaptive laws.

This paper is organized as follows. In Section 2, the second-order models of two heterogeneous nonlinear coupled networks are given. Moreover, some preliminaries are introduced to solve the adaptive synchronization. Section 3 presents the main results and the theoretical analysis of the second-order synchronization of two heterogeneous non-linear coupled networks. Some numerical simulations of the theoretical results are given in Section 4. Finally, the conclusion is made in Section 5.

2. Preliminaries and Problem Statement

Consider the second-order models of two heterogeneous nonlinear coupled networks consisting of N identical nodes described by

$$\begin{aligned} \dot{x}_{i}(t) &= v_{i}(t) \\ \dot{v}_{i}(t) \\ &= f_{i}\left(t, x_{i}(t), v_{i}(t)\right) + \sum_{j \in \mathcal{N}_{i}} \mu_{1} a_{ij}\left(x_{j}(t) - x_{i}(t)\right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{2} a_{ij}\left(v_{j}(t) - v_{i}(t)\right) \\ &+ \xi\left(f_{i}\left(t, q_{i}(t), p_{i}(t)\right) - g_{i}\left(t, x_{i}(t), v_{i}(t)\right)\right) + u_{i}, \\ \dot{q}_{i}(t) &= p_{i}(t) \\ \dot{p}_{i}(t) \end{aligned}$$
(1)

$$= g_{i}(t, q_{i}(t), p_{i}(t)) + \sum_{j \in \mathcal{N}_{i}} \mu_{3} b_{ij}(q_{j}(t) - q_{i}(t))$$

$$+ \sum_{j \in \mathcal{N}_{i}} \mu_{4} b_{ij}(p_{j}(t) - p_{i}(t))$$

$$+ (1 - \xi)(f_{i}(t, q_{i}(t), p_{i}(t)) - g_{i}(t, x_{i}(t), v_{i}(t))),$$
(2)

where $x_i(t) = (x_{i1}(t), x_{i2}(t), \dots, x_{in}(t))^T \in \mathbb{R}^n$, $q_i(t) = (q_{i1}(t), q_{i2}(t), \dots, q_{in}(t))^T \in \mathbb{R}^n$ $(i = 1, 2, \dots, N)$ describe the position vectors of networks (1) and (2), respectively, and $v_i(t) = (v_{i1}(t), v_{i2}(t), \dots, v_{in}(t))^T \in \mathbb{R}^n$, $p_i(t) = (p_{i1}(t), p_{i2}(t), \dots, p_{in}(t))^T \in \mathbb{R}^n$ $(i = 1, 2, \dots, N)$ are their velocity vectors, respectively. $f_i : \mathbb{R}^n \to \mathbb{R}^n$ and $g_i : \mathbb{R}^n \to \mathbb{R}^n$ are continuous functions. $\mu_i > 0$ (i = 1, 2, 3, 4) are the position and velocity coupling strengths in two networks, respectively. $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ and $B = [b_{ij}] \in \mathbb{R}^{n \times n}$ denote the coupling configurations of the two networks, respectively. If there exists communication channel between node *i* and node *j*, then $a_{ij} > 0$, $b_{ij} > 0$ $(i \neq j)$; otherwise, $a_{ij} = 0$, $b_{ij} = 0$ $(i \neq j)$, and the diagonal elements are defined as $a_{ii} = -\sum_{j=1, j\neq i}^{N} a_{ij}, b_{ii} = -\sum_{j=1, j\neq i}^{N} b_{ij}$. N_i is the neighbor set of node *i*. $0 \le \xi \le 1$ describe the nonlinear coupling parameter of both networks. u_i is the controller of network. Define the position error and velocity error of the *i*th node

$$m_{i}(t) = x_{i}(t) - q_{i}(t)$$

$$n_{i}(t) = v_{i}(t) - p_{i}(t).$$
(3)

Differentiating $m_i(t)$ and $n_i(t)$, then

as

$$\begin{split} \dot{m}_{i}(t) &= \dot{x}_{i}(t) - \dot{q}_{i}(t) = v_{i}(t) - p_{i}(t) = n_{i}(t) \\ \dot{n}_{i}(t) \\ &= f_{i}(t, x_{i}(t), v_{i}(t)) + \sum_{j \in \mathcal{N}_{i}} \mu_{1} a_{ij}(x_{j}(t) - x_{i}(t)) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{2} a_{ij}(v_{j}(t) - v_{i}(t)) - g_{i}(t, q_{i}(t), p_{i}(t)) \\ &- \sum_{j \in \mathcal{N}_{i}} \mu_{3} b_{ij}(q_{j}(t) - q_{i}(t)) \\ &- \sum_{j \in \mathcal{N}_{i}} \mu_{4} b_{ij}(p_{j}(t) - p_{i}(t)) \\ &+ (2\xi - 1)(f_{i}(t, q_{i}(t), p_{i}(t)) - g_{i}(t, x_{i}(t), v_{i}(t))) \\ &+ u_{i}. \end{split}$$

$$(4)$$

Denoting $\tilde{a}_{ij} = a_{ij} - \hat{a}_{ij}$ and $\tilde{b}_{ij} = b_{ij} - \hat{b}_{ij}$, we can have

$$\begin{aligned} \dot{\bar{a}}_{ij} &= -\dot{\bar{a}}_{ij} = -(n_i + m_i)^T \\ &\cdot \left(\mu_1 \left(x_j \left(t\right) - x_i \left(t\right)\right) + \mu_2 \left(v_j \left(t\right) - v_i \left(t\right)\right)\right), \\ \dot{\bar{b}}_{ij} &= -\dot{\bar{b}}_{ij} = \left(n_i + m_i\right)^T \\ &\cdot \left(\mu_3 \left(q_j \left(t\right) - q_i \left(t\right)\right) + \mu_4 \left(p_j \left(t\right) - p_i \left(t\right)\right)\right), \\ \dot{E}_i &= -2 \left\|n_i\right\|^2 - 2 \left\|m_i\right\|^2. \end{aligned}$$
(5)

The controller is designed as

$$\begin{split} u_{i}(t) &= -2\xi \left(f_{i}\left(t, q_{i}\left(t \right), p_{i}\left(t \right) \right) - g_{i}\left(t, x_{i}\left(t \right), v_{i}\left(t \right) \right) \right) \\ &+ E_{i}\left(m_{i}\left(t \right) + n_{i}\left(t \right) \right) \\ &- \sum_{j \in \mathcal{N}_{i}} \mu_{1} \widehat{a}_{ij} \left(x_{j}\left(t \right) - x_{i}\left(t \right) \right) \\ &- \sum_{j \in \mathcal{N}_{i}} \mu_{2} \widehat{a}_{ij} \left(v_{j}\left(t \right) - v_{i}\left(t \right) \right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{3} \widehat{b}_{ij} \left(q_{j}\left(t \right) - q_{i}\left(t \right) \right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{4} \widehat{b}_{ij} \left(p_{j}\left(t \right) - p_{i}\left(t \right) \right) . \end{split}$$
(6)

ł

Combining (6), system (4) can be rewritten as

$$\begin{split} \dot{m}_{i}(t) &= n_{i}(t) \\ \dot{n}_{i}(t) &= f_{i}\left(t, x_{i}(t), v_{i}(t)\right) - f_{i}\left(t, q_{i}(t), p_{i}(t)\right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{1} \tilde{a}_{ij}\left(x_{j}(t) - x_{i}(t)\right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{2} \tilde{a}_{ij}\left(v_{j}(t) - v_{i}(t)\right) \\ &+ g_{i}\left(t, x_{i}(t), v_{i}(t)\right) - g_{i}\left(t, q_{i}(t), p_{i}(t)\right) \\ &- \sum_{j \in \mathcal{N}_{i}} \mu_{3} \tilde{b}_{ij}\left(q_{j}(t) - q_{i}(t)\right) \\ &- \sum_{j \in \mathcal{N}_{i}} \mu_{4} \tilde{b}_{ij}\left(p_{j}(t) - p_{i}(t)\right) \\ &+ E_{i}\left(m_{i}(t) + n_{i}(t)\right). \end{split}$$
(7)

In the following, the necessary definition, assumption, and lemma will be presented for discussing the second-order synchronization of two heterogeneous nonlinear coupled networks.

Definition 1. Networks (1) and (2) are said to achieve secondorder synchronization if

$$\lim_{t \to \infty} \|x_i(t) - q_i(t)\| = 0$$

$$\lim_{t \to \infty} \|v_i(t) - p_i(t)\| = 0;$$
(8)

that is,

$$\begin{split} &\lim_{t \to \infty} \left\| m_i(t) \right\| = 0 \\ &\lim_{t \to \infty} \left\| n_i(t) \right\| = 0, \end{split} \tag{9}$$

for i = 1, 2, ..., N.

Assumption 2. For every f_i of network (1) and g_i of network (2) $(\forall x, v, q, p \in \mathbb{R}^n)$, there exist the constants $\rho_1 > 0, \rho_2 > 0$ such that

$$\begin{aligned} \left\| f_{i}\left(t, x_{i}\left(t\right), v_{i}\left(t\right)\right) - f_{i}\left(t, q_{i}\left(t\right), p_{i}\left(t\right)\right) \right\| \\ &\leq \rho_{1}\left(\left\| x_{i}\left(t\right) - q_{i}\left(t\right) \right\| + \left\| v_{i}\left(t\right) - p_{i}\left(t\right) \right\| \right), \\ \left\| g_{i}\left(t, x_{i}\left(t\right), v_{i}\left(t\right)\right) - g_{i}\left(t, q_{i}\left(t\right), p_{i}\left(t\right)\right) \right\| \\ &\leq \rho_{2}\left(\left\| x_{i}\left(t\right) - q_{i}\left(t\right) \right\| + \left\| v_{i}\left(t\right) - p_{i}\left(t\right) \right\| \right). \end{aligned}$$
(10)

Lemma 3 (see [17]). For any vectors $x, y \in \mathbb{R}^n$ and positive definite matrix $G \in \mathbb{R}^{n \times n}$, the following matrix inequality holds:

$$2x^{T}y \le x^{T}Gx + y^{T}G^{-1}y.$$
 (11)

3. Main Results

In this section, we will investigate the second-order synchronization of two heterogeneous nonlinear coupled networks and provide the detailed analysis.

Theorem 4. Consider networks (1) and (2) steered by (6) under Assumption 2, then the position and velocity of each node can asymptotically synchronize.

Proof. Constructing the Lyapunov function,

$$V(t) = V_1(t) + V_2(t) + V_3(t), \qquad (12)$$

where

$$V_{1}(t) = \frac{1}{2} \sum_{i=1}^{N} (m_{i}(t) + n_{i}(t))^{T} (m_{i}(t) + n_{i}(t)),$$

$$V_{2}(t) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_{i}} \tilde{b}_{ij}^{2} + \frac{1}{2} \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_{i}} \tilde{a}_{ij}^{2},$$

$$V_{3}(t) = \frac{1}{2} \sum_{i=1}^{N} (E_{i} + k_{i})^{2},$$
(13)

