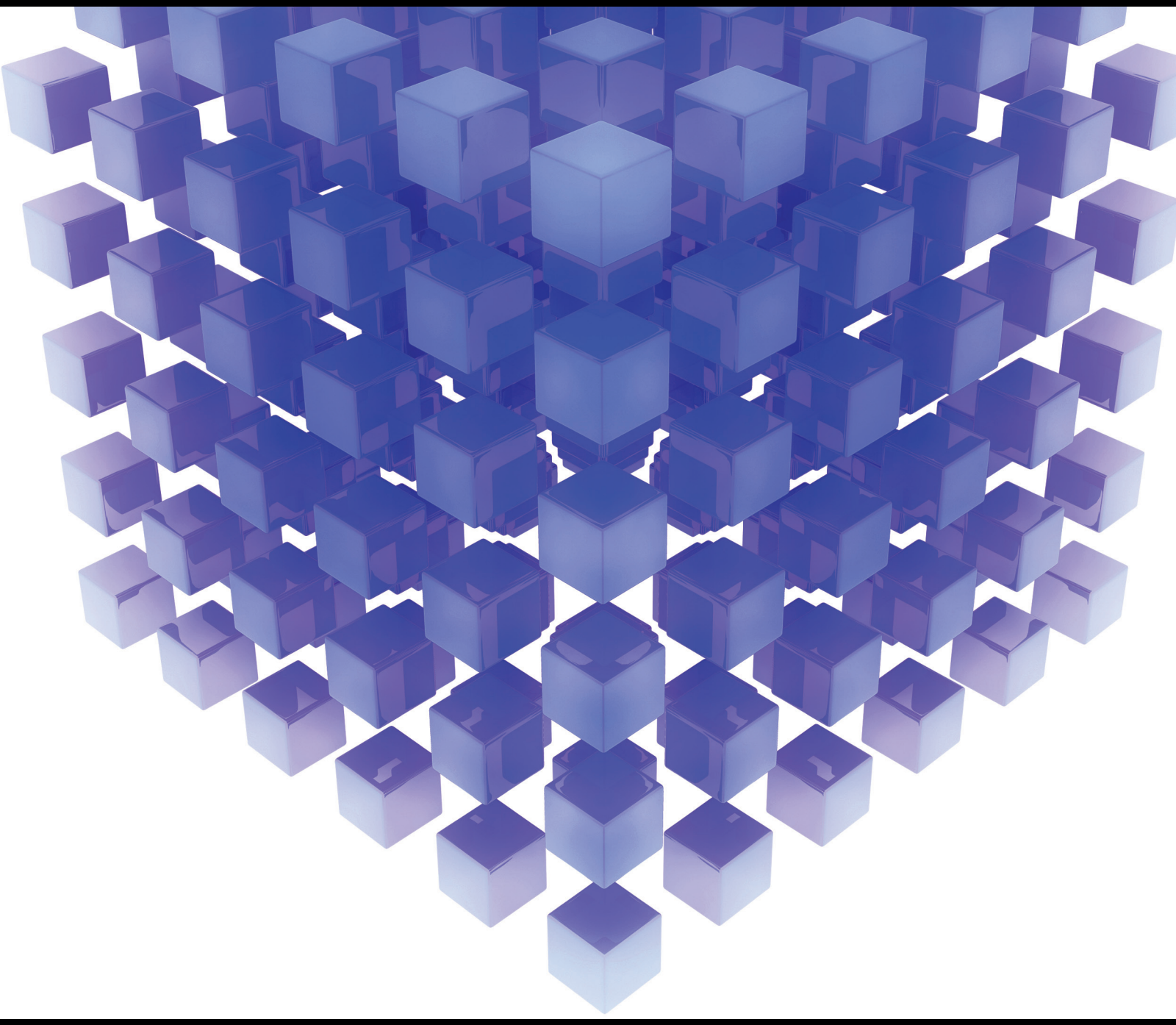
Mathematical Problems in Engineering

Modeling and Control of Complex Networked Systems

Guest Editors: Guoqiang Hu, Wenwu Yu, Guanghui Wen, Housheng Su, and Ying Tan



Modeling and Control of Complex Networked Systems

Mathematical Problems in Engineering

Modeling and Control of Complex Networked Systems

Guest Editors: Guoqiang Hu, Wenwu Yu, Guanghui Wen, Housheng Su, and Ying Tan

Editorial Board

Contents

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 831582, 2 pages
http://dx.doi.org/10.1155/2014/831582

Editorial
Modeling and Control of Complex Networked Systems

Guoqiang Hu,[1] Wenwu Yu,[2] Guanghui Wen,[2] Housheng Su,[3] and Ying Tan[4]

[1] *School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore*
[2] *Research Center for Complex Systems and Network Sciences, Department of Mathematics, Southeast University, Nanjing, China*
[3] *School of Automation, Huazhong University of Science and Technology, Wuhan, China*
[4] *Department of Electrical and Electronic Engineering, The University of Melbourne Parkville, VIC, Australia*

Correspondence should be addressed to Guoqiang Hu; gqhu@ntu.edu.sg

Received 8 July 2014; Accepted 8 July 2014; Published 21 July 2014

Complex networked systems refer to networks of interacting agents, which have been actively studied due to their fundamental scientific, technological, financial, and societal significance. The commonly studied complex networks include but are not limited to social networks, communication networks, sensor networks, actuator networks, economic networks, transportation networks, biological networks, power networks, and autonomous robot networks.

This special issue focuses on modeling and control of complex networked systems. It tries to understand not only the complex networked systems' characteristics but also the design of control strategies to achieve desired collective behaviors.

Call for papers has been carefully prepared by the guest editors and posted on the journal's web page, which has received much attention from researchers in different scientific communities. We have received 31 papers in this research fields. All manuscripts submitted to this special issue went through a thorough peer-refereeing process. Based on the reviewers' reports, 15 original research articles are finally accepted. The contents of this special issue contain modeling, analysis and control of complex networks, collective behaviors in multiagent systems, and analysis and synthesis of complex networked systems. The contents are summarized as follows.

(1) Modeling, Analysis, and Control of Complex Networks. In "*A dynamic microblog network and information dissemination in "@" mode*" by M. Tang et al., a dynamic microblog model is proposed based on the users' behaviors. "*Reorganizing complex network to improve large-scale multiagent teamwork*" by Y. Xu et al. studies some factors that affect team performance and proposes an integrated network adjustment algorithm to reorganize the network to expedite the team performance. "*A comparison of online social networks and real-life social networks: a study of Sina Microblogging,*" by D. Zhang and G. Guo, explores the structural characteristics of Sina Microblogging and reveals the similarities and differences between online social networks and real-life social networks. "*Local community detection in complex networks based on maximum cliques extension*" by M. Fanrong et al. focuses on detecting local community structure in complex networks. "*Matthew effect of the random drift on the evolution of cooperation,*" by C. Liu and R. Li, studies the effect of random drift on evolutionary Prisoner's dilemma on regular lattices. Synchronizability of a class of local-world dynamical networks is studied in "*Synchronization in a novel local-world dynamical network model*" by J. Tang and P. Liu. Synchronization problems are considered in "*Synchronization control of time-varying complex dynamic network with non-identical nodes and coupling time-delay*" by Y. Zhang et al. and "*Adaptive asymptotical synchronization for stochastic complex networks with time-delay and Markovian switching*" by X. Jiang et al. These two articles consider the synchronization problem for a class of time-varying complex dynamic network and stochastic complex dynamic networks, respectively. "*Pinning Lur'e complex networks via output feedback control,*" by F. Liu et al., investigates the pinning synchronization in a

network of Lur'e dynamical systems based on distributed output feedback control without using the full state information of network nodes.

(2) Collective Behaviors in Multiagent Systems. "*Distributed cooperative current-sharing control of parallel chargers using feedback linearization,*" by J. Liu et al., proposes a distributed current-sharing scheme to address the output current imbalance problem for the parallel chargers in the energy storage type light rail vehicle system. In "*Formation control of multirobot based on I/O feedback linearization and potential function*" by J. Dong el al., a formation control method of multirobot system based on potential function is proposed. A new control law is designed by choosing a proper potential function and employing Lyapunov stability theory.

(3) Analysis and Synthesis of Complex Networked Systems. "*Finite-time fault detection for large-scale networked systems with randomly occurring nonlinearity and fault,*" by Y. Zhang et al., investigates the finite-time fault detection problem for a class of nonlinear quantized large-scale networked systems. Moving target positioning based on a distributed camera network is studied by L. Zhao et al. in "*Moving target positioning based on a distributed camera network*." "*Dynamic request routing for online video-on-demand service: a Markov Decision Process approach,*" by J. Wan et al., investigates the request routing problem in the CDN-based video-on-demand system in which Markov decision process (MDP) and bounded-parameter MDP (BMDP) are used in the formulation of the system. "*Unified finite horizon H∞ fusion filtering for networked dynamical system,*" by C. Wen et al., addresses the H∞ fusion filtering problem for networked dynamical systems under different measurement arrival scenes in a unified manner.

Note that the selected topics and papers are not a comprehensive representation of the area of the special issue. It is also worth noting that the published papers in this special issue do provide some recent advances in the field of complex networked systems, which could benefit the current research more or less.

Acknowledgments

We would like to thank the authors for their contributions. We also thank anonymous reviewers for their time and help in assessing the manuscripts. Our special thanks go to the editorial board members of this journal owing to their great support and help for this special issue.

Guoqiang Hu
Wenwu Yu
Guanghui Wen
Housheng Su
Ying Tan

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 564058, 7 pages
http://dx.doi.org/10.1155/2014/564058

Research Article

Adaptive Asymptotical Synchronization for Stochastic Complex Networks with Time-Delay and Markovian Switching

Xueling Jiang,[1,2] Shuijie Qin,[1] Dongbing Tong,[3,4] and Liping Wang[5]

[1] *Laboratory for Photoelectric Technology and Application, Guizhou University, Guiyang, Guizhou 550025, China*
[2] *College of Computer and Information Engineering, Guangxi Teachers Education University, Nanning, Guangxi 530023, China*
[3] *College of Electronic and Electrical Engineering, Shanghai University of Engineering Sciences, Shanghai 201620, China*
[4] *College of Information Sciences and Technology, Donghua University, Shanghai 201620, China*
[5] *School of Computer Science, Engineering and Mathematics, Flinders University, Adelaide, SA 5042, Australia*

Correspondence should be addressed to Shuijie Qin; shuijie.qin@gmail.com and Dongbing Tong; tongdongbing@163.com

Received 7 February 2014; Revised 14 May 2014; Accepted 20 May 2014; Published 4 June 2014

Academic Editor: Housheng Su

The problem of adaptive asymptotical synchronization is discussed for the stochastic complex dynamical networks with time-delay and Markovian switching. By applying the stochastic analysis approach and the M-matrix method for stochastic complex networks, several sufficient conditions to ensure adaptive asymptotical synchronization for stochastic complex networks are derived. Through the adaptive feedback control techniques, some suitable parameters update laws are obtained. Simulation result is provided to substantiate the effectiveness and characteristics of the proposed approach.

1. Introduction

As is known to all, complex networks widely exist in nature, such as brain structures, protein interactions, social networks, electrical power, and World Wide Web. Recently, the dynamical behaviors of complex networks have attracted ever-increasing research interest from a variety of communities such as mathematicians, computer scientists, and control engineers. As a result, a number of dynamic analysis issues have been extensively investigated for complex networks, such as synchronization, consensus, and flocking phenomenon, in which synchronization is one of the most important and has attracted special attention of researchers in different fields [1–13]. In [4], by using Lyapunov method and some properties of Kronecker product, a sufficient condition is proposed to ensure that the dynamics of the considered network globally exponentially synchronizes with the desired solution in the mean square sense. In particular, the proposed criteria for network synchronization are in terms of linear matrix inequalities (LMIs). In [13], a modified Lyapunov-Krasovskii functional is constructed by employing the more general decomposition approach; the novel delay-dependent synchronization conditions are derived in terms of LMIs, which can be easily solved by various convex optimization algorithms.

Meanwhile, the stability and synchronization of complex networks can be applied to secure communication systems [14], information science [15], and brain science [16], and so on. The synchronization of complex networks is to achieve the accordance of the states of the drive complex network and the response complex network in a moment. That is to say, the state of the error system can achieve zero eventually when the time approaches infinity. In particular, the adaptive synchronization for a complex network is such synchronization that the parameters of the drive complex network need to be estimated and the synchronization control law needs to be updated in real time when the complex network evolves. Furthermore, the stochastic complex dynamic network contains inherent time delay, which may cause instability or oscillation.

It should be pointed out that, up to now, the problem of adaptive asymptotical synchronization for stochastic complex networks with time-delay and Markovian switching has received very little research attention.