and where k_i is a positive constant. Differentiating $V_1(t)$, then

$$\begin{split} \dot{V}_{1}(t) &= \sum_{i=1}^{N} m_{i}^{T} n_{i} + \sum_{i=1}^{N} m_{i}^{T} \dot{n}_{i} + \sum_{i=1}^{N} n_{i}^{T} n_{i} + \sum_{i=1}^{N} n_{i}^{T} \dot{n}_{i} \leq \frac{1}{2} \\ &\cdot \sum_{i=1}^{N} m_{i}^{T} m_{i} + \frac{3}{2} \sum_{i=1}^{N} n_{i}^{T} n_{i} + \sum_{i=1}^{N} (m_{i} + n_{i})^{T} \dot{n}_{i} \leq \frac{1}{2} \\ &\cdot \sum_{i=1}^{N} m_{i}^{T} m_{i} + \frac{3}{2} \sum_{i=1}^{N} n_{i}^{T} n_{i} + \sum_{i=1}^{N} (m_{i} + n_{i})^{T} \left[\rho_{1} (m_{i} + n_{i}) \right] \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{1} \tilde{a}_{ij} \left(x_{j} (t) - x_{i} (t) \right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{2} \tilde{a}_{ij} \left(v_{j} (t) - v_{i} (t) \right) + \rho_{2} (m_{i} + n_{i}) \\ &- \sum_{j \in \mathcal{N}_{i}} \mu_{3} \tilde{b}_{ij} \left(q_{j} (t) - q_{i} (t) \right) \\ &- \sum_{j \in \mathcal{N}_{i}} \mu_{4} \tilde{b}_{ij} \left(p_{j} (t) - p_{i} (t) \right) + E_{i} (m_{i} + n_{i}) \right] = \frac{1}{2} \\ &\cdot \sum_{i=1}^{N} m_{i}^{T} m_{i} + \frac{3}{2} \sum_{i=1}^{N} n_{i}^{T} n_{i} + \sum_{i=1}^{N} (m_{i} + n_{i})^{T} \left(\rho_{1} + \rho_{2} \right) (m_{i} \\ &+ n_{i}) + \sum_{i=1}^{N} (m_{i} + n_{i})^{T} \left[\sum_{j \in \mathcal{N}_{i}} \mu_{1} \tilde{a}_{ij} \left(x_{j} (t) - x_{i} (t) \right) \right] \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{2} \tilde{a}_{ij} \left(v_{j} (t) - v_{i} (t) \right) \right] + \sum_{i=1}^{N} (m_{i} + n_{i})^{T} \\ &\cdot \left[- \sum_{j \in \mathcal{N}_{i}} \mu_{3} \tilde{b}_{ij} \left(q_{j} (t) - q_{i} (t) \right) \right] \end{split}$$

$$\begin{split} &-\sum_{j \in \mathcal{N}_{i}} \mu_{4} \widetilde{b}_{ij} \left(p_{j} \left(t \right) - p_{i} \left(t \right) \right) \right] + \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} E_{i} \left(m_{i} \right) \\ &+ n_{i} \right) = \frac{1}{2} \sum_{i=1}^{N} m_{i}^{T} m_{i} + \frac{3}{2} \sum_{i=1}^{N} n_{i}^{T} n_{i} + \sum_{i=1}^{N} m_{i}^{T} \rho m_{i} \\ &+ \sum_{i=1}^{N} m_{i}^{T} \rho n_{i} + \sum_{i=1}^{N} n_{i}^{T} \rho m_{i} + \sum_{i=1}^{N} n_{i}^{T} \rho n_{i} + \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \\ &\cdot \left[\sum_{j \in \mathcal{N}_{i}} \mu_{i} \widetilde{a}_{ij} \left(x_{j} \left(t \right) - x_{i} \left(t \right) \right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{2} \widetilde{a}_{ij} \left(p_{j} \left(t \right) - p_{i} \left(t \right) \right) \right] + \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} E_{i} \left(m_{i} \\ &+ n_{i} \right) \leq \frac{1}{2} \sum_{i=1}^{N} m_{i}^{T} m_{i} + \frac{3}{2} \sum_{i=1}^{N} n_{i}^{T} n_{i} + \rho \sum_{i=1}^{N} m_{i}^{T} m_{i} \\ &+ \rho \sum_{i=1}^{N} n_{i}^{T} n_{i} + \rho \sum_{i=1}^{N} m_{i}^{T} m_{i} + \rho \sum_{i=1}^{N} n_{i}^{T} n_{i} + \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \\ &\cdot \left[\sum_{j \in \mathcal{N}_{i}} \mu_{1} \widetilde{a}_{ij} \left(x_{j} \left(t \right) - x_{i} \left(t \right) \right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{1} \widetilde{a}_{ij} \left(x_{j} \left(t \right) - x_{i} \left(t \right) \right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{1} \widetilde{a}_{ij} \left(p_{j} \left(t \right) - p_{i} \left(t \right) \right) \\ &+ \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} E_{i} \left(m_{i} \\ &+ n_{i} \right) = \left(\frac{1}{2} + 2\rho \right) \sum_{i=1}^{N} m_{i}^{T} m_{i} + \left(\frac{3}{2} + 2\rho \right) \sum_{i=1}^{N} n_{i}^{T} n_{i} \\ &+ \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \left[\sum_{j \in \mathcal{N}_{i}} \mu_{1} \widetilde{a}_{ij} \left(x_{j} \left(t \right) - x_{i} \left(t \right) \right) \\ &+ \sum_{i=1}^{N} \mu_{2} \widetilde{a}_{ij} \left(v_{j} \left(t \right) - v_{i} \left(t \right) \right) \\ &+ \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \left[\sum_{j \in \mathcal{N}_{i}} \mu_{1} \widetilde{a}_{ij} \left(x_{j} \left(t \right) - x_{i} \left(t \right) \right) \\ &+ \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \left[\sum_{j \in \mathcal{N}_{i}} \mu_{1} \widetilde{a}_{ij} \left(x_{j} \left(t \right) - x_{i} \left(t \right) \right) \\ &+ \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \left[\sum_{j \in \mathcal{N}_{i}} \mu_{1} \widetilde{a}_{ij} \left(x_{j} \left(t \right) - x_{i} \left(t \right) \right) \\ &+ \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \left[\sum_{i=1}^{N} \mu_{i} \widetilde{a}_{ij} \left(x_{j} \left(t \right) - x_{i} \left(t \right) \right) \\ &+ \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \left[\sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \right] \\ &+ \sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \left[\sum_{i=1}^{N} \left(m_{i} + n_{i} \right)^{T} \right]$$

Mathematical Problems in Engineering

$$-\sum_{j\in\mathcal{N}_{i}}\mu_{4}\widetilde{b}_{ij}\left(p_{j}\left(t\right)-p_{i}\left(t\right)\right)\right]+\sum_{i=1}^{N}\left(m_{i}+n_{i}\right)^{T}E_{i}\left(m_{i}+n_{i}\right),$$

$$+n_{i},$$
(14)

where $\rho = \rho_1 + \rho_2$. Differentiating $V_2(t)$, we get

$$\begin{split} \dot{V}_{2}(t) &= \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_{i}} \tilde{b}_{ij} \dot{\tilde{b}}_{ij} + \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_{i}} \tilde{a}_{ij} \dot{\tilde{a}}_{ij} \\ &= \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_{i}} \tilde{b}_{ij} \left(n_{i} + m_{i} \right)^{T} \\ &\cdot \left[\mu_{3} \left(q_{j}(t) - q_{i}(t) \right) + \mu_{4} \left(p_{j}(t) - p_{i}(t) \right) \right] \\ &+ \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_{i}} \tilde{a}_{ij} \left(n_{i} + m_{i} \right)^{T} \\ &\cdot \left[-\mu_{1} \left(x_{j}(t) - x_{i}(t) \right) - \mu_{2} \left(v_{j}(t) - v_{i}(t) \right) \right]. \end{split}$$
(15)

Differentiating $V_3(t)$, then

$$\dot{V}_{3}(t) = \sum_{i=1}^{N} (E_{i} + k_{i}) \dot{E}_{i} = \sum_{i=1}^{N} E_{i} \dot{E}_{i} + \sum_{i=1}^{N} k_{i} \dot{E}_{i}$$
$$= \sum_{i=1}^{N} E_{i} \left(-2 \|n_{i}\|^{2} - 2 \|m_{i}\|^{2}\right)$$
(16)
$$+ \sum_{i=1}^{N} k_{i} \left(-2 \|n_{i}\|^{2} - 2 \|m_{i}\|^{2}\right).$$

Combining $\dot{V}_1(t), \dot{V}_2(t),$ and $\dot{V}_3(t),$ then we can have

$$\begin{split} \dot{V}(t) &\leq \left(\frac{1}{2} + 2\rho\right) \sum_{i=1}^{N} m_i^T m_i + \left(\frac{3}{2} + 2\rho\right) \sum_{i=1}^{N} n_i^T n_i \\ &+ \sum_{i=1}^{N} \left(m_i + n_i\right)^T E_i \left(m_i + n_i\right) \\ &+ \sum_{i=1}^{N} E_i \left(-2 \|n_i\|^2 - 2 \|m_i\|^2\right) \\ &+ \sum_{i=1}^{N} k_i \left(-2 \|n_i\|^2 - 2 \|m_i\|^2\right) \\ &= \left(\frac{1}{2} + 2\rho\right) \sum_{i=1}^{N} m_i^T m_i + \left(\frac{3}{2} + 2\rho\right) \sum_{i=1}^{N} n_i^T n_i \\ &+ \sum_{i=1}^{N} m_i^T E_i n_i + \sum_{i=1}^{N} n_i^T E_i n_i + \sum_{i=1}^{N} m_i^T E_i m_i \\ &+ \sum_{i=1}^{N} n_i^T E_i m_i + \sum_{i=1}^{N} E_i \left(-2 \|n_i\|^2 - 2 \|m_i\|^2\right) \end{split}$$

$$\begin{aligned} &+ \sum_{i=1}^{N} k_{i} \left(-2 \|n_{i}\|^{2} - 2 \|m_{i}\|^{2}\right) \\ &\leq \left(\frac{1}{2} + 2\rho\right) \sum_{i=1}^{N} m_{i}^{T} m_{i} + \left(\frac{3}{2} + 2\rho\right) \sum_{i=1}^{N} n_{i}^{T} n_{i} \\ &+ 2 \sum_{i=1}^{N} m_{i}^{T} E_{i} m_{i} + 2 \sum_{i=1}^{N} n_{i}^{T} E_{i} n_{i} \\ &+ \sum_{i=1}^{N} E_{i} \left(-2 \|n_{i}\|^{2} - 2 \|m_{i}\|^{2}\right) \\ &+ \sum_{i=1}^{N} k_{i} \left(-2 \|n_{i}\|^{2} - 2 \|m_{i}\|^{2}\right) \\ &= \left(\frac{1}{2} + 2\rho\right) \sum_{i=1}^{N} m_{i}^{T} m_{i} + \left(\frac{3}{2} + 2\rho\right) \sum_{i=1}^{N} n_{i}^{T} n_{i} \\ &- 2k_{i} \sum_{i=1}^{N} \|m_{i}\|^{2} - 2k_{i} \sum_{i=1}^{N} \|n_{i}\|^{2} \leq 0, \end{aligned}$$

if $k_i \ge 3/4 + \rho$. Based on LaSalle invariance principle, we can know that, for any initial states, the error solution will tend to zero, which implies that networks (1) and (2) can asymptotically synchronize with controller (6).

(17)

Remark 5. When the topology structure is unknown, the two heterogeneous nonlinear coupled networks also can be asymptotically synchronized by controller (6).

Corollary 6. If the coupled networks (1) and (2) have the identical dynamics, the networks can asymptotically synchronize through the following controller:

$$\begin{split} u_{i}(t) &= -\sum_{j \in \mathcal{N}_{i}} \mu_{1} \widehat{a}_{ij} \left(x_{j}(t) - x_{i}(t) \right) \\ &- \sum_{j \in \mathcal{N}_{i}} \mu_{2} \widehat{a}_{ij} \left(v_{j}(t) - v_{i}(t) \right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{3} \widehat{b}_{ij} \left(q_{j}(t) - q_{i}(t) \right) \\ &+ \sum_{j \in \mathcal{N}_{i}} \mu_{4} \widehat{b}_{ij} \left(p_{j}(t) - p_{i}(t) \right) \\ &+ E_{i} \left(m_{i}(t) + n_{i}(t) \right), \end{split}$$
(18)

where $\hat{a}_{ii}, \hat{b}_{ii}, E_i$ are the same as Theorem 4.

4. Simulations

In this section, several numerical simulations are given to illustrate the analytical results.



FIGURE 1: The error trajectories for velocity with controllers.

We choose Lorentz system of two different parameters as nonlinear dynamics of networks (1) and (2). The Lorentz system is described as the following:

$$\dot{x}(t) = v(t)$$

$$\dot{v}(t) = \begin{cases} \alpha \left(v_y - v_x \right) \\ \beta v_x - v_x v_z - v_y \\ v_x v_y - \gamma v_y, \end{cases}$$
(19)

where α , β , and γ are the parameters. For network (1), let $\alpha = 10$, $\beta = 28$, and $\gamma = 8/3$; and for network (2), let $\alpha = 16$, $\beta = 4$, and $\gamma = 45.92$.

The coupling matrixes of networks (1) and (2) with four nodes, respectively, are described by the following matrixes:

$$A = \begin{bmatrix} -2 & 1 & 0 & 1 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix},$$

$$B = \begin{bmatrix} -2 & 1 & 0 & 1 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 1 & 0 & 1 & -2 \end{bmatrix}.$$
(20)

With controller (6), it can be found that the velocity and position of networks (1) and (2) can synchronize to the synchronous state, described as Figures 1-2, respectively. However, if networks (1) and (2) without the controller are in the same conditions, we can find that velocity and position of the networks cannot achieve synchronization, when $\xi = 0.1$. The simulations are shown in Figures 3 and 4.



FIGURE 2: The error trajectories for position with controllers.



FIGURE 3: The error trajectories for velocity without controllers.

5. Conclusion

In this paper, we have considered the adaptive second-order synchronization of two heterogeneous nonlinear coupled networks. By constructing a valid Lyapunov function, we have proved that the networks can achieve asymptotically synchronization with the given controller and adaptive laws. Particularly, even if the topological structure is unknown, the networks also can be synchronized by the given controller and adaptive laws.



FIGURE 4: The error trajectories for position without controllers.

Conflict of Interests

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

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China under Grant (no. 61304049), "The-Great-Wall-Scholar" Candidate Training-Plan of North China University of Technology, Construction Plan for Innovative Research Team of North China University of Technology, and the Plan training project of excellent young teacher of North China University of Technology, the special project of North China University of Technology (no. XN085).

References

- H. Su, G. Jia, and M. Z. Q. Chen, "Semi-global containment control of multi-agent systems with input saturation," *IET Control Theory & Applications*, vol. 8, no. 18, pp. 2229–2237, 2014.
- [2] H. Su and M. Z. Chen, "Multi-agent containment control with input saturation on switching topologies," *IET Control Theory & Applications*, vol. 9, no. 3, pp. 399–409, 2015.
- [3] B. Liu, W.-N. Hu, J. Zhang, and H.-S. Su, "Controllability of discrete-time multi-agent systems with multiple leaders on fixed networks," *Communications in Theoretical Physics*, vol. 58, no. 6, pp. 856–862, 2012.
- [4] Y. Liang and X. Wang, "Adaptive synchronization in complex networks through nonlinearly coupling," *Computer Engineering and Applications*, vol. 48, no. 10, pp. 25–28, 2012.
- [5] W. Yu, W. Ren, W. X. Zheng, G. Chen, and J. Lü, "Distributed control gains design for consensus in multi-agent systems with second-order nonlinear dynamics," *Automatica*, vol. 49, no. 7, pp. 2107–2115, 2013.
- [6] W. Yu, G. Chen, M. Cao, and J. Kurths, "Second-Order consensus for multiagent systems with directed topologies and

nonlinear dynamics," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 40, no. 3, pp. 881–891, 2010.

- [7] Q. Ma, Z. Wang, and G. Miao, "Second-order group consensus for multi-agent systems via pinning leader-following approach," *Journal of the Franklin Institute*, vol. 351, no. 3, pp. 1288–1300, 2014.
- [8] B. Liu, S. Li, and L. Wang, "Adaptive synchronization of two time-varying delay nonlinear coupled networks," in *Proceedings* of the 33rd Chinese Control Conference (CCC '14), pp. 3800– 3804, July 2014.
- [9] P. Lin and Y. Jia, "Consensus of second-order discrete-time multi-agent systems with nonuniform time-delays and dynamically changing topologies," *Automatica*, vol. 45, no. 9, pp. 2154– 2158, 2009.
- [10] Y. Zheng, Y. Zhu, and L. Wang, "Finite-time consensus of multiple second-order dynamic agents without velocity measurements," *International Journal of Systems Science*, vol. 45, no. 3, pp. 579–588, 2014.
- [11] Z.-J. Tang, T.-Z. Huang, J.-L. Shao, and J.-P. Hu, "Consensus of second-order multi-agent systems with nonuniform timevarying delays," *Neurocomputing*, vol. 97, pp. 410–414, 2012.
- [12] H. Su, Z. Rong, M. Z. Q. Chen, X. Wang, G. Chen, and H. Wang, "Decentralized adaptive pinning control for cluster synchronization of complex dynamical networks," *IEEE Transactions on Cybernetics*, vol. 43, no. 1, pp. 394–399, 2013.
- [13] B. Liu, H. Su, R. Li, D. Sun, and W. Hu, "Switching controllability of discrete-time multi-agent systems with multiple leaders and time-delays," *Applied Mathematics and Computation*, vol. 228, pp. 571–588, 2014.
- [14] H. Su, M. Z. Chen, J. Lam, and Z. Lin, "Semi-global leaderfollowing consensus of linear multi-agent systems with input saturation via low gain feedback," *IEEE Transactions on Circuits* and Systems. I. Regular Papers, vol. 60, no. 7, pp. 1881–1889, 2013.
- [15] H. Su, N. Zhang, M. Z. Chen, H. Wang, and X. Wang, "Adaptive flocking with a virtual leader of multiple agents governed by locally Lipschitz nonlinearity," *Nonlinear Analysis: Real World Applications*, vol. 14, no. 1, pp. 798–806, 2013.
- [16] Y. Zheng, Y. Zhu, and L. Wang, "Consensus of heterogeneous multi-agent systems," *IET Control Theory & Applications*, vol. 5, no. 16, pp. 1881–1888, 2011.
- [17] B. Liu, X. Wang, H. Su, Y. Gao, and L. Wang, "Adaptive secondorder consensus of multi-agent systems with heterogeneous nonlinear dynamics and time-varying delays," *Neurocomputing*, vol. 118, pp. 289–300, 2013.
- [18] Y. Zheng and L. Wang, "Finite-time consensus of heterogeneous multi-agent systems with and without velocity measurements," *Systems & Control Letters*, vol. 61, no. 8, pp. 871–878, 2012.
- [19] Y. Zheng and L. Wang, "Containment control of heterogeneous multi-agent systems," *International Journal of Control*, vol. 87, no. 1, pp. 1–8, 2014.
- [20] H. Su, M. Z. Q. Chen, X. Wang, and J. Lam, "Semiglobal observer-based leader-following consensus with input saturation," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 6, pp. 2842–2850, 2014.
- [21] B. Liu, X. Wang, H. Su, H. Zhou, Y. Shi, and R. Li, "Adaptive synchronization of complex dynamical networks with timevarying delays," *Circuits, Systems, and Signal Processing*, vol. 33, no. 4, pp. 1173–1188, 2014.
- [22] J.-A. Wang, "Adaptive generalized synchronization between two different complex networks with time-varying delay coupling," *Acta Physica Sinica*, vol. 61, no. 2, Article ID 020509, 2012.

- [23] H. Tang, L. Chen, J.-A. Lu, and C. K. Tse, "Adaptive synchronization between two complex networks with nonidentical topological structures," *Physica A: Statistical Mechanics and its Applications*, vol. 387, no. 22, pp. 5623–5630, 2008.
- [24] C. Li, W. Sun, and J. Kurths, "Synchronization between two coupled complex networks," *Physical Review E*, vol. 76, no. 4, Article ID 046204, 2007.
- [25] Y. Du, W. Sun, and C. Li, "Adaptive synchronization between two nonlinear complex networks," *Communications in Applied Mathematics and Computational Science*, vol. 23, no. 2, pp. 87– 94, 2009.
- [26] J. Chen, L. Jiao, J. Wu, and X. Wang, "Adaptive synchronization between two different complex networks with time-varying delay coupling," *Chinese Physics Letters*, vol. 26, no. 6, Article ID 060505, 2009.
Research Article Algorithm for Identifying Minimum Driver Nodes Based on Structural Controllability

Reza Haghighi¹ and HamidReza Namazi²

¹Department of Electrical and Computer Engineering, National University of Singapore, Singapore 119077 ²Department of Mechanical Engineering, Faculty of Engineering, Universiti Malaysia Sarawak, 94300 Kuching, Malaysia

Correspondence should be addressed to HamidReza Namazi; m080012@e.ntu.edu.sg

Received 15 May 2015; Accepted 30 July 2015

Academic Editor: Hou-Sheng Su

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

Existing methods on structural controllability of networked systems are based on critical assumptions such as nodal dynamics with infinite time constants and availability of input signals to all nodes. In this paper, we relax these assumptions and examine the structural controllability for practical model of networked systems. We explore the relationship between structural controllability and graph reachability. Consequently, a simple graph-based algorithm is presented to obtain the minimum driver nodes. Finally, simulation results are presented to illustrate the performance of the proposed algorithm in dealing with large-scale networked systems.

1. Introduction

Advances in communications technology have opened up new challenges in the area of networked systems. Controllability of multiagent networked systems as a fundamental concept in this field has received considerable attention. The pioneer work in analysing controllability of multiagent systems with leader-follower architecture had been carried out by Tanner [1], where controllability conditions were provided for multiagent systems with undirected graph topology based on eigenvectors of the Laplacian matrix. In further development, some algebraic conditions for controllability of multiagent systems are presented in [2, 3]. Ji and Egerstedt [4] introduced network equitable partitions to present a necessary condition for the controllability of leaderfollower multiagent systems. Inspired by [4], Rahmani et al. [5] proposed the controllability of multiagent systems with multiple leaders. Liu et al. [6] derived a simple controllability condition for discrete-time single-leader switching networks, which was further extended to continuous-time single-leader switching networks [7]. Ji et al. [8] derived a necessary and sufficient condition for the controllability of leaderfollower multiagent systems, by dividing the overall system into several connected components. The other related topics in this area are leader-follower consensus [9, 10], leader-follower formation control [11–13], containment control [14, 15], and pinning-controllability of networked systems [16, 17].

The concept of structural controllability has been studied extensively since the classical work by Lin [18]. In [18], structural controllability of SISO linear systems was explored by introducing a notion of structured matrix whose elements are either fixed zeros or independent free parameters. Shields and Pearson [19] extended the results of [18] to structural controllability of multiinput linear systems. Since then various works have been carried out on the structural controllability of linear systems [20-22]. Recently, structural controllability of networked systems has emerged as a major interest in the network sciences. A notable work in this area is carried out by Liu et al. [23] which addressed the structural controllability of complex networks. Jafari et al. [24] studied structural controllability of a leader-follower multiagent system with multiple leaders. Sundaram and Hadjicostis [25] developed a graph-theoretic characterization of controllability and observability of linear systems over finite fields. Haghighi and Cheah [26] employed the concept of structural observability to examine the weight-balanceability of networked systems.

For large-scale networked systems, it is infeasible to apply input signals to all network nodes due to the high control cost and the difficulty of practical implementations. In this case, a fundamental problem is to identify a certain amount of nodes to be driven externally to bring the whole network under control. This problem was addressed in [23], where a theoretical framework was developed to solve the minimum input problem based on Lin's structural controllability theorem [18]. As pointed out in [27], the results in [23] are based on the assumption that each node has an infinite time constant, which do not generally represent the dynamics of the physical and biological systems.

Despite the model in [23], Cowan et al. [27] considered internal dynamics for all nodes of the network. Cowan's result states that structural controllability does not depend on degree distribution. Hence, the structural controllability can always be conferred with a single independent control input. However, the result in [27] suffers from a drawback that each independent input is connected to all nodes in the network, which is practically infeasible.

In this paper, we examined the structural controllability in networked systems by relaxing the critical assumptions in previous results. Consequently, we provide a graph-theoretic method to identify driver nodes. We present an algorithm to determine minimum driver nodes in networked systems. The contribution of this paper is twofold: (i) we relaxed the assumptions in existing methods, such as infinite time constant for each node, and having direct access to input signals by all nodes, on structural controllability of networked systems. (ii) We provide a simple algorithm to obtain minimum driver nodes in networked systems.

The paper is organized as follows. Section 2 presents some preliminaries in graph theory and controllability. Section 3 presents the model of the networked systems. Section 4 addresses the structural controllability in networked systems and presents the linkage between the structural controllability and the graph reachability in networked systems. An algorithm for identifying minimum driver nodes in networked systems is proposed in Section 5. Section 6 presents the simulation results and Section 7 concludes this paper.

2. Preliminaries

The communication between nodes can be expressed by a weighted directed graph $\mathscr{C}(\mathscr{V}, \mathscr{E}, \mathscr{A})$, such that $\mathscr{V} = \{v_1, v_2, \ldots, v_N\}$ represents the set of nodes, $\mathscr{E} \subseteq \mathscr{V} \times \mathscr{V}$ is the edge set, and $\mathscr{A} = [a_{ij}]$ is the weighted adjacency matrix where $a_{ij} > 0$ if $(i, j) \in \mathscr{E}$ and $a_{ij} = 0$ otherwise. A graph \mathscr{H} is said to be a subgraph of a graph \mathscr{E} if $\mathscr{V}(\mathscr{H}) \subseteq \mathscr{V}(\mathscr{E})$ and $\mathscr{E}(\mathscr{H}) \subseteq \mathscr{E}(\mathscr{E})$.