Summarizing the above discussions, the focus of this paper is on the adaptive asymptotical synchronization problem for stochastic delayed complex networks with Markovian switching. The main purpose of this paper is to establish stability criteria for testing whether the stochastic complex network is adaptive asymptotical synchronization. By using the stochastic analysis approach and the M-matrix method, several sufficient conditions to ensure adaptive synchronization for stochastic complex networks are derived. Via the adaptive feedback technique, some suitable parameters update laws are obtained. Moreover, a simulation example is provided to show the effectiveness of the proposed controller design scheme. The main novelty of our contribution is threefold: (1) adaptive asymptotical synchronization control is addressed for stochastic complex networks with time-delay and Markovian switching; (2) using the adaptive feedback control techniques, adaptive feedback controller is designed; (3) the M-matrix method of adaptive synchronization controller is given by employing a new nonnegative function.

The organization of this paper is as follows. In Section 2, the mathematical model of the stochastic complex networks with time-delay and Markovian switching is presented and some preliminaries are given. The main results of adaptive asymptotical synchronization are proved in Section 3. In Section 4, a simple example is given to demonstrate the effectiveness of the proposed results. Finally, the conclusions are presented in Section 5.

2. Problem Formulation and Preliminaries

The coupled complex networks can be called drive complex network and described as follows:

$$\dot{x}_l(t) = f(x_l(t)) + \sum_{p=1}^{N} a_{lp}\Theta x_p(t) + \sum_{p=1}^{N} b_{lp}\Theta x_p(t-\tau(t)), \quad l = 1, 2, \ldots, N, \tag{1}$$

where $t \geq 0$, $x_l(t) = [x_{l1}(t), x_{l2}(t), \ldots, x_{lN}(t)]^T \in \mathbb{R}^N$ is the state vector of the lth node, $f(x_l(t)) \in \mathbb{R}^N$ is a nonlinear vector-valued function, $\Theta = I_n = \mathrm{diag}\{1, 1, \ldots, 1\} \in \mathbb{R}^{N\times N}$ is an inner-coupling matrix, $A = (a_{lp})^{N\times N} \in \mathbb{R}^{N\times N}$ and $B = (b_{lp})^{N\times N} \in \mathbb{R}^{N\times N}$ are the connection weight and the delayed connection weight matrices, and a_{lp} and b_{lp} are the weight or coupling strength. If there exists a link from node l to p ($l \neq p$), then $a_{lp} \neq 0$ and $b_{lp} \neq 0$. Otherwise, $a_{lp} = 0$ and $b_{lp} = 0$. $\tau(t)$ is the time-varying delay satisfying $0 < \tau(t) \leq \overline{\tau}$ and $\dot{\tau}(t) \leq \widehat{\tau} < 1$, where $\overline{\tau}$ and $\widehat{\tau}$ are constants.

Given a probability space $(\Omega, \mathcal{F}, \text{and } P)$, $\{r(t), t \geq 0\}$ is a homogeneous finite-state Markovian process with right continuous trajectories and taking values in finite set $S = \{1, 2, \ldots, N\}$ with the initial model $r(0) = r_0$. Let generator $\Gamma = (\gamma_{ij})_{N\times N}$, $i, j \in S$, be the transition rate matrix with transition probability

$$P\{r(t+\delta) = j \mid r(t) = i\} = \begin{cases} \gamma_{ij}\delta + o(\delta) & \text{if } i \neq j, \\ 1 + \gamma_{ii}\delta + o(\delta) & \text{if } i = j, \end{cases} \tag{2}$$

where $\delta > 0$ and $\gamma_{ij} \geq 0$ is the transition rate from i to j if $i \neq j$, while

$$\gamma_{ii} = -\sum_{j=1, j\neq i}^{N} \gamma_{ij}. \tag{3}$$

Let $x(t) = [x_1^T(t), x_2^T(t), \ldots, x_N^T(t)]^T \in \mathbb{R}^{N\times N}$, $f(x(t)) = [f^T(x_1(t)), f^T(x_2(t)), \ldots, f^T(x_N(t))]^T$; the drive complex network (1) with Markovian switching can be rewritten as

$$dx(t) = [f(x(t)) + A(r(t)) \otimes I_n x(t) + B(r(t)) \otimes I_n x(t-\tau(t))]\, dt, \tag{4}$$

where, for the purpose of simplicity, we denote $r(t) = i$, $A(r(t)) = A^i$, $B(r(t)) = B^i$, and $x(t-\tau(t)) = x_\tau(t)$, respectively.

For the drive complex network (4), a response complex network is constructed in the following form:

$$dy(t) = [f(y(t)) + A(r(t)) \otimes I_n y(t) + B(r(t)) \otimes I_n y_\tau(t) + U(t)]\, dt + \sigma(t, y(t) - x(t), y_\tau(t) - x_\tau(t))\, d\omega(t), \tag{5}$$

where $y(t)$ is the state vector of the response complex network (5). $U(t) = (u_1(t), u_2(t), \ldots, u_N(t))^T \in R^N$ is a control input vector with the form of

$$U(t) = K(t)(y(t) - x(t)) = \mathrm{diag}\{k_1(t), k_2(t), \ldots, k_n(t)\}(y(t) - x(t)), \tag{6}$$

$\omega(t) = (\omega_1(t), \omega_2(t), \ldots, \omega_N(t))^T$ is an N-dimensional Brown moment defined on a complete probability space $(\Omega, \mathcal{F}, \text{and } P)$ with a natural filtration $\{\mathcal{F}_t\}_{t\geq 0}$ (i.e., $\mathcal{F}_t = \sigma\{\omega(s) : 0 \leq s \leq t\}$ is a σ-algebra) and is independent of the Markovian process $\{r(t)\}_{t\geq 0}$, and σ is the noise intensity matrix and can be regarded as a result from the occurrence of eternal random fluctuation and other probabilistic causes.

Let $e_l(t) = y_l(t) - x_l(t)$ and $e(t) = [e_1^T(t), e_2^T(t), \ldots, e_N^T(t)]^T \in \mathbb{R}^{n\times N}$. As a matter of convenience, we mark $e(t-\tau(t)) = e_\tau(t)$ and $\phi(e(t)) = f(y(t)) - f(x(t))$. From the complex networks (4) and (5), the error system is arranged as

$$de(t) = [\phi(e(t)) + A(r(t)) \otimes I_n e(t) + B(r(t)) \otimes I_n e_\tau(t) + U(t)]\, dt + \sigma(t, e(t), e_\tau(t))\, d\omega(t). \tag{7}$$

The main purpose of the rest of this paper is to set up a criterion of adaptive asymptotical synchronization for the system (4)–(7) via employing adaptive control and M-matrix methods. Next, we firstly introduce assumptions, definitions, and lemmas which will be used in the proofs of main results.

Assumption 1. The activation function $f(x(t))$ satisfies the Lipschitz condition. That is to say, there exists a constant $L > 0$ such that

$$|f(u) - f(v)| \leq L|u - v|, \quad \forall u, v \in R^n. \tag{8}$$

Assumption 2. The noise intensity matrix $\sigma(\cdot,\cdot,\cdot)$ satisfies the linear growth condition. That is to say, there exist positives H_1 and H_2, such that

$$\operatorname{trace}\left(\sigma\left(t,e,e_\tau\right)\right)^T\left(\sigma\left(t,e,e_\tau\right)\right) \le H_1|e|^2 + H_2\left|e_\tau\right|^2. \quad (9)$$

Definition 3 (see [17]). The trivial solution $e(t,\zeta)$ of the error system (7) is said to be almost surely asymptotically stable if

$$P\left(\lim_{t\to\infty}\left|e\left(t;i,\zeta\right)\right| = 0\right) = 1 \quad (10)$$

for any $\zeta \in L^2_{\mathscr{F}_0}([-\tau,0];R^n)$.

The response system (5) and the drive system (4) are said to be asymptotically synchronized if the error system (7) is asymptotically stable.

Definition 4 (see [18]). Consider an n-dimensional stochastic delayed differential equation (SDDE, for short) with Markovian switching:

$$\begin{aligned} dx\left(t\right) &= \mathbb{f}\left(t,r\left(t\right),x\left(t\right),x_\tau\left(t\right)\right)dt \\ &\quad + \mathbb{g}\left(t,r\left(t\right),x\left(t\right),x_\tau\left(t\right)\right)d\omega\left(t\right) \end{aligned} \quad (11)$$

on $t \in [0,\infty)$ with the initial data given by

$$\left\{x\left(\theta\right) : -\tau \le \theta \le 0\right\} = \xi \in L^2_{\mathscr{F}_0}\left(\left[-\tau,0\right];R^n\right). \quad (12)$$

For $V \in C^{2,1}(R_+ \times S \times R^n; R_+)$, define an operator $\mathscr{L}$ from $R_+ \times S \times R^n$ to R by

$$\begin{aligned} &\mathscr{L}V\left(t,i,x\left(t\right),x_\tau\left(t\right)\right) \\ &\quad = V_t\left(t,i,x\left(t\right)\right) + V_x\left(t,i,x\left(t\right)\right)\mathbb{f}\left(t,i,x\left(t\right),x_\tau\left(t\right)\right) \\ &\qquad + \left(\frac{1}{2}\right)\operatorname{trace}\left(\mathbb{g}^T\left(t,i,x\left(t\right),x_\tau\left(t\right)\right)V_{xx}\left(t,i,x\left(t\right)\right)\right. \\ &\qquad\qquad \left.\cdot\,\mathbb{g}\left(t,i,x\left(t\right),x_\tau\right)\left(t\right)\right) \\ &\qquad + \sum_{j=1}^{N}\gamma_{ij}V\left(t,j,x\left(t\right)\right), \end{aligned} \quad (13)$$

where

$$\begin{aligned} V_t\left(t,i,x\left(t\right)\right) &= \frac{\partial V\left(t,i,x\left(t\right)\right)}{\partial t}, \\ V_x\left(t,i,x\left(t\right)\right) &= \left(\frac{\partial V\left(t,i,x\left(t\right)\right)}{\partial x_1}, \frac{\partial V\left(t,i,x\left(t\right)\right)}{\partial x_2}, \ldots, \frac{\partial V\left(t,i,x\left(t\right)\right)}{\partial x_n}\right), \\ V_{xx}\left(t,i,x\left(t\right)\right) &= \left(\frac{\partial^2 V\left(t,i,x\left(t\right)\right)}{\partial x_j \partial x_k}\right)_{n\times n}. \end{aligned} \quad (14)$$

Lemma 5 (see [18]). *Let $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^n$; then $x^T y + y^T x \le \epsilon x^T x + \epsilon^{-1} y^T y$, for any $\epsilon > 0$.*

Lemma 6 (see [18]). *If $M = (m_{ij})_{n\times n} \in R^{n\times n}$ with $m_{ij} < 0$ $(i \ne j)$, then the following statements are equivalent.*

(i) *M is a nonsingular M-matrix.*

(ii) *Every real eigenvalue of M is positive.*

(iii) *M is positive stable. That is, M^{-1} exists and $M^{-1} > 0$ (i.e., $M^{-1} \ge 0$ and at least one element of M^{-1} is positive).*

Lemma 7 (see [17]). *Assume that there are functions $V \in C^{2,1}(R_+ \times S \times R^n; R_+)$, $\psi \in L^1(R_+; R_+)$, and $w_1, w_2 \in C(R^n; R_+)$ such that*

$$\begin{aligned} \mathscr{L}V\left(t,i,x,y\right) &\le \psi\left(t\right) - w_1\left(x\right) + w_2\left(y\right), \\ &\forall\left(t,i,x,y\right) \in R_+ \times S \times R^n \times R^n, \end{aligned} \quad (15)$$

$$w_1\left(0\right) = w_2\left(0\right) = 0, \quad w_1\left(x\right) > w_2\left(x\right) \quad \forall x \ne 0,\ y \ne 0, \quad (16)$$

$$\lim_{|x|\to\infty}\inf_{0\le t<\infty, i\in S} V\left(t,i,x\right) = \infty. \quad (17)$$

Then the solution of (11) *is almost surely asymptotically stable.*

3. Main Results

In this section, some criteria of adaptive asymptotical synchronization will be obtained for the system (4), (5), and (7).

Theorem 8. *Assume that $M := -\operatorname{diag}\{\underbrace{\theta,\theta,\ldots,\theta}_{N}\} - \Gamma$ is a nonsingular M-matrix, where*

$$\begin{aligned} \theta &= 1 + L^2 + \alpha + \beta + H_1, \\ \alpha &= \lambda_{\max}\left(A^i \otimes I_n\right), \\ \beta &= \lambda_{\max}\left(\frac{1}{2}\left(B^i \otimes I_n\right)\left(B^i \otimes I_n\right)^T\right). \end{aligned} \quad (18)$$

Let $m > 0$ and $\vec{m} = (\underbrace{m,m,\ldots,m}_{N})^T$. That is to say, all elements of $M^{-1}\vec{m}$ are positive. According to Lemma 6, $(q_1,q_2,\ldots,q_N)^T := M^{-1}\vec{m} \gg 0$. In addition, assume also that

$$\left(1 + H_2\right)\overline{q} < -\left(\theta q_i + \sum_{v=1}^{N}\gamma_{iv}q_v\right), \quad \forall i \in S, \quad (19)$$

where $\overline{q} = \max_{i\in S} q_i$.