A directed path in a digraph is an ordered sequence of nodes so that any two consecutive nodes in the sequence are an edge of the digraph. An undirected graph is a tree if and only if, for any two nodes, there is a unique path connecting them. A directed spanning tree or arborescence is a digraph such that there is a unique directed path from a designated root node to every other node. *Definition 1.* A digraph \mathscr{G} is called an arborescence diverging from node u, if there is only one directed path between root u and any other node of \mathscr{G} . If \mathscr{G} is an arborescence diverging from u, then its reverse digraph (i.e., all edges of \mathscr{G} are reversed) is called an arborescence converging to u [28].

If there is an arborescence subdigraph diverging from an arbitrary node *u*, then *u* is called a globally reachable node.

Definition 2. Driver nodes are nodes in a network that have to be controlled in order to completely control the entire network.

Definition 3. A matrix A_s is said to be a structured matrix if its elements are either fixed zeros or independent free parameters [29].

Definition 4. Two dynamical systems are called structurally equivalent, if their interconnection structures are identical. Hence, we can say (A, B) has the same structure as (A1, B1), if for every fixed zero entry of the matrix $\begin{bmatrix} A & B \end{bmatrix}$, the corresponding entry of the matrix $\begin{bmatrix} A1 & B1 \end{bmatrix}$ is fixed zero and vice versa.

Definition 5. The structural rank (srank) of a matrix is the maximum rank of all structurally equivalent matrices [30].

Theorem 6 (controllability test [31]). $(A_{n \times n}, B_{n \times m})$ is controllable if and only if there is no left eigenvector of A that is orthogonal to B; that is,

$$\forall w \quad where \ w^T A = \lambda w^T \Longrightarrow w^T B \neq 0. \tag{1}$$

Theorem 7 (Popov-Belevitch-Hautus controllability test [32]). $(A_{n \times n}, B_{n \times m})$ is controllable if and only if

$$\operatorname{Rank}\left[\lambda I - A \quad B\right] = n,\tag{2}$$

where λ is an eigenvalue of A.

3. Model of Interconnected Networks

We consider each node in the network corresponding to a dynamical system, governed by the following equation:

$$\dot{x}_{i} = -a_{ii}x_{i} + \sum_{\substack{j=1\\j\neq i}}^{N} a_{ij}x_{j} + b_{i}u_{i},$$
(3)

where $x_i \in R$ denotes the state of node *i*, *N* is the total number of nodes, and u_i denotes an external input.

Interconnected system (3) can be represented in matrix form

$$\dot{x} = Ax + Bu,\tag{4}$$

where $x = [x_1, x_2, ..., x_N]^T$, $u = [u_1, u_2, ..., u_m]^T$, $B \in \mathbb{R}^{N \times m}$ is input matrix, and $A \in \mathbb{R}^{n \times n}$ is defined as $A = \mathscr{A} - \mathscr{D}$, where \mathscr{A} is the adjacency matrix and $\mathscr{D} = \text{diag}\{a_{11}, a_{22}, ..., a_{NN}\}$.

4. Controllability and Graph Reachability

According to classical control theory, a dynamical system is controllable if for any initial state there exists an input that can drive the system to any final state in a finite time. It is well known that the system (A, B) is controllable if and only if the following controllability matrix:

$$\mathscr{C} = \left[B, AB, \dots, A^{N-1}B \right]$$
(5)

has full rank. Even though a system with a pair of (A, B) might be uncontrollable, it can be controllable for another structurally equivalent pair (A^*, B^*) [18].

Definition 8. A dynamical system (A, B) is structurally controllable if there exists a structurally equivalent system (A^*, B^*) that is controllable [33].

In what follows, we first consider network with single driver node and present the relation between graph reachability and controllability.

4.1. Controllability of Networks with Single Driver Node. Consider a network of nodes with single driver node, *i*, which is expressed as follows:

$$\dot{x} = Ax + Bu,\tag{6}$$

where $B = [\underbrace{0, \dots, 0, b_i, 0, \dots, 0}_{i \text{th element}}]^T$. To examine the controlla-

bility, we form the following zero-state response:

$$x(t) = \int_0^t e^{A(t-\tau)} Bu(\tau) d\tau.$$
⁽⁷⁾

The term $e^{A(t-\tau)}B$ refers to the *i*th column of the matrix $e^{A(t-\tau)}$ multiplied by b_i . Using Cayley-Hamilton theorem, $e^{A(t-\tau)}$ can be expanded as follows:

$$e^{A(t-\tau)} = \sum_{i=0}^{N-1} \alpha_i (t-\tau) A^i,$$
(8)

where $\alpha_i(\cdot)$ are scalar functions. We state the following theorem.

Theorem 9. Consider the network expressed by (6). The network is structurally controllable if and only if there is an arborescence subdigraph diverging from driver node i.

Proof (necessity condition). According to Lemma A.1 in the Appendix, the (j, i)th element of matrix series (8) is zero, if there is no path from node *i* to node *j*. In this case, the *j*th element of the $e^{A(t-\tau)}B$ is zero and remains zero for all values of the network link weights; therefore at least $x^* = [\underbrace{0, \ldots, 0, \alpha, 0, \ldots, 0}]^T$ is an uncontrollable state of the system. *j*th element

Sufficiency Condition. We show that a network, which contains an arborescence subdigraph diverging from its driver node, is structurally controllable. Without loss of generality, we assume that node 1 is the driver node. In structural controllability, independent nonzero parameters can take any values including zero. Hence, we zero out the weights for redundant links in such a way that the digraph associated with the network becomes an arborescence diverging from the driver node.

Matrices *A* and *B* in (4) for an arborescence diverging from driver node can be expressed as follows:

$$A = \begin{bmatrix} -a_{11} & 0 & \cdots & 0 \\ a_{21} & -a_{22} & \vdots \\ \vdots & \vdots & \ddots & 0 \\ a_{N1} & a_{N2} & \cdots & -a_{NN} \end{bmatrix},$$
(9)
$$B = \begin{bmatrix} b_1 & 0 & \cdots & 0 \end{bmatrix}^T.$$

Let $w_i = [w_1^i, \dots, w_N^i]^T$ be the left eigenvector associated with eigenvalue λ_i . Using Theorem 6, we have

$$w_i^T B \neq 0 \Longleftrightarrow w_1^i b_1 \neq 0 \Longleftrightarrow w_1^i \neq 0.$$
 (10)

Therefore, to show that the network is structurally controllable, we need to prove the existence of weights such that $w_1^i \neq 0$, for i = 1, ..., N. To do so, let a_{ii} be a strictly monotonic sequence for i = 1, ..., N. Since A is triangular matrix with distinct diagonal entries, eigenvalues of A are its diagonal entries; that is, $\lambda_i = -a_{ii}$ for i = 1, ..., N. Therefore, we obtain w_1^i by solving $w_i^T A = \lambda_i w_i$ as follows:

$$\begin{split} \lambda_{1} &= -a_{11} \\ \implies w_{1}^{1} = 1, \\ \lambda_{2} &= -a_{22} \\ \implies w_{1}^{2} = \frac{a_{21}}{a_{11} - a_{22}}, \\ \lambda_{3} &= -a_{33} \\ \implies w_{1}^{3} = \frac{a_{31}}{a_{11} - a_{33}} + \frac{a_{32}a_{21}}{(a_{22} - a_{33})(a_{11} - a_{33})} \\ \vdots \end{split}$$

$$(11)$$

 $\lambda_N = -a_{NN}$

+

$$\implies w_1^N = \frac{a_{N1}}{a_{11} - a_{NN}} + \sum_{l_1=2}^{N-1} \frac{a_{Nl_1}a_{l_11}}{(a_{l_1l_1} - a_{NN})(a_{11} - a_{NN})}$$
$$+ \sum_{l_1=3}^{N-1} \sum_{l_2=2}^{l_1-1} \frac{a_{Nl_1}a_{l_1l_2}a_{l_21}}{(a_{l_1l_1} - a_{NN})(a_{l_2l_2} - a_{NN})(a_{11} - a_{NN})}$$
$$\vdots$$

$$\sum_{l_1=N-1} \cdots \sum_{l_{N-1}=2} \frac{a_{l_1l_1}}{(a_{l_1l_1}-a_{NN})\cdots (a_{l_{N-1}l_{N-1}}-a_{NN})(a_{11}-a_{NN})}.$$

$$w_{1}^{i} = \alpha_{1}^{1} a_{i1} + \sum_{l_{1}=1}^{i-1} \alpha_{2}^{l_{1}} \underbrace{a_{il_{1}} a_{l_{1}1}}_{\text{path of length}} \\ + \sum_{l_{1}=1}^{i-1} \sum_{l_{2}=1}^{l_{1}-1} \alpha_{3}^{l_{1}l_{2}} \underbrace{a_{il_{1}} a_{l_{1}l_{2}} a_{l_{2}1}}_{\text{path of length}} + \cdots \\ + \sum_{l_{1}=i-1}^{i-1} \cdots \sum_{l_{i-1}=2}^{l_{i-2}-1} \alpha_{1-1}^{l} \underbrace{a_{il_{1}} a_{l_{1}l_{2}} \cdots a_{l_{i-1}}}_{\text{path of length}},$$
(12)

where α_k^l are same sign scalars for different values of k and l. Existence of arborescence diverging from node 1 guarantees that $w_1^i \neq 0$. Hence, $w^i B \neq 0$ for i = 1, ..., N.

Since an arborescence diverging from the driver node is structurally controllable, we can conclude that any networked system which contains an arborescence subdigraph diverging from the driver node is structurally controllable.

Corollary 10. *A network with a globally reachable driver node is structurally controllable.*

In the above, we examine networks with single driver node. In what follows, we generalize the result for networks with multiple driver nodes.

4.2. Controllability of Networks with Multiple Driver Nodes. Consider a network of nodes with multiple driver nodes, which is expressed as follows:

$$\dot{x} = Ax + Bu,\tag{13}$$

where $B = r \operatorname{diag}\{b_1, b_2, \dots, b_m\} \in \mathbb{R}^{N \times m}$, where $r \operatorname{diag}\{\cdot\}$ refers to the rectangular diagonal matrix and b_r are positive scalars. The following theorem expresses the controllability condition in networks with multiple driver nodes.

Theorem 11. Consider the network expressed by (13) which consists of multiple driver nodes. The network is structurally controllable if and only if there is a path from at least one driver node to any arbitrary node.

Proof. For simplicity, we assume that nodes i = 1, ..., m are driver nodes of the network. Hence matrix *B* can be expressed as follows:

$$B = \begin{bmatrix} b_1 & 0 & \cdots & 0 \\ 0 & b_2 & \vdots \\ \vdots & \ddots & 0 \\ 0 & \cdots & 0 & b_m \\ & \mathbf{0}_{(N-m) \times m} \end{bmatrix}.$$
 (14)

Necessity Condition. We assume that there is node *j* and that there is no path from any input node to that node. According to Lemma A.1 in the Appendix, matrix series (8) has zero elements in columns 1 to *m* of row *j*. Therefore, $e^{A(t-\tau)}B$ has zero row *j*, which yields existence of $x^* = [\underbrace{0, \ldots, 0, \alpha, 0, \ldots, 0}]^T$ as an uncontrollable state of the system. *j*th element

Sufficiency Condition. We assume that the network contains m driver nodes. By zeroing out the weights of redundant links, we decompose the network into m components such that driver node k controls over nodes of component k. Hence, matrices A and B can be expressed as follows:

$$A = \begin{bmatrix} A_1 & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & A_m \end{bmatrix}, \qquad (15)$$
$$B = \begin{bmatrix} B_1 & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & B_m \end{bmatrix}.$$

If *l*th node in *k*th component is driver node, we have $B_k = [\underbrace{0, \ldots, 0, b_k, 0, \ldots, 0}_{\text{lth element}}]^T$ for $k = 1, \ldots, N$. Using Theorem 7, we obtain

$$\operatorname{Rank} \begin{bmatrix} \lambda I - A & B \end{bmatrix}$$

$$= \operatorname{Rank} \begin{bmatrix} \lambda I_{1} - A_{1} & B_{1} & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & \lambda I_{m} - A_{m} & B_{m} \end{bmatrix}, \quad (16)$$

where I_k are identity matrices with the same size as A_k for k = 1, ..., N. Using Theorem 9, for an arbitrary component k, $[\lambda I_k - A_k \ B_k]$ is full row-rank. Since $[\lambda I - A \ B]$ is block diagonal matrix with full row-rank block matrices, therefore

$$\operatorname{Rank}\left[\lambda I - A \quad B\right] = N. \tag{17}$$

For better underdressing, in what follows, we compare the proposed structural controllability condition and Liu's structural controllability condition. An example of Liu's structural controllability is presented in Figure 1. It is shown that controlling node 1 is not sufficient for full control (see Figure 1(a)). To gain full control, we must simultaneously control node 1 and any node among x_2 , x_3 (see Figure 1(b)). In contrast, in the proposed structural controllability (see Figure 2), controlling node 1 is sufficient for full control over the networked system.