Under Assumptions 1 and 2, the response complex network (5) *can be adaptively synchronized with the drive complex network* (4)*, if the feedback gain $K(t)$ with the update law is chosen as*

$$\dot{k}_j = -\nu_j q_i e_j^2. \quad (20)$$

Proof. Choose a nonnegative function candidate as

$$V\left(t,e\right) = q_i|e|^2 + \sum_{j=1}^{n}\frac{1}{\nu_j}k_j^2. \quad (21)$$

The computation of $\mathscr{L}V(t,e)$ along with the solution of error system (7) and using (20) is

$$\begin{aligned}
&\mathscr{L}V(t,e)\\
&\quad = V_t(t,e) + V_e(t,e)\\
&\qquad \times \left[\phi(e(t)) + A^i \otimes I_n e(t) + B^i \otimes I_n e_\tau(t) + U(t)\right]\\
&\qquad + \frac{1}{2}\mathrm{trace}\left(\sigma^T(t,e,e_\tau) V_{ee}(t,e)\sigma(t,e,e_\tau)\right)\\
&\qquad + \sum_{v=1}^{N}\gamma_{iv}V(t,v,e)\\
&\quad = 2\sum_{j=1}^{n}\frac{1}{\nu_j}k_j\dot{k}_j + 2q_i e^T\\
&\qquad \times \left[\phi(e(t)) + A^i \otimes I_n e(t) + B^i \otimes I_n e_\tau(t) + Ke(t)\right]\\
&\qquad + \frac{1}{2}\mathrm{trace}\left(\sigma^T(t,e,e_\tau) V_{ee}(t,e)\sigma(t,e,e_\tau)\right)\\
&\qquad + \sum_{v=1}^{N}\gamma_{iv}q_v|e|^2\\
&\quad = 2q_i e^T\left[\phi(e(t)) + A^i \otimes I_n e(t) + B^i \otimes I_n e_\tau(t)\right]\\
&\qquad + q_i\mathrm{trace}\left(\sigma^T(t,e,e_\tau)\sigma(t,e,e_\tau)\right) + \sum_{v=1}^{N}\gamma_{iv}q_v|e|^2.
\end{aligned} \tag{22}$$

Now, according to Assumptions 1 and 2 together with Lemma 5, one obtains

$$\begin{aligned}
e^T\phi(e(t)) &\le \frac{1}{2}e^T e + \frac{1}{2}\phi^T(e)\phi(e) \le \frac{1}{2}\left(1+L^2\right)|e|^2,\\
e^T A^i \otimes I_n e &\le \alpha|e|^2,\\
e^T B^i \otimes I_n e_\tau &\le \frac{1}{2}e^T\left(B^i \otimes I_n\right)\left(B^i \otimes I_n\right)^T e + \frac{1}{2}e_\tau^T e_\tau\\
&\le \beta|e|^2 + \frac{1}{2}|e_\tau|^2,\\
\mathrm{trace}\left(\sigma^T(t,e,e_\tau)\sigma(t,e,e_\tau)\right) &\le H_1|e|^2 + H_2|e_\tau|^2.
\end{aligned} \tag{23}$$

Substituting (23) into (22), one gets

$$\begin{aligned}
&\mathscr{L}V(t,e)\\
&\quad \le 2q_i\left[\frac{1}{2}\left(1+L^2\right)|e|^2 + \alpha|e|^2 + \left(\beta|e|^2 + \frac{1}{2}|e_\tau|^2\right)\right]\\
&\qquad + q_i\left(H_1|e|^2 + H_2|e_\tau|^2\right) + \sum_{v=1}^{N}\gamma_{iv}q_v|e|^2\\
&\quad = \left(\theta q_i + \sum_{v=1}^{N}\gamma_{iv}q_v\right)|e|^2 + \left(1+H_2\right)q_i|e_\tau|^2\\
&\quad \le -m|e|^2 + \left(1+H_2\right)\overline{q}|e_\tau|^2,
\end{aligned} \tag{24}$$

where $m = -(\theta q_i + \sum_{v=1}^{N}\gamma_{iv}q_v)$ by $[q_1, q_2, \ldots, q_N]^T = M^{-1}\vec{m}$.

Let $\psi(t) = 0$, $\omega_1(e) = m|e|^2$, and $\omega_2(e_\tau) = (1+H_2)\overline{q}|e_\tau|^2$. Then inequality (24) holds such that inequality (15) holds. Consider $\omega_1(0) = 0$ and $\omega_2(0) = 0$ when $e = 0$ and $e_\tau = 0$, and inequality (19) implies $\omega_1(e) > \omega_2(e_\tau)$. So (16) holds. Moreover, (17) holds when $|e| \to \infty$ and $|e_\tau| \to \infty$. By Lemma 7, the error system (7) is adaptive almost surely asymptotically stable, and hence the noise-perturbed response complex network (5) can be adaptively almost surely asymptotically synchronized with the drive complex network (4). This completes the proof. □

Remark 9. For complex networks (1), the method in the paper can be used in some systems, such as multiagent systems [19–21] and wireless sensor networks [22], which are the next research topic for us.

Now, we are in a position to consider two cases of the complex networks (4)-(5), which have the following corollaries.

The Markovian switching is removed from the complex networks. That is to say, the drive complex network, the response complex network, and the error system can be represented, respectively, as follows:

$$\begin{aligned}
dx(t) &= \left[f(x(t)) + A \otimes I_n x(t) + B \otimes I_n x_\tau(t)\right]dt,\\
dy(t) &= \left[f\left(y(t)\right) + A \otimes I_n y(t)\right.\\
&\qquad \left. + B \otimes I_n y_\tau(t) + U(t)\right]dt\\
&\qquad + \sigma\left(t, y(t) - x(t), y_\tau(t) - x_\tau(t)\right)dw(t),\\
de(t)&\\
&= \left[\phi(e(t)) + A \otimes I_n e(t) + B \otimes I_n e_\tau(t) + U(t)\right]dt\\
&\qquad + \sigma\left(t, e(t), e_\tau(t)\right)dw(t).
\end{aligned} \tag{25}$$

For this case, one can get the following result analogous to Theorem 8.

Corollary 10. *Assume that* $M := -\mathrm{diag}\{\underbrace{\theta, \theta, \ldots, \theta}_{N}\} - \Gamma$ *is a nonsingular M-matrix, where* $\theta < 0$, $\theta = 1 + L^2 + \alpha + \beta + H_1$, *and*

$$1 + H_2 < -\theta. \tag{26}$$

Under Assumptions 1 and 2, the noise-perturbed response complex network can be adaptively asymptotically synchronized with the drive complex network, if the feedback gain $K(t)$ *of the controller* (6) *with the update law is chosen as*

$$\dot{k}_j = -\nu_j e_j^2. \tag{27}$$

Proof. Choose the following nonnegative function:

$$V(t,e) = |e|^2 + \sum_{j=1}^{n}\frac{1}{\nu_j}k_j^2. \tag{28}$$

The remaining proof is similar to that of Theorem 8 and hence omitted. □

The noise-perturbation is removed from the response complex network (5); then the response complex network and the error system can be represented, respectively, as follows:

$$\begin{aligned} dy(t) = {} & \left[f\left(y(t)\right) + A\left(r(t)\right) \otimes I_n y(t)\right. \\ & \left. + B\left(r(t)\right) \otimes I_n y_\tau(t) + U(t)\right] dt, \\ de(t) = {} & \left[\phi\left(e(t)\right) + A\left(r(t)\right) \otimes I_n e(t)\right. \\ & \left. + B \otimes I_n e_\tau(t) + U(t)\right] dt, \end{aligned} \tag{29}$$

which can lead to the following results.

Corollary 11. *Assume that* $M := -\operatorname{diag}\{\underbrace{\theta, \theta, \ldots, \theta}_{N}\} - \Gamma$ *is a nonsingular M-matrix, where* $\theta = 1 + L^2 + \alpha + \beta$ *and*

$$\overline{q} < -\left(\theta q_i + \sum_{v=1}^{N} \gamma_{iv} q_v\right), \quad \forall i \in S, \tag{30}$$

where $\overline{q} = \max_{i \in S} q_i$.

Under Assumptions 1 and 2, the noiseless-perturbed response complex network can be adaptively asymptotically synchronized with the drive complex network, if the feedback gain $K(t)$ *of the controller* (6) *with the update law is chosen as* (20).

Proof. The proof is similar to that of Theorem 8 and hence omitted. □

4. Illustrative Example

In this section, an illustrative example will be given to demonstrate the effectiveness of the proposed methods.

Example 1. The Lorenz system is described by

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} a\left(x_2(t) - x_1(t)\right) \\ b x_1(t) - x_2(t) - x_1(t) x_3(t) \\ x_1(t) x_2(t) - c x_3(t) \end{bmatrix}, \tag{31}$$

where $a = 10$, $b = 28$, and $c = 10/3$.

According to Theorem 8, the complex networks (drive complex network and response complex network) with four nodes are described as follows:

$$\begin{aligned} \dot{x}_{l1} &= a x_{l2}(t) - a x_{l1}(t) + \sum_{p=1}^{4} a_{lp} x_{p1}(t) + \sum_{p=1}^{4} b_{lp} x_{p1}(t-\tau), \\ \dot{x}_{l2} &= b x_{l1}(t) - x_{l2}(t) - x_{l1}(t) x_{l3}(t) + \sum_{p=1}^{4} a_{lp} x_{p2}(t) \\ &\quad + \sum_{p=1}^{4} b_{lp} x_{p2}(t-\tau), \\ \dot{x}_{l3} &= x_{l1}(t) x_{l2}(t) - c x_{l3}(t) + \sum_{p=1}^{4} a_{lp} x_{p3}(t) \\ &\quad + \sum_{p=1}^{4} b_{lp} x_{p3}(t-\tau), \\ \dot{y}_{l1} &= a y_{l2}(t) - a y_{l1}(t) + \sum_{p=1}^{4} a_{lp} y_{p1}(t) + \sum_{p=1}^{4} b_{lp} y_{p1}(t-\tau) \\ &\quad + k_l \left[y_{l1} - x_{l1} + \tanh\left(y_{l1}\right) - \tanh\left(x_{l1}\right)\right] \\ &\quad + 0.4\left[y_{l1}(t) - x_{l1}(t)\right], \\ \dot{y}_{l2} &= b y_{l1}(t) - y_{l2}(t) - y_{l1}(t) y_{l3}(t) + \sum_{p=1}^{4} a_{lp} y_{p2}(t) \\ &\quad + \sum_{p=1}^{4} b_{lp} y_{p2}(t-\tau) \\ &\quad + k_l \left[y_{l2} - x_{l2} + \tanh\left(y_{l2}\right) - \tanh\left(x_{l2}\right)\right] \\ &\quad + 0.3\left[y_{l2}(t) - x_{l2}(t)\right], \\ \dot{y}_{l3} &= y_{l1}(t) x_{l2}(t) - c y_{l3}(t) + \sum_{p=1}^{4} a_{lp} y_{p3}(t) \\ &\quad + \sum_{p=1}^{4} b_{lp} y_{p3}(t-\tau) \\ &\quad + k_l \left[y_{l3} - x_{l3} + \tanh\left(y_{l3}\right) - \tanh\left(x_{l3}\right)\right] \\ &\quad + 0.3\left[y_{l3}(t-\tau) - x_{l3}(t-\tau)\right]. \end{aligned} \tag{32}$$

In the simulation, let

$$A = \begin{bmatrix} -6 & 2 & 1 & 3 \\ 2 & -5 & 2 & 1 \\ 0 & 1 & -1 & 0 \\ 3 & 1 & 0 & -4 \end{bmatrix}, \quad B_1 = \begin{bmatrix} -8 & 3 & 1 & 4 \\ 3 & -5 & 0 & 2 \\ 1 & 0 & -3 & 2 \\ 1 & 3 & 2 & -6 \end{bmatrix},$$

$$B_2 = \begin{bmatrix} -2 & 1 & 1 & 0 \\ 2 & -3 & 0 & 1 \\ 1 & 0 & -2 & 1 \\ 0 & 2 & 2 & -4 \end{bmatrix}, \quad \Gamma = \begin{bmatrix} -1.2 & 1.2 \\ 0.5 & -0.5 \end{bmatrix},$$

$$\tau = 0.1. \tag{33}$$

These parameters fully satisfy Assumptions 1 and 2 and condition (19). Therefore, it will prove the main result to be correct if the error system can be adaptively asymptotically synchronized satisfying Theorem 8.