5. Algorithm for Identifying Minimum Driver Nodes

We have shown the relationship between the structural controllability and graph reachability. Thus the problem of



FIGURE 2: An example of the proposed structural controllability.

 $N = 3, m = 1, \operatorname{srank}(\mathscr{C}) = 3$

examining the structural controllability of the networked systems described by (5) can be converted into graph reachability problem. Here, we are interested in determining the minimum number of driver nodes in a directed network, denoted by N_D , to obtain controllability over the networked systems. However, difficulties in identifying minimum number of driver nodes in large-scale networks lead to the requirement for a simple systematic method. In what follows, we propose a simple algorithm to determine the minimum number of driver nodes using graph reachability approach.

To check the graph reachability between each two arbitrary nodes, we present the following theorem.

Theorem 12. Consider a network of nodes with an associated structured adjacency matrix \mathcal{A}_s . For any two arbitrary nodes *i* and *j*, if (*j*, *i*)th element of the matrix $(\mathbf{I} - \mathcal{A}_s t)^{-1}$ is zero, then there is no path from node *i* to node *j*, where **I** is an identity matrix and *t* is a positive constant such that the spectral radius of $\mathcal{A}_s t$ is less than 1.

Proof. To prove this theorem, we first use the Taylor series expansion of the matrix inverse (see Lemma A.3 in the Appendix). Consider

$$\left(\mathbf{I} - \mathscr{A}_{s}t\right)^{-1} = \sum_{i=0}^{\infty} t^{i} \mathscr{A}_{s}^{i}.$$
(18)

Using Lemma A.2, $(\mathscr{A}_s^k)_{ji}$ for $k = 1, 2, ..., \infty$ is zero if and only if there is no path from node *i* to node *j*. Hence, the zeroness of the (j, i)th entry of \mathscr{A}_s^k for $k = 1, 2, ..., \infty$ leads to the zeroness of the (j, i)th entry of $(\mathbf{I} - \mathscr{A}_s t)^{-1}$.

Remark 13. Using Gershgorin's theorem [34], the suitable t which satisfies the condition in Theorem 12 is obtained as follows:

$$t = \left(\max_{\substack{i \\ j \neq i}}^{N} a_{ij} \right)^{-1} - \varepsilon,$$
(19)

where ε is a small number.

To illustrate the result in Theorem 12, let us consider the network in Figure 3.

The associated structured matrix can be defined as Boolean matrix as follows:

$$\mathscr{A}_{s} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$
 (20)



FIGURE 3: An example of a network with 4 nodes.

From (19), we obtain t = 0.1. Therefore, matrix $(\mathbf{I} - \mathcal{A}_s t)^{-1}$ is obtained in structured format as follows:

$$\begin{bmatrix} * & * & * & * \\ 0 & * & 0 & * \\ * & * & * & * \\ 0 & * & 0 & * \end{bmatrix},$$
 (21)

where * represents nonzero parameters such that in matrix (21), for example, entry (4, 1) is zero, which means that there is no path from node 1 to node 4. Since the network is small, driver nodes in Figure 3 can be easily identified, which are either node 2 or node 4. The same result can be obtained by examining $(I - \mathscr{A}_s t)^{-1}$. In matrix (21), columns full of nonzero elements represent globally reachable nodes. For columns which contain zero elements, we define *graph reachability index* as follows.

Definition 14. Node *u* is said to have graph reachability index *r*, if there are paths from *u* to maximum *r*, other nodes of the network.

Therefore, we can express the following corollary.

Corollary 15. In matrix $(\mathbf{I} - \mathcal{A}t)^{-1}$, columns with higher nonzero elements represent nodes with higher graph reachability index.

We can deduce that nodes with higher graph reachability index are suitable to be assigned as driver nodes.

Remark 16. To find the minimum driver nodes to obtain a structurally controllable network, we start by assigning the node with the highest graph reachability index as the driver node. Then, we remove all the nodes that are in the path rooted for the assigned driver node. We repeat the above procedure for the remaining network till the condition in Theorem 11 is satisfied.

Using the above mentioned results, we present a systematic algorithm to identify the minimum driver nodes in a networked system such that the structural controllability of the network is guaranteed. The algorithm for determining the minimum driver nodes of the network is described as follows.



FIGURE 4: An example of a network consisting of 30 nodes.

Consider graph \mathcal{G} with the associated structured adjacency matrix \mathcal{A}_s ,

Step 1. Compute graph reachability matrix $\mathscr{S} = (\mathbf{I}_N - \mathscr{A}_s t)^{-1}$.

Step 2. Identify the node with the highest graph reachability index by finding the columns of matrix S with the largest nonzero elements. If there is more than one node with the highest graph reachability index, we can randomly choose one of them.

Step 3. Assign that node as the driver node and zero out all the rows with the nonzero elements in the column associated with that driver node.

Step 4. Go back to Step 2 and repeat the procedure till all elements of matrix S are zero.

The above procedure is expressed in Algorithm 1.

Remark 17. It should be noted that the set of minimum driver nodes is usually not unique depending on the network configurations, and one can determine other sets with the same number of driver nodes.

6. Simulations

In this section, we present simulation results to illustrate the performance of the proposed method for networked systems of various sizes and topologies. For the numerical calculations and simulations, we used MATLAB software. For illustration purpose, we first consider a network with 30 nodes which are distributed randomly as depicted in Figure 4. The weights of links are randomly selected from [0, 1]. We compute $(\mathbf{I} - \mathcal{A}t)^{-1}$, where \mathcal{A} is the associated Laplacian matrix. The sparsity pattern of matrix $(\mathbf{I} - \mathcal{A}t)^{-1}$ is plotted in Figure 5, where the blue solid circles represent nonzero elements of the matrix. Applying the proposed algorithm, the driver nodes of the network are identified by magenta circles in Figure 6. The result of the first simulation is summarized in Table 1, where N is the number of nodes, L is the number Input: A; Method: (1) Compute *t* from (19); (2) Compute graph reachability matrix $\mathscr{S} = (\mathbf{I}_N - \mathscr{A}_s t)^{-1}$; (3) k = 0;(4) while $max(any(\mathscr{S})) \neq 0$ do (5) k = k + 1;% *k* represents the number of driver nodes. % *v* represents the vector of the number of nonzero elements in each column. $v = \operatorname{sum}(\mathscr{S} \neq 0, 1);$ (6)[value, ind] = max(v);% ind represents the column with the largest graph reachability index. (7)% Dnode represents the array of driver nodes. Dnode(k) = ind;(8)(9) $w = \operatorname{find}(\mathscr{S}(:, \operatorname{ind}));$ % *w* represents the rows with nonzero elements in the driver node column. (10) $\mathscr{S}(w, :) = 0;$ (11) end while

ALGORITHM 1: Finding driver nodes in each connected component.

 TABLE 1: The characteristics of the network represented in Figure 6.



of links, N_D is the computed number of driver nodes, and n_d is the computed density of driver nodes obtained by $n_d = N_d/N$.

To illustrate the capability of the purposed algorithm in dealing with large-scale networks, we consider a network of 1000 nodes which are distributed randomly within a square region as shown in Figure 7. The communication links are generated between neighboring nodes with the probability of 0.5. The weights of links are randomly selected from [0, 1]. The sparsity pattern of matrix $(\mathbf{I} - \boldsymbol{\mathscr{A}}t)^{-1}$ is plotted for the network in Figure 8. Applying the proposed algorithm, the driver nodes of the network are identified by magenta circles in Figure 9. The result of the second simulation is summarized in Table 2.



FIGURE 6: Driver nodes of the network identified by magenta circles.

TABLE 2: The characteristics of the network represented in Figure 9.

Ν	L	N_D	n _d
1000	1361	96	0.096

TABLE 3: The characteristics of some randomly generated networks.

Ν	L	N _D	n _d	
2000	2929	128	0.0640	
5000	6931	408	0.0816	
10000	13951	748	0.0748	

We applied the proposed algorithm on some randomly generated networks, and the results are illustrated in Table 3.

7. Conclusion

In this paper, we have addressed the structural controllability problem for networked systems. Despite the existing methods governed by some impractical assumptions on nodal dynamics and availability of input signals, we have examined structural controllability for networked systems in



FIGURE 7: An example of a network consisting of 1000 nodes.



FIGURE 8: The sparsity pattern of matrix $(\mathbf{I} - \mathcal{A}t)^{-1}$.



FIGURE 9: Driver nodes of the network identified by magenta circles.

practical framework. Using controllability analysis, we have presented the connection between networks driver nodes and graph reachability. Consequently, based on results on graph reachability, we have put forward a simple algorithm to determine minimum driver nodes in networked systems. Finally, simulation results have been presented to illustrate the performance of the proposed methods.

Appendix

Lemma A.1. Let $\mathscr{L} = \mathscr{A} - \mathscr{D}$ where \mathscr{A} is the adjacency matrix and $\mathscr{D} = \text{diag}\{a_{11}, a_{22}, \dots, a_{NN}\}$. Consider the following matrix:

$$\mathscr{P} = \sum_{i=1}^{N-1} \beta_i \mathscr{L}^i, \qquad (A.1)$$

where β_i are scalars. $(\mathcal{P})_{ij}$ is zero for any arbitrary values of β_i , *if there is no path of any length from node j to node i.*

Proof. To prove the lemma, we show that the (i, j)th element of all matrices \mathscr{L}^i where i = 1, 2, ..., N is zero, if there is no path of any length from node j to node i. Since there is no adjacent path from node j to i, then $a_{ij} = 0$. Therefore, the (i, j)th element of the \mathscr{L}^2 can be expressed as follows:

$$\left(\boldsymbol{\mathscr{Z}}^{2}\right)_{ij} = \sum_{\substack{k_{1}=1\\k_{1}\neq j}}^{N} a_{ik_{1}}a_{k_{1}j} = \left(\boldsymbol{\mathscr{A}}^{2}\right)_{ij}.$$
 (A.2)

Using Lemma A.2, we obtain $(\mathscr{D}^2)_{ij} = 0$. Therefore, the (i, j)th element of the \mathscr{D}^3 can be expressed as follows:

$$\left(\mathscr{Q}^{3}\right)_{ij} = \sum_{\substack{k_{1}=1\\k_{1}\neq i,j}}^{N} \sum_{\substack{k_{2}=1\\k_{2}\neq i,j}}^{N} a_{ik_{1}}a_{k_{1}k_{2}}a_{k_{2}j} = \left(\mathscr{A}^{3}\right)_{ij}.$$
 (A.3)

Using Lemma A.2, we obtain $(\mathcal{L}^3)_{ij} = 0$. Similarly, we can proceed for $\mathcal{L}^4, \mathcal{L}^5, \ldots, \mathcal{L}^{N-1}$ and show that $(\mathcal{L}^k)_{ij} = 0$ for $k = 1, 2, \ldots, N-1$.

Lemma A.2 (see [35]). Let \mathscr{A} be the adjacency matrix of a digraph \mathscr{G} ; then $(\mathscr{A}^k)_{ij}$ is greater than zero if and only if there is a path of length k from node j to node i.

Lemma A.3. For two arbitrary matrices A and B, the Taylor series expansion of the matrix inverse is expressed as follows:

$$(A+B)^{-1} = A^{-1} \sum_{i=0}^{\infty} (-1)^{i} (BA^{-1})^{i}, \qquad (A.4)$$

where the spectral radius of BA^{-1} is less than 1.