To illustrate the effectiveness of the developed theory, we employ the nonnegative function to solve the solutions for stochastic complex networks and to simulate the dynamics of error system and the adaptive feedback gain. The simulation figures are shown in Figures 1, 2, 3, and 4.

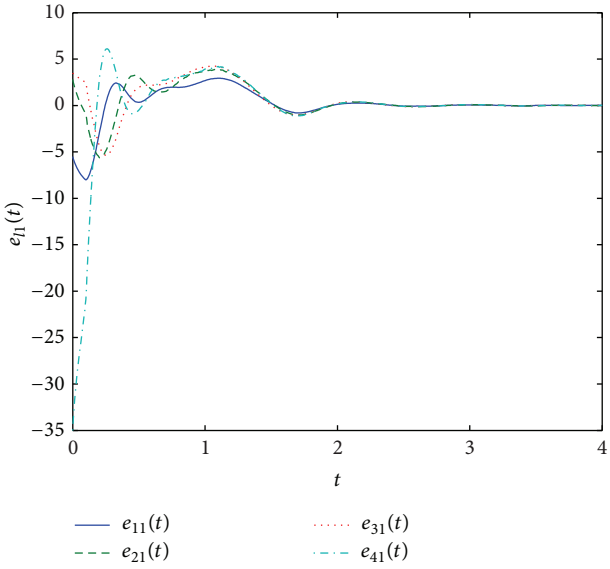


Figure 1: The error states of complex network e_{l1}.

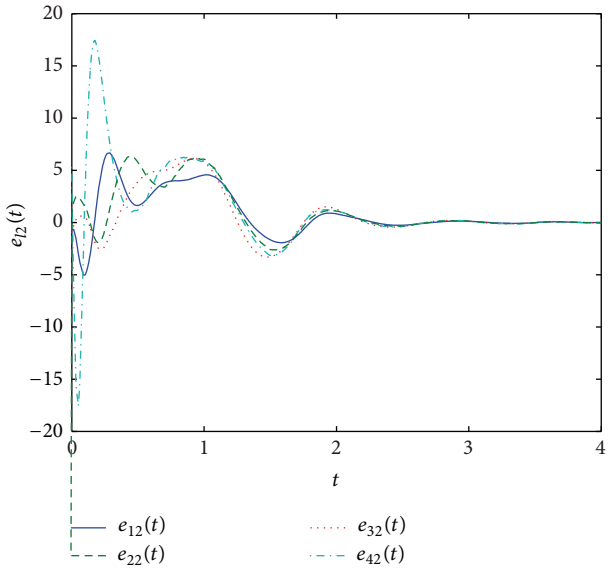


Figure 2: The error states of complex network e_{l2}.

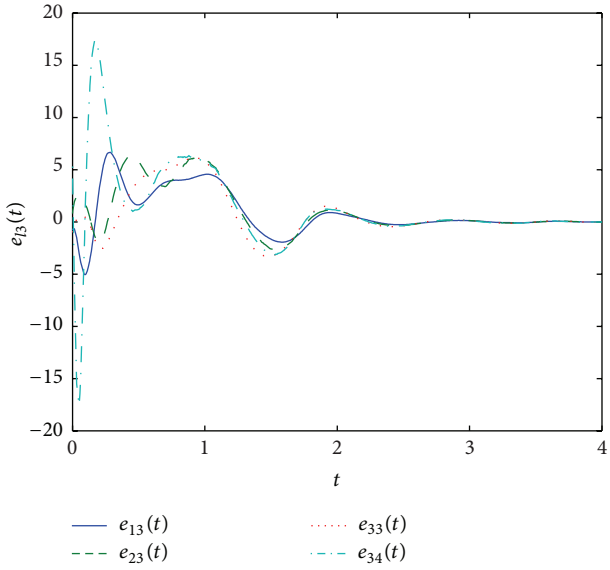


Figure 3: The error states of complex network e_{l3}.

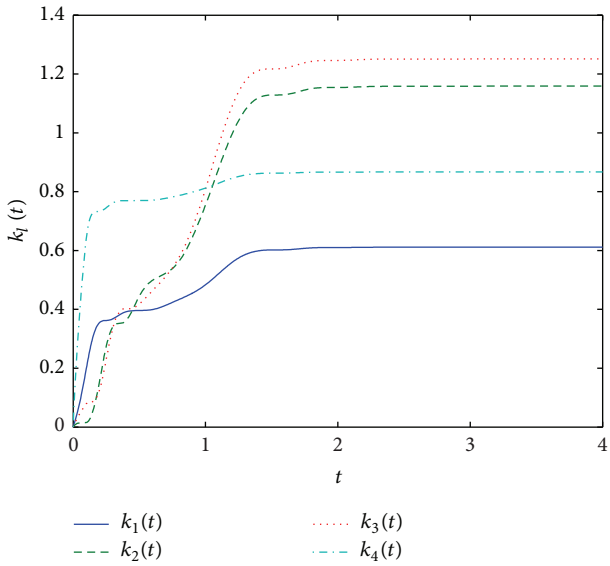


Figure 4: The feedback gain.

Among them, Figures 1–3 plot the error states of complex networks $e_{l1}(t)$, $e_{l2}(t)$, and $e_{l3}(t)$. Figure 4 depicts the adaptive feedback gain. From all these figures, one can find that the stochastic complex networks are adaptively asymptotically synchronized.

5. Conclusions

In this paper, we have investigated the adaptive synchronization problem for the stochastic complex networks with time-delay and Markovian switching. By combining the Lyapunov functional, stochastic analysis method, and M-matrix approach, some sufficient conditions have derived the above adaptive synchronization for the stochastic delayed complex networks. Through the adaptive control techniques, some suitable parameters update laws are obtained. Finally, an illustrative example has been used to demonstrate the effectiveness of the main results which are obtained in this paper.

Conflict of Interests

The authors declared that they have no conflict of interests to this work.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61363074), the Scientific Research Fund of Guangxi Education Department of China, the Natural Science Foundation of Guizhou Province of China ((2009)2219), the Natural Science Foundation of Guangxi Province of China (2013GXNSFAA019346), and the Scientific Research Fund of Guangxi Education Department of China (2007, 2013YB148).

References

[1] W. Zhou, D. Tong, Y. Gao, C. Ji, and H. Su, "Mode and delay-dependent adaptive exponential synchronization in pth moment for stochastic delayed neural networks with markovian switching," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 4, pp. 662–668, 2012.

[2] H. Su, Z. Rong, M. Z. 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.

[3] Y. Gao, W. Zhou, C. Ji, D. Tong, and J. Fang, "Globally exponential stability of stochastic neutral-type delayed neural networks with impulsive perturbations and Markovian switching," *Nonlinear Dynamics*, vol. 70, no. 3, pp. 2107–2116, 2012.

[4] Y. Zhang, D.-W. Gu, and S. Xu, "Global exponential adaptive synchronization of complex dynamical networks with neutral-type neural network nodes and stochastic disturbances," *IEEE Transactions on Circuits and Systems*, vol. 60, no. 10, pp. 2709–2718, 2013.

[5] D. Tong, Q. Zhu, W. Zhou, Y. Xu, and J. Fang, "Adaptive synchronization for stochastic T-S fuzzy neural networks with time-delay and Markovian jumping parameters," *Neurocomputing*, vol. 117, pp. 91–97, 2013.

[6] W. Zhou, A. Dai, D. Tong, and J. Yang, "Exponential synchronization of stochastic complex dynamical networks with impulsive perturbations and Markovian switching," *Mathematical Problems in Engineering*, vol. 2014, Article ID 927858, 10 pages, 2014.

[7] J. Lu, J. Kurths, J. Cao, N. Mahdavi, and C. Huang, "Synchronization control for nonlinear stochastic dynamical networks: pinning impulsive strategy," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 2, pp. 285–292, 2012.

[8] Y. Xu, H. Yang, D. Tong, and Y. Wang, "Adaptive exponential synchronization in pth moment for stochastic time varying multi-delayed complex networks," *Nonlinear Dynamics*, vol. 73, no. 3, pp. 1423–1431, 2013.

[9] Q. Zhu, W. Zhou, D. Tong, and J. Fang, "Adaptive synchronization for stochastic neural networks of neutral-type with mixed time-delays," *Neurocomputing*, vol. 99, pp. 477–485, 2013.

[10] Z.-G. Wu, P. Shi, H. Su, and J. Chu, "Exponential synchronization of neural networks with discrete and distributed delays under time-varying sampling," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 9, pp. 1368–1376, 2012.

[11] W. Zhou, Y. Gao, D. Tong, C. Ji, and J. Fang, "Adaptive exponential synchronization in pth moment of neutraltype neural networks with time delays and markovian switching," *International Journal of Control, Automation and Systems*, vol. 11, no. 4, pp. 845–851, 2013.

[12] Y. Xu, C. Xie, and D. Tong, "Adaptive synchronization for dynamical networks of neutral type with time-delay," *Optik*, vol. 125, no. 1, pp. 380–385, 2014.

[13] H. Li, Z. Ning, Y. Yin, and Y. Tang, "Synchronization and state estimation for singular complex dynamical networks with time-varying delays," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 1, pp. 194–208, 2013.

[14] J. Huang and A. L. Swindlehurst, "Cooperative jamming for secure communications in MIMO relay networks," *IEEE Transactions on Signal Processing*, vol. 59, no. 10, pp. 4871–4884, 2011.

[15] A. Hernando, D. Villuendas, C. Vesperinas, M. Abad, and A. Plastino, "Unravelling the size distribution of social groups with information theory in complex networks," *European Physical Journal B*, vol. 76, no. 1, pp. 87–97, 2010.

[16] E. Bullmore and O. Sporns, "Complex brain networks: graph theoretical analysis of structural and functional systems," *Nature Reviews Neuroscience*, vol. 10, no. 3, pp. 186–198, 2009.

[17] C. Yuan and X. Mao, "Robust stability and controllability of stochastic differential delay equations with Markovian switching," *Automatica*, vol. 40, no. 3, pp. 343–354, 2004.

[18] C. Yuan and X. Mao, "Asymptotic stability in distribution of stochastic differential equations with Markovian switching," *Stochastic Processes and their Applications*, vol. 103, no. 2, pp. 277–291, 2003.

[19] Y. Zhao, Z. Duan, G. Wen, and G. Chen, "Robust consensus tracking of multi-agent systems with uncertain Lur'e-type nonlinear dynamics," *IET Control Theory and Applications*, vol. 7, no. 9, pp. 1249–1260, 2013.

[20] Y. Zhao, Z. Li, and Z. Duan, "Distributed consensus tracking of multi-agent systems with nonlinear dynamics under a reference leader," *International Journal of Control*, vol. 86, no. 10, pp. 1859–1869, 2013.

[21] H. Su, G. Chen, X. Wang, and Z. Lin, "Adaptive second-order consensus of networked mobile agents with nonlinear dynamics," *Automatica*, vol. 47, no. 2, pp. 368–375, 2011.

[22] J. Mou, W. Zhou, T. Wang, C. Ji, and D. Tong, "Consensus of the distributed varying scale wireless sensor networks," *Mathematical Problems in Engineering*, vol. 2013, Article ID 862518, 9 pages, 2013.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 492753, 15 pages
http://dx.doi.org/10.1155/2014/492753

Research Article

A Dynamic Microblog Network and Information Dissemination in "@" Mode

Mingsheng Tang, Xinjun Mao, Shuqiang Yang, and Huiping Zhou

College of Computer, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Mingsheng Tang; tms110145@gmail.com

Received 13 February 2014; Revised 15 April 2014; Accepted 19 April 2014; Published 3 June 2014

Academic Editor: Guoqiang Hu

Social media, especially the microblogs, emerge as a part of our daily life and become a key way to information spread. Thus, information dissemination in the microblog became a research hotspot. Based on some principles that are summarized from the microblog users' behaviors, this paper proposes a dynamic microblog network model. Through simulations this network has the features of periodicity of average degree, high clustering coefficient, high degree of modularity, and community. Besides, an information dissemination model through "@" in the microblog has been presented. With the microblog network model and the zombie-city model, this paper has modelled an artificial microblog and has simulated the information dissemination in the artificial microblog with different scenes. Therefore, some interesting findings have been presented. (1) Due to a better connectivity, information could spread widely in a random network; (2) information spreads more quickly in a stable microblog network; (3) the decay rate of the relationships will have an effect on information dissemination; that is, with a lower decay rate, information spreads more quickly and widely; (4) the higher active level of users in microblog could promote information spread widely and quickly; (5) the "@" mode of information dissemination makes a high modularity of the information diffusion network.