Conflict of Interests

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

References

- H. G. Tanner, "On the controllability of nearest neighbor interconnections," in *Proceedings of the 43rd IEEE Conference* on Decision and Control (CDC '04), vol. 3, pp. 2467–2472, December 2004.
- [2] M. Ji, A. Muhammad, and M. Egerstedt, "Leader-based multiagent coordination: controllability and optimal control," in *Proceedings of the American Control Conference*, pp. 1358–1363, June 2006.
- [3] A. Rahmani and M. Mesbahi, "On the controlled agreement problem," in *Proceedings of the American Control Conference*, pp. 1376–1381, IEEE, Minneapolis, Minn, USA, June 2006.
- [4] M. Ji and M. Egerstedt, "A graph-theoretic characterization of controllability for multi-agent systems," in *Proceedings of the American Control Conference (ACC '07)*, pp. 4588–4593, IEEE, New York, NY, USA, July 2007.
- [5] A. Rahmani, M. Ji, M. Mesbahi, and M. Egerstedt, "Controllability of multi-agent systems from a graph-theoretic perspective," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 162–186, 2009.
- [6] B. Liu, T. Chu, L. Wang, and G. Xie, "Controllability of a leader-follower dynamic network with switching topology," *IEEE Transactions on Automatic Control*, vol. 53, no. 4, pp. 1009– 1013, 2008.
- [7] B. Liu, T. Chu, L. Wang, Z. Zuo, G. Chen, and H. Su, "Controllability of switching networks of multi-agent systems," *International Journal of Robust and Nonlinear Control*, vol. 22, no. 6, pp. 630–644, 2012.
- [8] Z. J. Ji, Z. D. Wang, H. Lin, and Z. Wang, "Interconnection topologies for multi-agent coordination under leader-follower framework," *Automatica*, vol. 45, no. 12, pp. 2857–2863, 2009.
- [9] K. Peng and Y. Yang, "Leader-following consensus problem with a varying-velocity leader and time-varying delays," *Physica A*, vol. 388, no. 2-3, pp. 193–208, 2009.
- [10] W. Ni and D. Cheng, "Leader-following consensus of multiagent systems under fixed and switching topologies," *Systems & Control Letters*, vol. 59, no. 3-4, pp. 209–217, 2010.
- [11] L. Consolini, F. Morbidi, D. Prattichizzo, and M. Tosques, "Leader-follower formation control of nonholonomic mobile robots with input constraints," *Automatica*, vol. 44, no. 5, pp. 1343–1349, 2008.
- [12] R. Haghighi and C. C. Cheah, "On leader-based shape coordination," in *Proceedings of the 11th International Conference on Control, Automation, Robotics & Vision (ICARCV '10)*, pp. 404– 409, IEEE, Singapore, December 2010.
- [13] R. Haghighi and C. C. Cheah, "Multi-group coordination control for robot swarms," *Automatica*, vol. 48, no. 10, pp. 2526– 2534, 2012.
- [14] H. Su, G. Jia, and M. Z. Q. Chen, "Semi-global containment control of multi-agent systems with input saturation," *IET Control Theory & Applications*, vol. 8, no. 18, pp. 2229–2237, 2014.
- [15] H. Su and M. Z. Q. Chen, "Multi-agent containment control with input saturation on switching topologies," *IET Control Theory & Applications*, vol. 9, no. 3, pp. 399–409, 2015.
- [16] M. Porfiri and M. di Bernardo, "Criteria for global pinningcontrollability of complex networks," *Automatica*, vol. 44, no. 12, pp. 3100–3106, 2008.
- [17] Q. Song and J. Cao, "On pinning synchronization of directed and undirected complex dynamical networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 3, pp. 672–680, 2010.

- [18] C.-T. Lin, "Structural controllability," *IEEE Transactions on Automatic Control*, vol. 19, no. 3, pp. 201–208, 1974.
- [19] R. W. Shields and J. B. Pearson, "Structural controllability of multi-input linear systems," *IEEE Transactions on Automatic Control*, vol. AC-21, no. 2, pp. 203–212, 1976.
- [20] S. Hosoe and K. Matsumoto, "On the irreducibility condition in the structural controllability theorem," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 963–966, 1979.
- [21] J.-M. Dion, C. Commault, and J. van der Woude, "Generic properties and control of linear structured systems: a survey," *Automatica*, vol. 39, no. 7, pp. 1125–1144, 2003.
- [22] H. Mayeda, "On structural controllability theorem," *IEEE Transactions on Automatic Control*, vol. 26, no. 3, pp. 795–798, 1981.
- [23] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *Nature*, vol. 473, no. 7346, pp. 167–173, 2011.
- [24] S. Jafari, A. Ajorlou, and A. G. Aghdam, "Leader localization in multi-agent systems subject to failure: a graph-theoretic approach," *Automatica*, vol. 47, no. 8, pp. 1744–1750, 2011.
- [25] S. Sundaram and C. N. Hadjicostis, "Structural controllability and observability of linear systems over finite fields with applications to multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 1, pp. 60–73, 2013.
- [26] R. Haghighi and C. C. Cheah, "Distributed average consensus based on structural weight-balanceability," *IET Control Theory* & Applications, vol. 9, no. 2, pp. 176–183, 2015.
- [27] N. J. Cowan, E. J. Chastain, D. A. Vilhena, J. S. Freudenberg, and C. T. Bergstrom, "Nodal dynamics, not degree distributions, determine the structural controllability of complex networks," *PLoS ONE*, vol. 7, no. 6, Article ID e38398, 2012.
- [28] W. Tutte, Graph Theory, Addison-Wesley, 1984.
- [29] D. D. Siljak, Decentralized Control of Complex Systems, Academic Press, New York, NY, USA, 1991.
- [30] K. J. Reinschke, *Multivariable Control: A Graph-Theoretic Approach*, Springer, 1988.
- [31] P. R. Belanger, Control Engineering: A Modern Approach, Saunders College Publishing, 1995.
- [32] R. L. Williams and D. A. Lawrence, *Linear State-Space Control Systems*, John Wiley & Sons, 2007.
- [33] C. Sueur and G. Dauphin-Tanguy, "Bond-graph approach for structural analysis of MIMO linear systems," *Journal of the Franklin Institute*, vol. 328, no. 1, pp. 55–70, 1991.
- [34] R. S. Varga, Gershgorin and His Circles, Springer, Berlin, Germany, 2004.
- [35] G. Williams, *Linear Algebra with Applications*, Jones and Bartlett Publishers, 2012.

Research Article **Controllability of Multiagent Systems with a Directed Tree**

Li Wang¹ and Xiao Han²

¹Beijing Key Lab of Urban Intelligent Traffic Control Technology, North China University of Technology, Beijing 100144, China ²College of Science, North China University of Technology, Beijing 100144, China

Correspondence should be addressed to Xiao Han; tgncuthanxiao@163.com

Received 12 June 2015; Accepted 16 July 2015

Academic Editor: Michael Z. Q. Chen

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

This paper addresses the controllability problem of multiagent systems with a directed tree based on the classic agreement protocol, in which the information communication topologies being a directed tree and containing a directed tree are both investigated. Different from the literatures, a new method, the star transform, is proposed to study the controllability of multiagent systems with directed topology. Some sufficient and necessary conditions are given for the controllability of such multiagent system. Numerical examples and simulations are proposed to illustrate the theoretical results.

1. Introduction

Distributed coordination of multiple dynamic agents systems has become a hot topic of major interest [1–6] in recent years. Studies in this direction have been greatly inspired by the cooperative behavior in nature, such as bird flocks, fish schools, ant swarms, and bacteria colonies [7], as well as being driven by the broad applications in engineering fields [8–16], such as cooperative control of unmanned air vehicles (UAVs), schooling of underwater vehicles, formation control of multirobots, attitude alignment of satellite clusters, and congestion control of communication networks [17].

The controllability issue of multiagent systems is a key problem for coordinated control of multiagent systems and shows new features and difficulties. In general, the controllability of a multiagent network refers to transferring the follower agents of such system from any arbitrary initial states to any final state by controlling dynamics of leader agents under exchanged information between each other, in which the interplay between network topologies and agent dynamics plays an important role related to the controllability. The controllability problems were investigated for singleintegrator kinematics [17, 18], double-integrator dynamics [19], and high-order-integrator dynamics [20], respectively.

In 2004, Tanner [17] first studied a simple interconnected system model with a single leader that consists of multiple mobile agents with one-integrator dynamics, interconnected through nearest-neighbor rules. A necessary and sufficient condition was obtained for such system with fixed topology to be controllable by regulating the behaviors of the leader, which is assumed to be able to affect its neighbors but not be affected by other group members. In [18, 21], Liu et al. developed the controllability of the discrete system with switching topology and a single leader. The controllability of multiagent systems with multiple leaders based on fixed topology and switching topology was investigated in [22] and [23], respectively. Furthermore, the controllability of multiagent systems with double-integrator dynamics [19] and high-orderintegrator dynamics [20] is studied.

Notice that the results of [24–31] were studied based on an undirected nearest-neighbor topology. However, for the case of networks with directed topologies, as often encountered in practice, it is very hard to solve the controllability problems due to the complexity of the topology and make the controllability of dynamic networks an nontrivial new problem. To date, very few results available in the literature [32] for the controllability of multiagent systems with directed topologies were found. In the context of multiagent networks, we have focused on studying the controllability of multiagent systems with directed topologies in this paper. Some sufficient and necessary conditions for the controllability of such multiagent system with directed topologies are obtained. Compared to the existing works on the related problems [32], the contributions of this paper are summarized as follows: (1) the topology is directed; (2) a novel method, the star transform, is introduced; and (3) the geometric criteria for controllability of such multiagent system are given.

The rest of the paper is organized as follows. Section 2 states the problem formulation and some definitions. Section 3 gives the main results on the controllability. Section 4 presents numerical examples and simulation results. Section 5 summarizes the paper.

2. Preliminaries and Problem Formulation

2.1. Preliminaries. In this section, we briefly recall some basic notations and concepts in graph theory [33] which will be used in this paper.

A weighted directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ consists of a vertex set $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$ and an edge set $\mathcal{E} = \{(v_i, v_j) :$ $v_i, v_i \in \mathcal{V}$, where an *edge* is an ordered pair of distinct vertices of \mathcal{V} , and the nonsymmetric weighted adjacency matrix $\mathscr{A} = [a_{ij}]$, with $a_{ij} > 0$ if and only if $e_{ij} \in \mathscr{C}$ and $a_{ij} = 0$ if not. If all the elements of \mathcal{V} are unordered pairs, then the graph is called an *undirected graph*. If $v_i, v_j \in \mathcal{V}$, and $(v_i, v_j) \in \mathcal{E}$, then we say that v_i and v_j are adjacent or v_j is a neighbor of v_i . The neighborhood set of node v_i is denoted by $\mathcal{N}_i = \{v_i \in \mathcal{V} : (v_i, v_i) \in \mathcal{E}\}$. The number of neighbors of each vertex is its degree. A graph is called complete if every pair of vertices is adjacent. A *path* of length r from v_i to v_j in a graph is a sequence of r + 1 distinct vertices starting with v_i and ending with v_i such that consecutive vertices are adjacent. If there is a path between any two vertices of \mathcal{G} , then \mathcal{G} is *connected*. A *directed tree* is a directed graph, where the node without any parent is called root, and the root can be connected to any other nodes through paths. Other nodes have exactly one parent (e.g., Figure 1). A spanning tree of a digraph is a directed tree formed by graph edges that connect all the nodes of the graph. A binary tree of a digraph is a directed tree with at most two children for every node, and each child is designated as its left child or right child. A k*ary tree* is a directed tree with at most k children for every node. If every node has k children or no children, the kary tree is called a *full k-ary tree*. The *degree matrix* $\Delta(\mathcal{G})$ of \mathscr{G} is a diagonal matrix with rows and columns indexed by \mathcal{V} , in which the (v_i, v_i) -entry is the degree of vertex v_i . The symmetric matrix defined as

$$L(\mathscr{G}) = \Delta(\mathscr{G}) - A(\mathscr{G}) \tag{1}$$

is the *Laplacian* of \mathcal{G} . The Laplacian is always symmetric and positive semidefinite, and the algebraic multiplicity of its zero eigenvalue is equal to the number of connected components in the graph. For a weighted directed graph \mathcal{G} with *n* nodes, the *out-degree* and *in-degree* of node v_i in a weighted directed graph \mathcal{G} with *n* nodes are, respectively, defined as

$$\deg_{\text{out}} (v_i) = \sum_{j=1}^n a_{ji},$$

$$\deg_{\text{in}} (v_i) = \sum_{j=1}^n a_{ij}.$$
(2)



FIGURE 1: A directed tree.