1. Introduction

With the development of information and communication technology, social media are emerging. Social media become an important channel for information spread, especially microblogs. For example, in China, there are around 618,000,000 Internet users and the number of the microblog users is larger than 281,000,000 [1]. People could receive real-time information from the microblog, which is a novel and quicker way than traditional modes like newspaper. The microblog is changing the mode of information dissemination and even has an important impact on the real society. As one knows, microblog platforms such as Twitter have played an important role in Arab Spring [2, 3], and people could transmit and get the crucial information based on the microblog, which has promoted revolutionary information spread and social progress. Information spreads in microblog platforms through several ways such as posting microblogs, retweeting microblogs, and comments on microblogs. In addition, a user could use the "@" mode to "@" his/her followers (who follow this user) or followings (who are followed by this user), when he/she posts, retweets, or comments in the microblogs. This "@" mode could improve information dissemination in microblog platforms. Thus, studying the information dissemination through the "@" mode is a meaningful and interesting work.

The motivation of this paper is to research into the information dissemination in a microblog through "@" mode, including the effectiveness of information spread and factors affecting information spread. However, the microblog is a mapping of the real society, and the changes and evolutions of the microblog may have effects on the real society. Experiments on the microblog also result in negative impacts on the real society, and these experiments are also unrepeatable. Due the high cost and nonrepeatability of experiments in the real microblog platforms, we should borrow the idea of social computing and artificial society to research into information dissemination and the corresponding issues in the microblog. As the social network of the microblog plays a key role in information dissemination and is also an important part of artificial societies, firstly we should study how a microblog network dynamically evolves. Through the analysis of the

microblog users' behaviors, we could acquire some principles of the microblog network: (1) people have different limited abilities to follow others (different maximum out-degree); (2) mutual following or connecting; (3) people prefer to follow others with larger followers; (4) transitivity—people prefer to follow their followings' followings; (5) following relationships will decay with time. Based upon these characteristics, we could propose a microblog network model to reconstruct such a microblog network in the computer world. Moreover, the information dissemination model will affect the information spread, and we should also study on how information spreads. Some modes could aid in information dissemination in the microblog, and the "@" mode is an important mode for information diffusion in the microblog, which means that information is transmitted from user A to user B when user A "@" user B. Hence, the "@" mode is our focus, and we propose a model of information dissemination in "@" mode. In order to research into information spread in "@" mode, we need a general model to construct an artificial society, including users, social network, and the environment. Recently, there are some artificial society models, such as the classical artificial society model—sugarscape [4]. These models could not intuitively aid to construct artificial societies. Zombie-city model [5–7] is a new artificial society model and more suitable for the issues like infectious diseases spread and information dissemination. Based on the proposed microblog network and the information dissemination model, we could construct an artificial microblog with the zombie-city model, and then study the information dissemination in the microblog with different scenes. In this case study, some valuable and interesting results and findings could be seen.

Main contributions of this paper are as follows: (1) proposing a model for the dynamical microblog network based on the analysis of characteristics of the real microblog network; (2) aiming at the "@" mode in microblogs and borrowing the existing model for infectious disease spread, proposing the information dissemination model in "@" mode; (3) borrowing social computing and artificial society, acquiring some meaningful findings through experimental simulations in different scenes.

The remaining sections of this paper are organized as follows. Section 2 analyzes the characteristics of the real microblog networks and proposes a dynamic microblog network model based on the features of the microblog users' behaviors. Section 3 presents the information dissemination model through "@" mode in the microblog. Based on the zombie-city model, Section 4 makes some experiments for the information dissemination in an artificial microblog, and then experimental simulations and analysis show some interesting results. Section 5 summarizes the work of this paper and presents some conclusions and looks forward to the future works.

2. Dynamic Microblog Network

Like the friendship network (JGN) [8, 9] in the real society, the microblog network is dynamically growing. Based on three simple principles, we could acquire models of the friendship network and this network is an undirected network. A social network model is presented based on the analysis of microblog users' behaviors [10], which also shows the distributions of out-degree of the microblog network presenting power-law characters and the microblog has a high average clustering coefficient. As we know, the microblog network is a directed graph and the JGN network is not suitable for the microblog network because of an undirected graph. The microblog network is a directed network and is different from the friendship network. Therefore, we should summarize some characteristics about the real microblog network to reconstruct this network in computer world.

2.1. Characteristics of Microblog Networks. An increasing number of people use microblog platform to broadcast their news and express their mood. There are many connections between these users, that is, following relationships. When user A wants to get news from another user, B, then user A could follow user B in the microblog and user A could receive the microblog of user B in real time. The relationship from user A to user B is a following relationship that is directed. Thus the microblog users and their relationships could form a social network, which is similar to the friend social network (JGN). However, the microblog network is a directed graph; that is, the links or connections between users are directed. As for the dynamical behaviors of users, the microblog network is a complex and growing network and users dynamically follow other users or stop to follow other users. There are some characteristics about the dynamical microblog network. As we know, there will be an increasing number of users in microblog platform, but in a short time we could consider that the number of users is stable; that is, the population dynamics will not be mentioned.

(1) Different Abilities of Users. Each user has his/her own abilities or power (one user could not follow a great number of users), and abilities of users are distinct. As we know, fewer people have powerful abilities; that is, fewer users follow a large number of users and more users follow a small number of users. Different abilities of users bring the different max out-degree of users. The distribution of max out-degree may follow power-law distribution with the scaling feature [11].

(2) Mutual Following. A microblog user is inclined to follow the users who have followed this microblog user. As shown in Figure 1, user B has followed user A, and then it is more possible that user A may follow user B. Due to the mutual following, one user in a microblog platform may follow several fans or followers of this user.

(3) Transitivity of Following Relationships. Users may be interested in whom their followings follow, and these users more possibly want to follow their followings' followings. These following relationships have the characteristic of transitivity. As seen in Figure 2, user C is a following of user A and user B is a following of user C. Hence, user A is inclined to follow the users who are followed by user C; that is, user A will more probably follow user B. This situation is similar to the scene

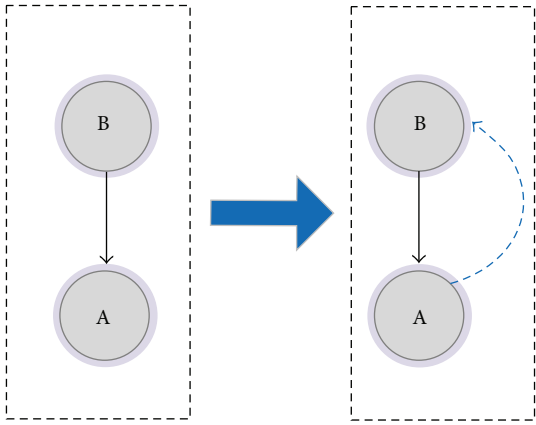


Figure 1: Mutual following.

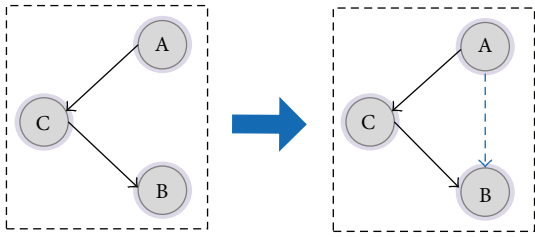


Figure 2: Transitivity of following relationships.

in real society, and people more possibly make friends with their friends' friends.

(4) Preferential Following (Matthew Effect). Usually, Matthew effect is used to describe the phenomenon that the rich will get richer and the poor will become poorer. Similarly, users in a microblog with a large number of followers will have more and more followers due to Mattew effect. As shown in Figure 3, users may more probably follow the users who have been followed by many users, which will make some users become the star users who have a large amount of fans, for example, the president of USA, Obama, in Twitter.

(5) Decay of Following Relationships. Like friendships in the real society, following relationships may decay with time. Because interests of users are developing and changing, some users may not be interested in their followings any more, and then users may cancel the following relationships. The stability of a microblog network connects with the decay rate of these following relationships. When the decay rate is higher, the stability will be worse and vice versa.

2.2. Mathematical Model of a Dynamic Microblog Network. Based on the above characteristics, whether node i connects to node j depends on the following factors: (1) the out-degree of node i; (2) whether node j has connected to node i; (3) the in-degree of node j; (4) the number of nodes, which are the out-neighbors of node i and also are the in-neighbors of

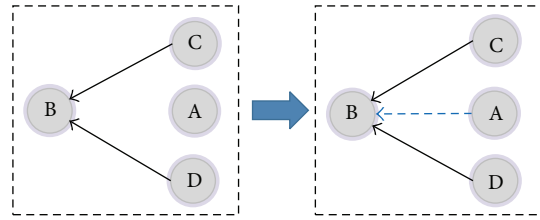


Figure 3: Preferential following (Matthew effect).

node j. Hence, the probability of node i connecting to node j could be expressed by the math equation

$$p_{ij} = f_1\left(z_{i\rightarrow}\right) f_2\left(\text{link}_{ji}\right) f_3\left(z_{j\leftarrow}\right) g\left(m\right). \tag{1}$$

The function $f_1(z_{i\rightarrow})$ is larger for small $z_{i\rightarrow}$ (the out-degree of node i) and decreases sharply around the transition value $\text{ownz}^*_{i\rightarrow}$. The distribution of $\text{ownz}^*_{i\rightarrow}$ follows the power-law exponential distribution ($P(\kappa) \propto e^{-\kappa/\mu}$). This represents that each node has a limit ability to connect to another node and their abilities are different. One possible formation of this function is as follows:

$$f_1\left(z_{i\rightarrow}\right) = \frac{1}{e^{\beta\left(z_{i\rightarrow} - \text{ownz}^*_{i\rightarrow}\right)} + 1}. \tag{2}$$

The function $f_2(\text{link}_{ji})$ is presumably large for $\text{link}_{ji} = 1$ (that means node j has connected to node i) and then falls off sharply with $\text{link}_{ji} = 0$. This means that a node will more probably connect to the node that has connected to itself. The possible functional form can be described as

$$f_2\left(\text{link}_{ji}\right) = 1 - \frac{p_0}{e^{\varepsilon \text{link}_{ji}}}. \tag{3}$$

The function $f_3(z_{j\leftarrow})$ is small for small $z_{i\leftarrow}$ (the in-degree of node j) and increases with the value $z_{i\leftarrow}$ heightening. This presents that the nodes, in which a larger number of nodes are connecting to, will attract more nodes to connect to themselves. The possible function can be formed as

$$f_3\left(z_{j\leftarrow}\right) = \frac{1}{1 + e^{-\lambda z_{j\leftarrow}}}. \tag{4}$$

The function $g(m)$ increases with the value m growing. The value m represents the number of nodes in which node i has connected to and has connected to node j. The functional form could be depicted as

$$g\left(m\right) = 1 - \left(1 - p_1\right) e^{-\alpha m}. \tag{5}$$

Above these situations, there is another situation that microblog users may cancel the following relationships. We could give the strength of this following relationship, and s_{ij} represents the strength of following relationship from user i to j. If node i begins to follow node j, then s_{ij} is set to 1. If node i has not connected to node j, then $s_{ij} = 0$. As time passes, the strength decreases exponentially as $s_{ij} = e^{-\kappa \Delta t}$. When s_{ij} is less than a threshold, node i will cancel the connection to node j. At the moment, the active level of Sina (Chinese largest microblog platform) users is around 0.7, so we could set the threshold to 0.3. If $s_{ij} < 0.3$, then node i will cancel the connection to node j.

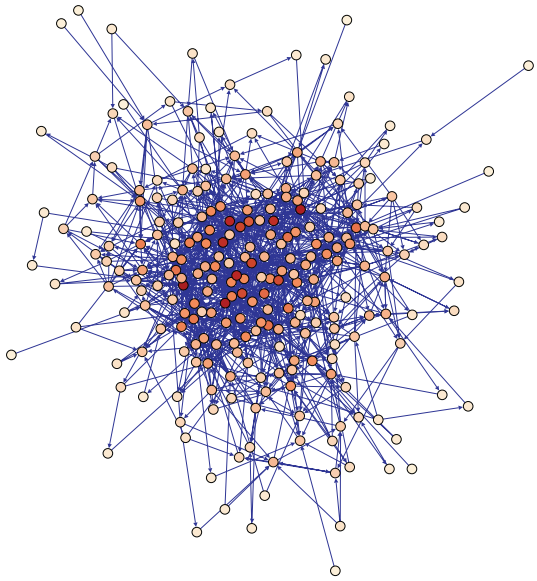


Figure 4: A snapshot of the growing microblog network ($N = 250$, $\beta = 10$, $p_0 = 0.9$, $\varepsilon = 100$, $\lambda = 0.05$, $p_1 = 0.005$, $\alpha = 10$, and $\kappa = 0.001$).