FIGURE 2: The topology \mathcal{G} .

2.2. Problem Formulation. Consider a multiagent system with a directed topology which is a directed tree, composed of *n* agents. Choose the first agent (labeled 1) as the root and the remainder n - 1 agents (labeled from 2 to *n*) as children or followers, and each agent moves according to the following dynamics:

$$\dot{x}_{i}(t) = \sum_{j \in \mathcal{N}_{i}} a_{ij} \left(x_{j}(t) - x_{i}(t) \right) + z_{i}, \quad i = 1, \dots, n, \quad (3)$$

where $x_i \in \mathbb{R}^1$ is the state of agent *i* and $i \in \underline{n}, \underline{n}$ is the index set (1, 2, ..., n); \mathcal{N}_i is the neighbor set of agent *i*; $A = [a_{ij}] \in \mathbb{R}^{m \times m}$ is a matrix describing the interaction or coupling weight between agents, with $a_{ij} \ge 0$ and $a_{ii} = 0$. Since the topology is a directed tree, the root agent plays a special role and is responsible for receiving external input or signal and conveying information to the children agents, and hence in model (3), the external input or signal only acts on the root agent (e.g., Figure 2). So, $z_1 \ne 0$, $z_i = 0$, and $a_{1i} = 0$ for $\forall i = 2, ..., n$.

Let $x = (x_1, x_2, ..., x_n)^T$ be the state vector of all the agents; then system (3) follows that

$$\dot{x} = Ax + bz_1,\tag{4}$$

where

$$A = \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ A_{1(n-1) \times 1} & A_{e(n-1) \times (n-1)} \end{bmatrix} \in \mathbb{R}^{n \times n},$$

$$b = \begin{bmatrix} 1 \\ \mathbf{0}_{(n-1) \times 1} \end{bmatrix} \in \mathbb{R}^{n \times 1},$$

(5)

with

$$A_{1} = \begin{bmatrix} a_{21}, a_{31}, \dots, a_{n1} \end{bmatrix}^{T} \in R^{(n-1) \times 1},$$

$$A_{e} = \begin{bmatrix} a_{22} & \cdots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{n2} & \cdots & a_{nn} \end{bmatrix} \in R^{(n-1) \times (n-1)}.$$
(6)

Define a star transform matrix *S* as

$$S = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & -1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & -1 \end{bmatrix}.$$
 (7)

It can be easily seen that $S = S^{-1}$, and for the convenience of matrix operations, we partition *S* as

$$S = \begin{bmatrix} 1 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix}.$$
 (8)

Obviously, $S_e \in R^{(n-1)\times(n-1)}$ is a negative identity matrix. Denote $y = [y_1, y_e^T]^T$, where $y_e = [y_2, y_3, \dots, y_n]^T$. Let

$$y = Sx, \tag{9}$$

and then we can get $y_1 = x_1$ and $y_e = [x_1 - x_2, x_1 - x_3, \dots, x_1 - x_n]^T$. Therefore, we can have an equivalent system of system (4) as follows:

$$\dot{y} = SAS^{-1}y + Sbz_1. \tag{10}$$

According to block matrix multiplication, we can have

$$\begin{split} \dot{y}_{1} &= \begin{bmatrix} 1 & \mathbf{0}_{1 \times (n-1)} \end{bmatrix} \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ A_{1(n-1) \times 1} & A_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 1 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} y_{1} \\ y_{e} \end{bmatrix} + \begin{bmatrix} 1 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{l}_{(n-1) \times 1} \end{bmatrix} \begin{bmatrix} 1 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} y_{1} \\ y_{e} \end{bmatrix} + z_{1} = \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ y_{e} \end{bmatrix} + z_{1} = z_{1}, \\ \dot{y}_{e} &= \begin{bmatrix} \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ A_{1(n-1) \times 1} & A_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 1 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{0}_{(n-1) \times 1} \end{bmatrix} z_{1} = \begin{bmatrix} \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{0}_{(n-1) \times 1} & A_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} y_{1} \\ y_{e} \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{0}_{(n-1) \times 1} \end{bmatrix} z_{1} = \begin{bmatrix} \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{0}_{(n-1) \times 1} & A_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} y_{1} \\ y_{e} \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{0}_{(n-1) \times 1} \end{bmatrix} z_{1} = \begin{bmatrix} \mathbf{0}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} y_{1} \\ y_{e} \end{bmatrix} + \mathbf{l}_{(n-1) \times 1} z_{1} \\ &= S_{e(n-1) \times (n-1)} A_{e(n-1) \times (n-1)} S_{e(n-1) \times (n-1)} y_{e} + \mathbf{l}_{(n-1) \times 1} z_{1} = A_{e(n-1) \times (n-1)} y_{e} + \mathbf{l}_{(n-1) \times 1} z_{1}. \end{split}$$

Furthermore, we can know

$$\dot{y} = \begin{bmatrix} \dot{y}_{1} \\ \dot{y}_{e} \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ A_{1(n-1) \times 1} & A_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 1 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} y_{1} \\ y_{e} \end{bmatrix} + \begin{bmatrix} 1 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{l}_{(n-1) \times 1} & S_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{0}_{(n-1) \times 1} \end{bmatrix} z_{1} = \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{0}_{(n-1) \times 1} & A_{e(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} y_{1} \\ y_{e} \end{bmatrix} + \begin{bmatrix} 1 \\ \mathbf{1}_{(n-1) \times 1} \end{bmatrix} z_{1} = \widetilde{A}y + \widetilde{b}z_{1},$$
(12)

$$z \longrightarrow (1) \longrightarrow (2) \longrightarrow (3) \longrightarrow \cdots \longrightarrow (n)$$

FIGURE 3: A directed path.

where

$$\widetilde{A} = \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{0}_{(n-1) \times 1} & A_{e(n-1) \times (n-1)} \end{bmatrix} \in \mathbb{R}^{n \times n},$$

$$\widetilde{b} = \mathbf{1}_{n \times 1} \in \mathbb{R}^{n \times 1}.$$
(13)

3. Main Results

In this section, we first give the definition of the controllability of multiagent systems and some lemmas.

Definition 1. A nonzero state x_0 of system (4) is controllable at the initial time t_0 if there exists a finite time t_f and a control input $z_1(t)$, such that $x(t_0) = x_0$ and $x(t_f) = 0$. If any nonzero state x_0 of system (4) is controllable, then system (4) is said to be controllable.

Lemma 2 (see [2]). System (4) is controllable iff
$$Q_c$$
 has row full rank, where the controllability matrix of system (4) is defined as

$$Q_c = \begin{bmatrix} \widetilde{b} & \widetilde{A}\widetilde{b} & \cdots & \widetilde{A}^{n-1}\widetilde{b} \end{bmatrix}.$$
(14)

From Lemma 2, we can have the following result.

Theorem 3. System (4) is controllable iff \widetilde{Q}_c has row full rank, where

$$\widetilde{Q}_c = \begin{bmatrix} A_1 & A_e A_1 & \cdots & A_e^{(n-2)} A_1 \end{bmatrix}.$$
(15)

Proof. It is easy to find

$$\widetilde{A}^{k} = \begin{bmatrix} 0 & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{0}_{(n-1) \times 1} & A_{e}^{k} \\ \mathbf{0}_{(n-1) \times (n-1)} \end{bmatrix} \in \mathbb{R}^{n \times n}, \quad (16)$$

and $A_{e(n-1)\times(n-1)}\mathbf{1}_{(n-1)\times 1} = -A_{1(n-1)\times 1}$; then

$$Q_{c} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \mathbf{1}_{(n-1)\times 1} & A_{e(n-1)\times(n-1)}\mathbf{1}_{(n-1)\times 1} & A_{e}^{2}_{(n-1)\times(n-1)}\mathbf{1}_{(n-1)\times 1} & \cdots & A_{e}^{(n-1)}_{(n-1)\times(n-1)}\mathbf{1}_{(n-1)\times 1} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \mathbf{1}_{(n-1)\times 1} & -\widetilde{Q}_{c} \end{bmatrix}, \quad (17)$$

where $\widetilde{Q}_c = \begin{bmatrix} A_1 & A_e A_1 & \cdots & A_e^{(n-2)} A_1 \end{bmatrix}$. Therefore, Q_c has full row rank if and only if \widetilde{Q}_c has full row rank. This completes the proof.

Note that, for the multiagent system with multiple agents and high dimension, the controllable matrix of such system is too complex to calculate. But it is so easy to compute the eigenvalues of the system matrix using the PBH rank method by MATLAB. Next, we give a more simple and more easily checkable method.

Theorem 4 (PBH rank test). *System (4) is controllable iff system (4) satisfies one of the following conditions:*

- (i) rank($sI A_e, A_1$) = n 1, $\forall s \in \mathbb{C}$;
- (ii) rank $(\lambda_i I A_e, A_1) = n-1$, where $\lambda_i, \forall i = 1, 2, ..., n-1$, is the eigenvalue of matrix A_e .

Proof. The proof is similar to that of Theorem 3 in [22], here omitted. \Box

Since the topology graph is a directed tree, we can relabel all agents so that A is a lower triangular matrix, and then A_1

is a lower triangular matrix. So we can have the following results.

Theorem 5. System (4) is controllable if coupling weights among agents are all distinct.

Proof. The proof is similar to that of Theorem 3.1 in [32], here omitted. \Box

Corollary 6. A directed path is controllable.

Proof. From Figure 3, we have

$$A_{e} = \begin{bmatrix} -a_{21} & 0 & 0 & \cdots & 0 & 0 \\ a_{32} & -a_{32} & 0 & \cdots & 0 & 0 \\ 0 & a_{43} & -a_{43} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -a_{n-1,n-2} & 0 \\ 0 & 0 & 0 & \cdots & a_{n,n-1} & -a_{n,n-1} \end{bmatrix}.$$
 (18)



FIGURE 4: Containing a directed tree.

By computing, then

$$\widetilde{Q}_{c} = \begin{bmatrix} A_{e(n-1)\times(n-1)} \mathbf{1}_{(n-1)\times 1} & A_{e}^{2}_{(n-1)\times(n-1)} \mathbf{1}_{(n-1)\times 1} & \cdots & A_{e}^{(n-1)}_{(n-1)\times(n-1)} \mathbf{1}_{(n-1)\times 1} \end{bmatrix} \\
= \begin{bmatrix} -a_{21} & a_{21}^{2} & \cdots & (-1)^{n-2} a_{21}^{n-2} & (-1)^{n-1} a_{21}^{n-1} \\ 0 & -a_{32}a_{21} & \cdots & * & * \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ 0 & 0 & \cdots & -a_{n-1,n-2} \cdots a_{32}a_{21} & * \\ 0 & 0 & \cdots & 0 & -a_{n,n-1}a_{n-1,n-2} \cdots a_{32}a_{21} \end{bmatrix}.$$
(19)

Obviously, \widetilde{Q}_c has row full rank. Hence, the directed path is controllable.

Corollary 7. *A k-ary tree is uncontrollable if the weights of two children of some one agent are identical.*

Furthermore, in fact, the network can be a digraph, which is not a directed tree but contains a directed tree (e.g., Figure 4). We have the following result.

Theorem 8. System (4) containing a directed tree is controllable if the following conditions are satisfied:

(i) The eigenvalues of A_e are all distinct.

(ii) The elements of $U^{-1}A_1$ are all nonzero, where $U = [\varepsilon_2, \ldots, \varepsilon_n]$, with ε_i being eigenvector of A_e corresponding to eigenvalue λ_i $(i = 2, \ldots, n)$.