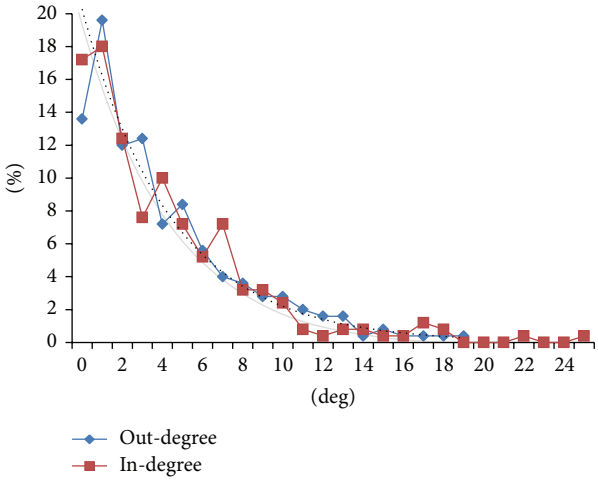


Figure 5: A snapshot of degree distribution.

2.3. Features. We could study the features and behaviors of the dynamical microblog network model through experimental simulations. In our simulations, we have initialized these simulations with an empty network (i.e., no edges or links). In the first simulation, we have set $N = 250$, $\beta = 10$, $p_0 = 0.9$, $\varepsilon = 100$, $\lambda = 0.05$, $p_1 = 0.005$, $\alpha = 10$, $\kappa = 0.001$, $\mu = 5$, and threshold (the strength threshold of connections) = 0.3, and Figure 4 presents a snapshot of this growing microblog network.

2.3.1. Degree Distribution. Since the growing microblog network is dynamic, the degree distribution is also mutative. Figure 5 shows a snapshot of degree distribution of this growing microblog network, including out-degree distribution and in-degree distribution. x-axis represents the degrees, and y-axis expresses the percentage of the vertexes with the corresponding degree. It represents in-degree with more flexibility for no limits for in-degree of a node or vertex. However, for the out-degree of each vertex or node, there is a limit or maximum out-degree, and the distribution of these maximum out-degrees should follow the power-law exponential distribution. As seen in the figure, we could fit the out-degree distribution curve with an exponential fitting curve.

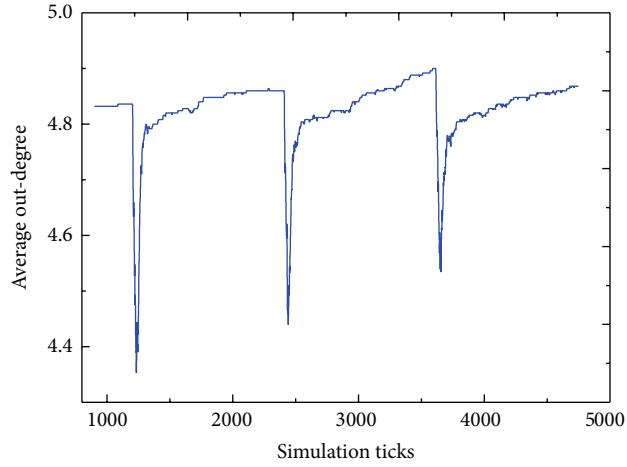


Figure 6: Average out-degree of the microblog network.

2.3.2. Average Degree. The microblog network has periodical behaviors that users could periodically release or broadcast their microblogs and could periodically skim these microblogs of their following users [12]. Figure 6 displays the average degree, and we could see that the fluctuations periodically appear with tranquillizations between these fluctuations. The periodical decay of connections results in these periodical fluctuations. With $\beta = 10$, out-degree of each vertex strictly complies with each out-degree limit. While decreasing the value of β with other parameters being fixed, the average degree will be larger.

2.3.3. Clustering Coefficient. For the clustering behavior of the dynamic microblog network, clustering coefficient is a directed way and method. For undirected networks, the clustering coefficient [13] of vertex i could be defined as follows, where a_{ij} means node i links with node j and d_i denotes the degree of node i:

$$C_i = \frac{\sum_{j,k} a_{ij} a_{jk} a_{ik}}{\sum_{j,k} a_{ij} a_{ik}} = \frac{\sum_{j,k} a_{ij} a_{jk} a_{ik}}{d_i (d_i - 1)/2}. \quad (6)$$

However, for direct networks, there are more ways to form a triangle and it is more complex to calculate the clustering coefficient [14]. For triangles, we could divide these triangles into four types, as depicted in Figure 7. The global clustering

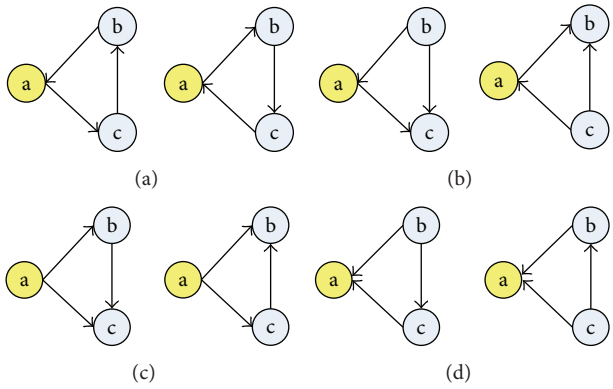


Figure 7: Potential triangles of the directed network.

coefficients $C(C(\text{loop}), C(\text{in\_out}), C(\text{all\_out}), C(\text{all\_in}))$ could be defined as follows:

$$C_i(\text{loop}) = \frac{\sum_{j\neq k} a_{ij}a_{jk}a_{ki}}{d_i(\text{in})\, d_i(\text{out}) - \text{pairs}_i},$$

$$C_i(\text{in\_out}) = \frac{\sum_{j\neq k} a_{ij}a_{kj}a_{ki}}{d_i(\text{in})\, d_i(\text{out}) - \sum_j a_{ij}a_{ji}},$$

$$C_i(\text{all\_out}) = \frac{\sum_{j\neq k} \left(a_{ij}a_{ik}a_{kj+}a_{ij}a_{ik}a_{jk}\right)}{d_i(\text{out})\left(d_i(\text{out}) - 1\right)}, \tag{7}$$

$$C_i(\text{all\_in}) = \frac{\sum_{j\neq k} \left(a_{ji}a_{ki}a_{kj+}a_{ji}a_{ki}a_{jk}\right)}{d_i(\text{in})\left(d_i(\text{in}) - 1\right)}.$$

As seen in (7), each subclustering coefficient has been defined and $d_i(\text{in})$ represents the in-degree of vertex i and $d_i(\text{out})$ expresses the out-degree of vertex i. Then, the global clustering coefficient could be depicted as

$$C = \frac{\sum_{i=1}^{N} \left(C_i(\text{loop}), C_i(\text{in\_out}), C_i(\text{all\_out}), C_i(\text{all\_in})\right)}{N}, \tag{8}$$

with $N = 250, \beta = 10, p_0 = 0.9, \varepsilon = 100, \lambda = 0.05, p_1 = 0.005, \alpha = 10, \kappa = 0.001, \mu = 5$, and threshold = 0.3. Figure 8 shows the global coefficient. All these global subclustering coefficients are higher than clustering coefficients of the corresponding random network (especially the value of $C(\text{all\_out})$), and most subclustering of the random network is less than 0.04. The periodical behaviors of microblog network could also be observed through the periodicity of the global clustering coefficient. As seen in Figure 8, there are periodical fluctuations and between these fluctuations the clustering coefficients are stable. In this simulation, the $C(\text{all\_out})$ is greater than other subclustering coefficients, and it could be mapped in the real microblog network. Most users of microblog will follow the stars (the users are stars or presidents) and one user may follow several stars, and these stars may not follow the general users and they may be apt to follow other stars. So this situation causes greater $C(\text{all\_out})$. Through adjusting these parameters, we could get different global clustering coefficients.

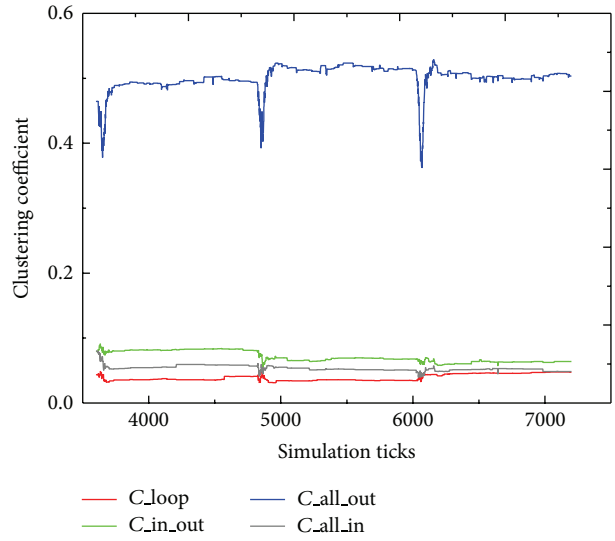


Figure 8: Global clustering coefficient of the dynamic microblog network ($N = 250, \beta = 10, p_0 = 0.9, \varepsilon = 100, \lambda = 0.05, p_1 = 0.005, \alpha = 10$, and $\kappa = 0.001$).

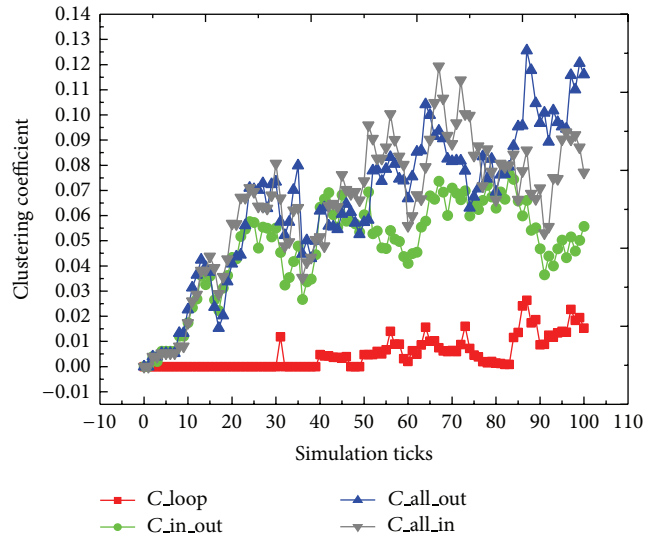


Figure 9: Global clustering coefficient with $\kappa = 0.1$.

Through changing the parameter κ to 0.1 ($\kappa = 0.1$) and fixing other parameters, the clustering coefficients sharply fluctuate with time, as shown in Figure 9. Adjusting parameter κ with a larger value, the clustering coefficients will rise and fall more sharply. Meanwhile, the average degree will also fluctuate sharply. There are many parameters in this dynamic microblog network model. Different β could give each node with different flexibilities about its out-degree. Less p_0 and ε will bring more mutual connections between vertexes. Larger λ will easily bring Matthew effect. Then, through adjusting the parameter α, we could control the transitivity of the network and bigger α could cause higher transitivity.

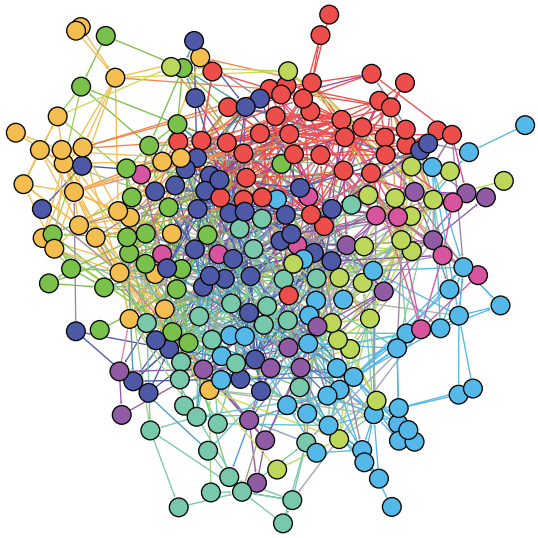


Figure 10: Communities with different colors (splitted by modularity classes).

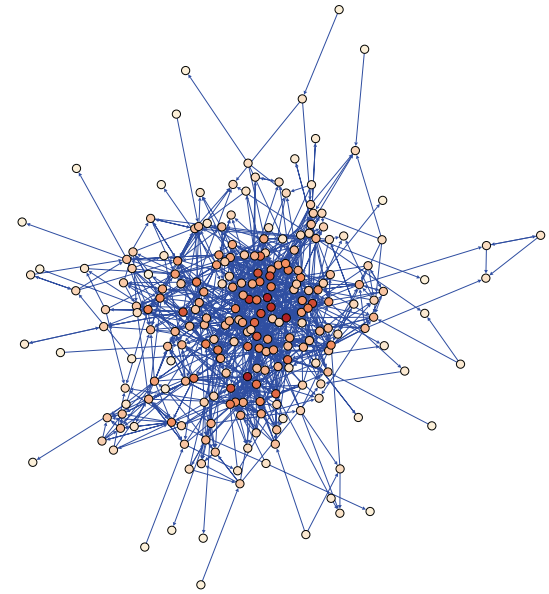


Figure 11: A snapshot of the dynamic microblog network ($N = 250$, $\mu = 5$, $r_0 = 0.0015$, $r_1 = 0.9$, $r_2 = 0.001$, $r_3 = 2$, $\gamma = 0.001$).

2.3.4. Community. Another measure for the structure of complex networks is the modularity [15, 16], which measures the strength of dividing a network into modules, groups, clusters, or communities. For the microblog networks, a high modularity denotes that the network has dense connections between the users within the same communities and sparse connections between users in different communities. Modularity of the network in Figure 4 is about 0.401, which is a high degree of modularity. Based on the modularity classes, this network could be divided into 9 communities, as shown in Figure 10. Nodes with different colors represent nodes in different communities. Compared with a random social network, with the same average degree, the modularity of the random network is about 0.28. The mechanism of dynamic microblog network results in a higher modularity.

2.4. Another Model for the Microblogging Network. In order to facilitate simulating such a dynamic microblog network, we provide another way to construct this microblog network. As seen in the former model, we could know that in each time the network will create new edges following specific rules and periodically cancel several edges. So we could propose a novel model equivalent with the former one; the algorithm is as follows.

Let N denote the number of nodes in the microblog social network, the maximum out-degree (signed as n_p) of the whole social network is $N(N-1)$, and existed out-degree number is $n_e = (1/2)\sum Z_{i\rightarrow}$ ($z_{i\rightarrow}$ is the out-degree of node i). The number of neighborhood links is $n_m = (1/2)\sum Z_{i\rightarrow}(Z_{i\rightarrow}-1)$. Then, $\text{ownz}^*_{i\rightarrow}$ is the max out-degree limit of node i, and maximum out-degree of all nodes follows a power law as the exponential distribution ($P(\kappa) \propto e^{-\kappa/\mu}$). Consider the following.

(a) In each step, randomly choose $n_p r_0$ pairs of nodes. For each pair of nodes, randomly choose one of these pairs of nodes signed as node i. If out-degree of node i is less than $\text{ownz}^*_{i\rightarrow}$, then node i will connect to the other one.

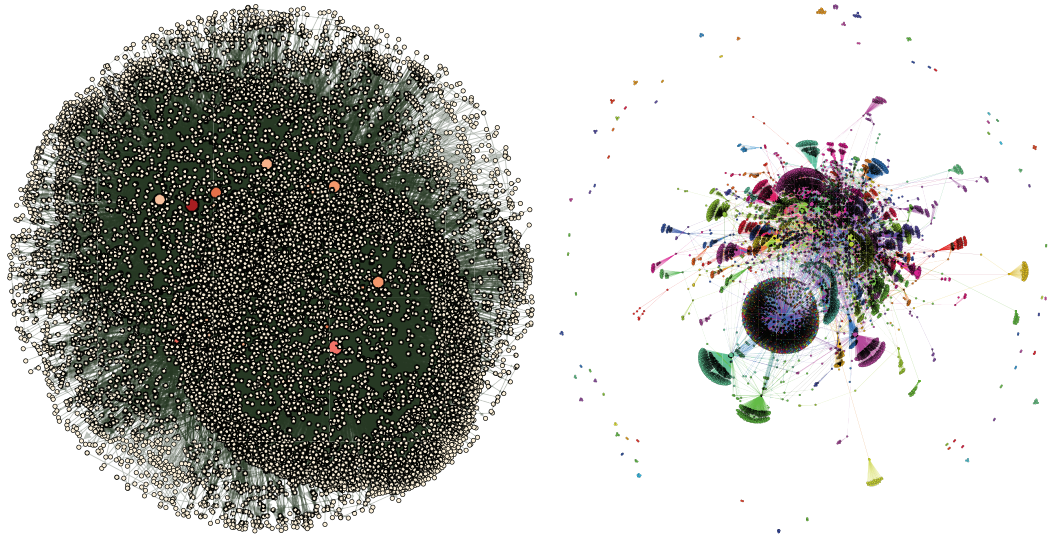
(b) In each step, randomly choose $n_p r_1$ pairs of nodes. If one of the chosen nodes (node j) has connected to the other (node i), and node i has not connected to node j and out-degree of the node i is less than $\text{ownz}^*_{i\rightarrow}$, then node i will connect to node j.

(c) In each step, randomly choose $n_p r_2$ pairs of nodes. For each pair of nodes, if this chosen node i with the minimum in-degree has not connected to the other node and the minimum in-degree of these two nodes is less than $\text{ownz}^*_{i\rightarrow}$, then node i will connect to the other one.

(d) In each step, proportionate to $z_{i\rightarrow}$ $(z_{i\rightarrow}-1)$, randomly choose $n_m r_3$ nodes. For each node, randomly choose one of nodes from its in-neighbor nodes which are connected to this node as node i, and randomly choose one of nodes from its out-neighbor nodes signed as node j. If node i has not connected to node j and out-degree of node i is less than $\text{ownz}^*_{i\rightarrow}$, then node i will connect to node j.

(e) In each step, with probability proportionate to $z_{i\rightarrow}$, randomly choose $n_e\gamma$ nodes (γ is a constant). For each node, randomly choose one of out-neighbor nodes and cancel the connection from this node to the out-neighbor node.

Most behaviors and features shown in the former model could also be reproduced with appropriate values of these parameters in this method, including the periodical behaviors and community. Figure 11 presents a snapshot of microblog network with this model.



(a) A part of "@" network in Sina Weibo (b) The "@" network after splitting with communities

Figure 12: A part of the "@" network in Sina Weibo (including 11,073 nodes and 13,046 links).

3. Information Dissemination Model in "@" Mode

Social media, especially the microblogs, have become an important channel for disseminating information, and with the social network model of microblog in Section 2 we could study several issues in the computer system, including the issue of information dissemination. Users of a microblog (like Sina Weibo) could transmit information mainly through the following methods or ways: posting on a microblog, retweeting on a microblog, and commenting on a microblog. If one user has posted, retweeted, or commented on a microblog, all of his/her fans (followers) will see this microblog in their man-machine interfaces of the microblogging platform. However, these fans may not pay attention to this microblog and miss the information in this microblog. Another mode could provide a stronger way to disseminate the information, called as "@" mode, and users could make some or all of their followers or followings know the information through posting, retweeting, or commenting on a microblog using "@". For example, when user A posts, retweets, or comments on a microblog with "@" user B, user B will prior read this microblog. The "@" mode is a more effective way to disseminate information. In order to better understand this mode, we have collected several microblog data especially the "@" relationships between users from the Sina with the open APIs released by Sina, and Figure 12(a) shows a snapshot of the "@" network in Sina, including 11,073 nodes and 13,046 links. The nodes represent the users in Sina Weibo, and the links denote the "@" relationships between users, which are directed edges; for example, the directed link from user A to user B means that user A already "@" user B. Figure 12(b) presents the feature of community and clustering of the "@" network, where modularity is 0.848, modularity with resolution is 0.848, and number of communities is 2,896.

The "@" mode is a stronger, more effective, and important way for information spread; thus the "@" mode is our focus in this paper. As one knows, the social network plays a key role in information dissemination in the microblog [17]. In order to research into information dissemination, some diffusion mathematical models [18–20] have been proposed, and some infectious disease spread models [21–24] could be applied in information diffusion. These works could not directly be used to research into information dissemination in the microblog, for the specificity of "@" mode. As shown in Figure 13, there are three situations that user A could "@" user B: (a) A is a follower of B; (b) B is a follower of A; (c) A is a follower of B and B is also a follower of A. If the user C is not the follower of A and A is not a follower of C, then A cannot send any information to user C through "@" mode. A user prefers "@" the users who are followers or followings of this user. Therefore, three parameters could be defined to distinguish the probability of user A "@" user B in these three situations, as shown in Figure 13. Parameters ρ, ς, and σ denote the probability of user A "@" user B in the situations (a), (b), and (c), respectively.

When a user wants to diffuse information in "@" mode, he/she will choose some users from his/her followers and followings to retweet a microblog with "@." The process of information dissemination through "@" mode in microblog is similar to the *SIR* model of disease propagation [21–24]. Classic disease propagation models are based upon the basic assumptions that initially a person is susceptible (S) to the disease and could become a host of the infectious disease. If he/she is exposed to the disease by an infectious contact, then the person will become infected (I) with a probability; with several medical measures or his/her immunity, he/she will become recovered (R). Information spread in microblog is similar to this *SIR* model. However, the kernel meaning of these three states (*SIR*) of information dissemination in

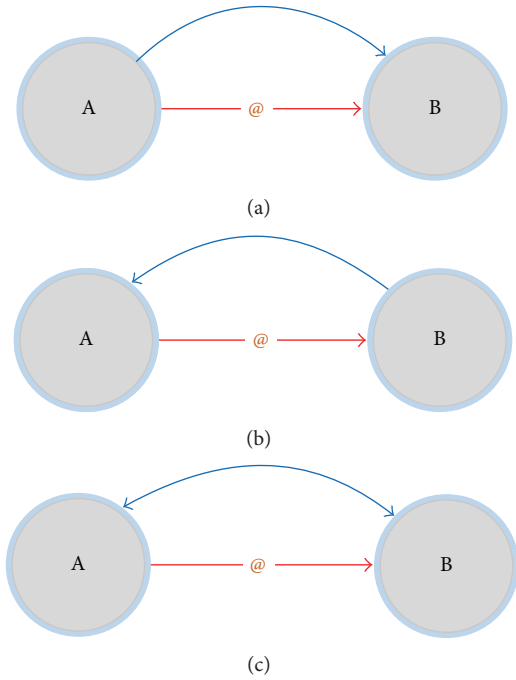


Figure 13: Three situations of user A "@" user B.

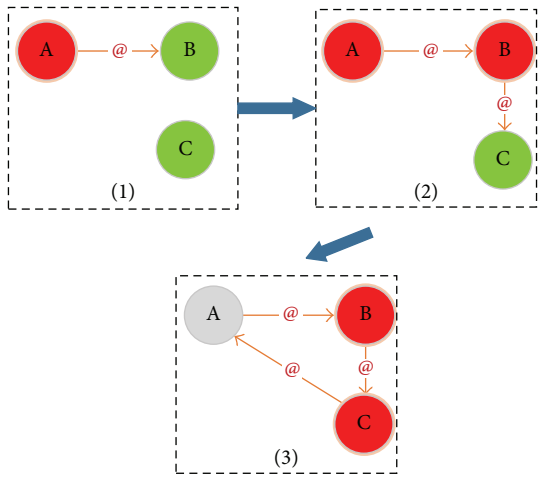


Figure 14: The process of information dissemination in "@" mode.

"@" mode has some differences with disease propagation. Figure 14 depicts the process of information dissemination from node A in "@" mode, and these nodes denote users of the microblog. A node with the green color indicates the person who is ignorant and interested to disseminate the information, a node with the red color means the person who is informed and still interested to disseminate the information, and a node with the gray color means the person who is informed and loses the interest to diffuse the information. These situations denote users in state S (green color), state I (red color), and state R (gray color), respectively.

For the previous research of information dissemination, we could assume that if a person receives the information and he/she will become informed ($I(i)$) then he/she will post on a microblog and diffuse the information into one of his/her neighbors with "@" mode. If the neighbor mentioned with "@" in this microblog has not known the information ($S(j)$), then the neighbor will acquire the information and become informed ($I(j)$). If the neighbor has known this information and has been informed and "@" for q times, then this neighbor will lose the interest to diffuse to any other users and become recovered ($R(j)$), as shown in Figure 14. The process could be described as follows:

$$\begin{aligned} I(i) + S(j) &\longrightarrow I(i) + I(j), \\ I(i) + I(j) &\longrightarrow I(i) + R(j), \\ I(i) + R(j) &\longrightarrow I(i) + R(j). \end{aligned} \tag{9}$$

4. Experiments

4.1. Method. To research into information spread in the microblog, we could construct an artificial microblog based on the zombie-city model. Zombie-city model [5–7] is a general model for modeling artificial societies. Agent, role, social network, environment, and rule are included. Rules could constrain agents, environments, and social network; that is, the agents, the environment, and social network must conform to these rules. Besides, agents have their own capabilities (e.g., move), and agents may be infected with viruses through interactions (e.g., "@" in microblogs) so that they will carry viruses. Agents could dynamically play different roles to adapt to various changes with the mechanism of dynamically playing roles [25]. Figure 15 depicts the metamodel of the zombie-city model.