Proof. Because A is a lower triangular matrix, A_e is also a lower triangular matrix. Then the diagonal elements of A_e are its eigenvalue $\lambda_i = -\deg_{in}(v_i)$ (i = 2, 3, ..., n). Let $U \triangleq [\varepsilon_2, ..., \varepsilon_n]$, where ε_i is eigenvector of A_e corresponding to eigenvalue λ_i (i = 2, ..., n). If the eigenvalues of A_e are all distinct, then U is invertible. Let $\tilde{x} = U^{-1}x$; then

$$\dot{\tilde{x}} = U^{-1}\dot{x} = U^{-1}A_eU\tilde{x} + U^{-1}A_1u = \Lambda\tilde{x} + U^{-1}A_1u, \quad (20)$$



FIGURE 5: A directed tree topology.

where $\Lambda = U^{-1}A_eU = \text{diag}(\lambda_2, \dots, \lambda_n)$. So

$$\begin{bmatrix} \lambda_{i}I - \Lambda, U^{-1}A_{1} \end{bmatrix} = \begin{bmatrix} \lambda_{i} - \lambda_{2} & 0 & \cdots & 0 & b_{2} \\ 0 & \lambda_{i} - \lambda_{3} & \cdots & 0 & b_{3} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \lambda_{i} - \lambda_{n} & b_{n} \end{bmatrix}.$$
 (21)

Obviously, if conditions (i) and (ii) hold, then the system is controllable. This completes the proof. $\hfill \Box$

4. Simulation Study

In this section, we give some numerical examples and simulations to illustrate the effectiveness of the proposed theoretical results.

4.1. Example 1. Consider a five-agent network with six agents and with a directed tree described by Figure 5. Let $a_{21} = 1$, $a_{31} = 2$, $a_{42} = 3$, $a_{52} = 4$, and $a_{63} = 5$, and system (4) is defined by

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 2 & 0 & -2 & 0 & 0 & 0 \\ 0 & 3 & 0 & -3 & 0 & 0 \\ 0 & 4 & 0 & 0 & -4 & 0 \\ 0 & 0 & 5 & 0 & 0 & -5 \end{bmatrix},$$

$$b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$
(22)



FIGURE 6: Aligning in a straight line.

Through the star transform, we can have

$$A_{e} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 & 0 \\ 3 & 0 & -3 & 0 & 0 \\ 4 & 0 & 0 & -4 & 0 \\ 0 & 5 & 0 & 0 & -5 \end{bmatrix},$$

$$A_{1} = \begin{bmatrix} 1 \\ 2 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$
(23)

By calculation, $Rank(\widetilde{Q}_c) = 5$. According to Theorem 3, the system is controllable.

Figure 6 shows the simulation result. The vertices (the agents, the black star dots) begin from random initial positions. Interconnections are depicted as lines connecting the corresponding vertices. Beginning from this initial configuration, the vertices are ultimately being controlled to a straight-line configuration.

4.2. *Example 2.* Consider a five-agent network containing a directed tree with six agents described by Figure 7. Let $a_{21} = 1$, $a_{31} = 2$, $a_{32} = 3$, $a_{41} = 4$, $a_{42} = 5$, $a_{51} = 6$, and $a_{63} = 7$, and system (4) is defined by

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 2 & 3 & -5 & 0 & 0 & 0 \\ 4 & 5 & 0 & -9 & 0 & 0 \\ 6 & 0 & 0 & 0 & -6 & 0 \\ 0 & 0 & 7 & 0 & 0 & -7 \end{bmatrix}$$



FIGURE 7: Containing a directed tree topology.

$$b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Through the star transform, we can have

$$A_{e} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ 3 & -5 & 0 & 0 & 0 \\ 5 & 0 & -9 & 0 & 0 \\ 0 & 0 & 0 & -6 & 0 \\ 0 & 7 & 0 & 0 & -7 \end{bmatrix},$$

$$A_{1} = \begin{bmatrix} 1 \\ 2 \\ 4 \\ 6 \\ 0 \end{bmatrix}.$$
(25)

By calculation, the eigenvalues of A_e are $\{-7, -5, -9, -1, -6\}$, and

$$U = \begin{bmatrix} 0 & 0 & 0 & 0.6065 & 0 \\ 0 & 0.2747 & 0 & 0.4549 & 0 \\ 0 & 0 & 1.0000 & 0.3790 & 0 \\ 0 & 0 & 0 & 0 & 1.0000 \\ 1.0000 & 0.9615 & 0 & 0.5307 & 0 \end{bmatrix},$$
$$U^{-1}$$

	[1.7500	-3.5000	0	0	1.0000	
	-2.7300	3.6401	0	0	0	
=	-0.6250	0	1.0000	0	0	,
	1.6489	0	0	0	0	
	0	0	0	1.0000	0	



FIGURE 8: Forming a regular triangle.

and then $U^{-1}A_1 = [5.2500, -4.5501, -3.3750, -1.6489, -6.0000]^T$. According to Theorem 8, the system is controllable.

Figure 8 shows the simulation result, in which the agents begin from this initial configuration and are ultimately being controlled to a regular triangle configuration.

5. Conclusion

(24)

(26)

In this paper, we have investigated the controllability of continuous time networked systems based on consensus protocol in directed graph which is a directed tree or contains a directed tree. By applying the star transform, we can simplify the system and get an equivalent reduced-order system. We have obtained the controllability conditions of the original system from the equivalent reduced-order system.

Conflict of Interests

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

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant no. 51308005, "The-Great-Wall-Scholar" Candidate Training-Plan of North China University of Technology, Construction Plan for Innovative Research Team of North China University of Technology, and the Plan Training Project of Excellent Young Teacher of North China University of Technology, the special project of North China University of Technology (no. XN085).

References

 E. Wang and S. Shi, *Higher Algebra*, Higher Education Press, 3rd edition, 2003.

- [2] D. Zheng, *Linear System Theory*, Tsinghua University Press, Beijing, China, 2nd edition, 2002.
- [3] M. Cao, S. Zhang, and M. K. Camlibel, "A class of uncontrollable diffusively coupled multiagent systems with multichain topologies," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 465–469, 2013.
- [4] X. Wang, X. Li, and G. Chen, Network Science: An Introduction, Higher Education Press, 2012.
- [5] B. Liu, X. Wang, H. Su, H. Zhou, Y. Shi, and R. Li, "Adaptive synchronization of complex dynamical networks with timevarying delays," *Circuits, Systems, and Signal Processing*, vol. 33, no. 4, pp. 1173–1188, 2014.
- [6] A.-H. Zhang, S.-J. Zhang, J. Chen, H.-B. Guo, X.-R. Kong, and F. Wang, "Graph-theoretic characterization of controllability for multi-agent systems," *Control and Decision*, vol. 26, no. 11, pp. 1621–1631, 2011.
- [7] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520– 1533, 2004.
- [8] H. Su, Z. Rong, M. Z. Q. Chen, X. Wang, G. Chen, and H. Wang, "Decentralized adaptive pinning control for cluster synchronization of complex dynamical networks," *IEEE Transactions on Cybernetics*, vol. 43, no. 1, pp. 394–399, 2013.
- [9] Z. Wang and J. Zhi, "Controllability of discrete-time multi-agent systems with time-delay in state," *Journal of Qingdao University*, vol. 24, no. 2, pp. 1–5, 2009.
- [10] Z. Ji, H. Lin, and T. H. Lee, "A graph theory based characterization of controllability for multi-agent systems with fixed topology," in *Proceedings of the 47th IEEE Conference on Decision and Control*, pp. 5262–5267, Cancun, Mexico, December 2008.
- [11] H. Su, G. Jia, and M. Z. Chen, "Semi-global containment control of multi-agent systems with input saturation," *IET Control Theory & Applications*, vol. 8, no. 18, pp. 2229–2237, 2014.
- [12] M. Ji and M. Egerstedt, "A graph-theoretic characterization of controllability for multi-agent systems," in *Proceedings of the American Control Conference (ACC '07)*, pp. 4588–4593, IEEE, New York, NY, USA, July 2007.
- [13] Y. Zheng, Y. Zhu, and L. Wang, "Consensus of heterogeneous multi-agent systems," *IET Control Theory & Applications*, vol. 5, no. 16, pp. 1881–1888, 2011.
- [14] S. Jafari, A. Ajorlou, A. G. Aghdam, and S. Tafazoli, "On the structural controllability of multi-agent systems subject to failure: a graph-theoretic approach," in *Proceedings of the 49th IEEE Conference on Decision and Control (CDC '10)*, pp. 4565–4570, December 2010.
- [15] H. Su, N. Zhang, M. Z. Chen, H. Wang, and X. Wang, "Adaptive flocking with a virtual leader of multiple agents governed by locally Lipschitz nonlinearity," *Nonlinear Analysis. Real World Applications*, vol. 14, no. 1, pp. 798–806, 2013.
- [16] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *Nature*, vol. 473, no. 7346, pp. 167–173, 2011.
- [17] H. G. Tanner, "On the controllability of nearest neighbor interconnections," in *Proceedings of the IEEE Conference on Decision* and Control, vol. 3, pp. 2467–2472, 2004.
- [18] B. Liu, T. Chu, L. Wang, and G. Xie, "Controllability of a leaderfollower dynamic network with switching topology," *IEEE Transactions on Automatic Control*, vol. 53, no. 4, pp. 1009–1013, 2008.

- [19] B. Liu, H. Feng, L. Wang et al., "Controllability of second-order multiagent systems with multiple leaders and general dynamics," *Mathematical Problems in Engineering*, vol. 2013, Article ID 587569, 6 pages, 2013.
- [20] F. Jiang, L. Wang, G. Xie, Z. Ji, and Y. Jia, "On the controllability of multiple dynamic agents with fixed topology," in *Proceedings* of the American Control Conference (ACC '09), pp. 5665–5670, June 2009.
- [21] B. Liu, T. Chu, L. Wang, and G. Xie, "Controllability of a class of multi-agent systems with a leader," in *Proceedings of the American Control Conference (ACC '06)*, pp. 2844–2849, IEEE, Minneapolis, Minn, USA, June 2006.
- [22] B. Liu, H. Su, R. Li, D. Sun, and W. Hu, "Switching controllability of discrete-time multi-agent systems with multiple leaders and time-delays," *Applied Mathematics and Computation*, vol. 228, pp. 571–588, 2014.
- [23] B. Liu, W. Hu, J. Zhang, and H. Su, "Controllability of discretetime multi-agent systems with multiple leaders on fixed networks," *Communications in Theoretical Physics*, vol. 58, no. 6, pp. 856–862, 2012.
- [24] K. Liu, G. Xie, W. Ren, and L. Wang, "Consensus for multiagent systems with inherent nonlinear dynamics under directed topologies," *Systems & Control Letters*, vol. 62, no. 2, pp. 152–162, 2013.
- [25] H. Su, M. Z. Q. Chen, J. Lam, and Z. Lin, "Semi-global leaderfollowing consensus of linear multi-agent systems with input saturation via low gain feedback," *IEEE Transactions on Circuits* and Systems-I. Regular Papers, vol. 60, no. 7, pp. 1881–1889, 2013.
- [26] Y. Sun and L. Wang, "Consensus of multi-agent systems in directed networks with nonuniform time-varying delays," *IEEE Transactions on Automatic Control*, vol. 54, no. 7, pp. 1607–1613, 2009.
- [27] J. Yu and L. Wang, "Group consensus of multi-agent systems with directed information exchange," *International Journal of Systems Science*, vol. 43, no. 2, pp. 334–348, 2012.
- [28] H. Su, M. Z. Q. Chen, X. Wang, and J. Lam, "Semiglobal observer-based leader-following consensus with input saturation," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 6, pp. 2842–2850, 2014.
- [29] J. Yu and L. Wang, "Group consensus in multi-agent systems with switching topologies and communication delays," *Systems* & Control Letters, vol. 59, no. 6, pp. 340–348, 2010.
- [30] Y. Zheng and L. Wang, "Containment control of heterogeneous multi-agent systems," *International Journal of Control*, vol. 87, no. 1, pp. 1–8, 2014.
- [31] H. Su and M. Z. Q. Chen, "Multi-agent containment control with input saturation on switching topologies," *IET Control Theory & Applications*, vol. 9, no. 3, pp. 399–409, 2015.
- [32] D. Guo, G. Yan, and Z. Lin, "Distributed verification of controllability for weighted out-tree based topology," in *Proceedings of the 2012 American Control Conference*, pp. 1507–1512, June 2012.
- [33] L. Hsu and C. Lin, *Graph Theory and Interconnection Networks*, CRC Press, 2008.