Based upon the zombie-city model, we could abstract the users of the microblog as agents, with the characteristics of proactivity, sociality, self-adaption, and interactivity. The following relationships between these users could be considered as the directed links of the social network. Then, in the circle of information dissemination, we could define three roles: S (is ignorant for the information), I (is informed to the information and interested to disseminate the information), and R (loses the interest to diffuse the information). Agents could dynamically play different roles with distinctive situations. Besides, there are some rules restricting the agents and social networks. However, in this case, the environment of agents could not be considered, because agents only interact with others through the social network. Above all, we could model this case as the following aspects.

(i) *Agents*. It is assumed that the number of agents is 1000.

(ii) *Role*. It is assumed that there are three roles: S (green color), I (red color), and R (gray color).

(iii) *Rules*. It includes rules of agents and rules of the social network.

Rules of the social network are used to construct a dynamic microblog network, as described in Section 2. All the directed links between agents constitute a social network. A link on the social network could be described

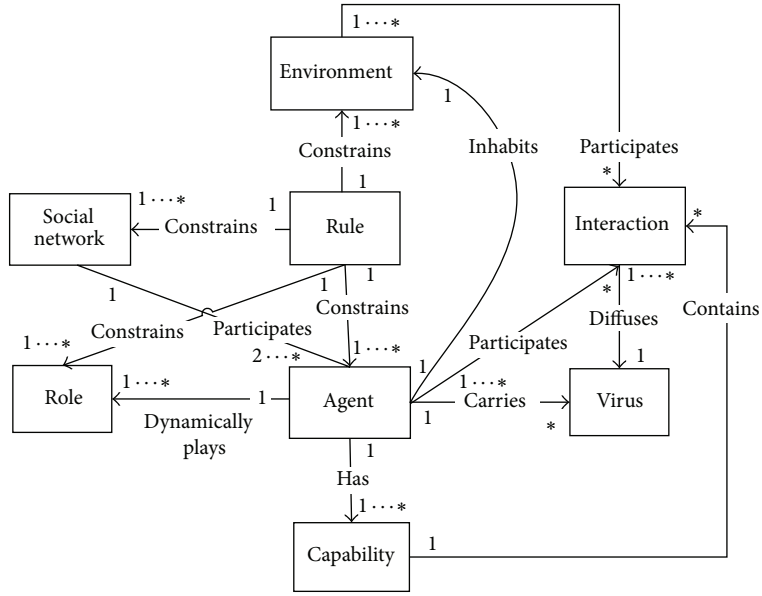


Figure 15: Metamodel of the zombie-city model.

Parameters input: $\alpha, \beta, \lambda, \varepsilon, \kappa, \mu, p_0, p_1$

While $t \in T$
  $t \leftarrow t + 1$
  for $a_i \leftarrow a_1$ to a_N
    $z_{i\rightarrow} \leftarrow \left|LINK^S_{ai,t}\right|$
    for $a_j \leftarrow a_1$ to a_N
      $z_{j\leftarrow} \leftarrow \left|LINK^T_{aj,t}\right|$
      for $a_k \leftarrow a_1$ to a_N
      $m \leftarrow \Sigma \operatorname{Min}\left(\left|LINK^S_{ak,t} \cap LINK^T_{aj,t}\right|, \left|LINK^S_{ai,t} \cap LINK^T_{ak,t}\right|\right)$
      if $a_i \in LINK^S_{aj,t}$
        then $link_{ji} \leftarrow 1$
      if $a_i \notin LINK^S_{aj,t}$
        then $link_{ji} \leftarrow 0$
      $p_{ij} \leftarrow (1 - (p_0/e^{\varepsilon link_{ji}}))(1/(e^{\beta(z_{i\rightarrow} - \text{ownz}^*_{i\rightarrow})} + 1))(1/(1 + e^{-\lambda z_{j\leftarrow}}))(1 - (1 - p_1)e^{-\alpha m})$
      if $a_j \notin LINK^S_{ai,t} \wedge p_{ij} \geq Random(1) \wedge \left|LINK^S_{ai,t}\right| < \text{ownz}^*_{i\rightarrow}$
        then $Create(a_i, a_j) \wedge \Delta t \leftarrow 0$ // Create Link
      if $a_j \in LINK^S_{ai,t} \wedge s_{ij} < 0.3$
        then $Delete(a_i, a_j)$ // Delete Link
      $s_{ij} \leftarrow e^{-\kappa \Delta t}$
      if $s_{ij} \geq 0.3$
        then $\Delta t \leftarrow \Delta t + 1$
end

Pseudocode 1: Pseudocode of constructing the dynamic mircoblogging social network.

as $l ::= \langle cid, s_{ij}, (a_i, a_j)\rangle$. The property s_{ij} ($s_{ij} = e^{-\kappa \Delta t}$) denotes the strength of this link, which will decay with the time Δt, and Δt means the age of the link. At any time, randomly choose an agent a_i and also randomly choose an agent a_j from others; if a_i does not connect to a_j, p_{ij} ($p_{ij} = f_1(z_{i\rightarrow}) f_2(\text{link}_{ji}) f_3(z_{j\leftarrow}) g(m)$) is greater than a random number (less than 1), and out-degree of a_i is less than the threshold of out-degree of a_i ($\text{ownz}^*_{i\rightarrow}$), then a_i will create a link from a_i to a_j. If there is a link from a_i to a_j and s_{ij} is less than a threshold (in this case we set the threshold as 0.3), then a_i will delete the link from a_i to a_j. This process of constructing the dynamical social network could be described by pseudocode, as shown in Pseudocode 1. For any agent a_i, $\text{LINK}^S_{ai,t}$ means an agents set. If a_i links to agent b, then b will belongs to $\text{LINK}^S_{ai,t}$. Meanwhile, $\text{LINK}^T_{ai,t}$ is also an agent set. $|\text{LINK}^S_{ai,t}|$ means the total number of agents belonging to this set. When any agent a links to a_i, agent

Parameters input: $\psi, \rho, \varsigma, \sigma$
(1) **While** $t \in T$
(2) $t \leftarrow t + 1$
//Initialization & PostMicroblog
(3) **if** $Average\_degree > 4 \wedge a_i = Random(\mathrm{AG}) \wedge Total(S) = |\mathrm{AG}|$
(4) **then** $a_i.Quit(S) \wedge a_i.Play(I)$
(5) **for** $a_i \leftarrow a_1$ *to* a_N
(6) **for** $a_j \leftarrow a_1$ *to* a_N
//Microblog Dissemination
(7) **if** $\psi \geq Random(1) \wedge a_i \Downarrow_t I \wedge a_j \in LINK^S_{ai,t} \wedge a_j \notin LINK^T_{ai,t} \wedge a_j \Downarrow_t S \wedge Random(1) < \rho$
(8) **then** $a_i.Post\_Blog\_@(a_j) \wedge a_i.Quit(S) \wedge a_i.Play(I) \wedge a_j.Quit(S) \wedge a_j.Play(I) \wedge a_j.weight{+}{+}$
(9) **if** $\psi \geq Random(1) \wedge a_i \Downarrow_t I \wedge a_j \in LINK^T_{ai,t} \wedge a_j \notin LINK^S_{ai,t} \wedge a_j \Downarrow_t S \wedge Random(1) < \varsigma$
(10) **then** $a_i.Post\_Blog\_@(a_j) \wedge a_i.Quit(S) \wedge a_i.Play(I) \wedge a_j.Quit(S) \wedge a_j.Play(I) \wedge a_j.weight{+}{+}$
(11) **if** $\psi \geq Random(1) \wedge a_i \Downarrow_t I \wedge a_j \in LINK^S_{ai,t} \wedge a_j \in LINK^T_{ai,t} \wedge a_j \Downarrow_t S \wedge Random(1) < \sigma$
(12) **then** $a_i.Post\_Blog\_@(a_j) \wedge a_i.Quit(S) \wedge a_i.Play(I) \wedge a_j.Quit(S) \wedge a_j.Play(I) \wedge a_j.weight{+}{+}$
//Lose Interest
(13) **if** $\psi \geq Random(1) \wedge a_i \Downarrow_t I \wedge a_j \in LINK^S_{ai,t} \wedge a_j \notin LINK^T_{ai,t} \wedge a_j \Downarrow_t I \wedge Random(1) < \rho \wedge weight > q$
(14) **then** $a_j.Quit(I) \wedge a_j.Play(R)$
(15) **if** $\psi \geq Random(1) \wedge a_i \Downarrow_t I \wedge a_j \in LINK^T_{ai,t} \wedge a_j \notin LINK^S_{ai,t} \wedge a_j \Downarrow_t I \wedge Random(1) < \varsigma \wedge weight > q$
(16) **then** $a_j.Quit(I) \wedge a_j.Play(R)$
(17) **if** $\psi \geq Random(1) \wedge a_i \Downarrow_t I \wedge a_j \in LINK^S_{ai,t} \wedge a_j \in LINK^T_{ai,t} \wedge a_j \Downarrow_t I \wedge Random(1) < \sigma \wedge weight > q$
(18) **then** $a_j.Quit(I) \wedge a_j.Play(R)$
(19) **end**

Pseudocode 2: Pseudocode of agents' behaviors.

a will belong to $\mathrm{LINK}^T_{ai,t}$. The functions $Create(a_i, a_j)$ and $Delete(a_i, a_j)$ represent to create a link from a_i to a_j and to delete the link from a_i to a_j, respectively.

Agents in the artificial microblog model are used to map users of the microblog, and initially all agents are playing S role; that is, all agents are ignorant to the information in the initial.

When the average degree of this microblog network reaches 4, randomly select an agent to get and post the information, and then this agent a_i will quit the role S and start to play the role I, which means that agent a_i has known and posted the information. The rules of retweeting the microblog with the information are complex. If the active level of a_i (any infected agent) is greater than ψ (i.e., $Random(1) < \psi$), and then a_i may spread information to others (signed as a_j). But one of the following situations should be satisfied: (1) a_j is one of the out-neighbors of a_i and a_j does not connect to a_i, meanwhile, a_j is playing role S and the random probability is less than ρ; (2) a_j connects to a_i and a_i does not link to a_j. Besides, a_j is playing role S and the random probability is less than ς; (3) a_j connects to a_i and a_i links to a_j, and a_j is playing role S and the random probability is less than σ. If a_i posts or retweets the information to a_j through "@" mode, then a_j will quit role S but play role I. When agents have been informed for q times with "@", they will lose the interest to retweet the information. Pseudocode 2 shows the pseudocode of agents' behaviors. $Total(S)$ means the total number of agents who play the role S, and $|AG|$ denotes the population of agents in this artificial microblog. $a_i.Quit(S)$ and $a_i.Play(I)$ denote that agent a_i stops to play role S and agent a_i plays role I. Function $a_i.Post\_Blog\_@(a_j)$ means agent a_i posts or retweets a microblog with the information through "@" a_j. Meanwhile, the parameter *weight* denotes the total "@" times of an agent received. When the *weight* value of any agent a_i is greater than q, agent a_i will lose the interest to retweet the information to other agents.

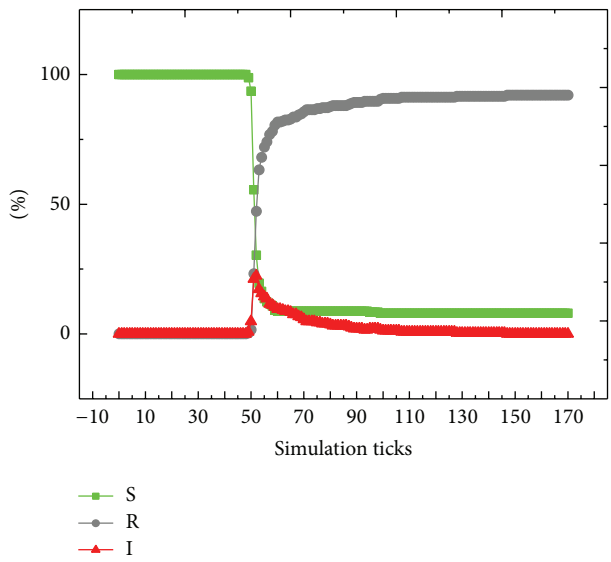
In order to compare the effectiveness of information dissemination in different scenes and situations, four scenes are defined as follows.

(1) Scene 1. Information spreads in a microblog network through "@" mode. We could assume these parameters as follows: $N = 250$, $\beta = 10$, $p_0 = 0.9$, $\varepsilon = 100$, $\lambda = 0.05$, $p_1 = 0.005$, $\alpha = 10$, $\kappa = 0.005$, $\psi = 0.7$, $\mu = 5$, $\rho = 0.05$, $\varsigma = 0.2$, and $\sigma = 0.5$, so the average out-degree of the microblog network is about 5.

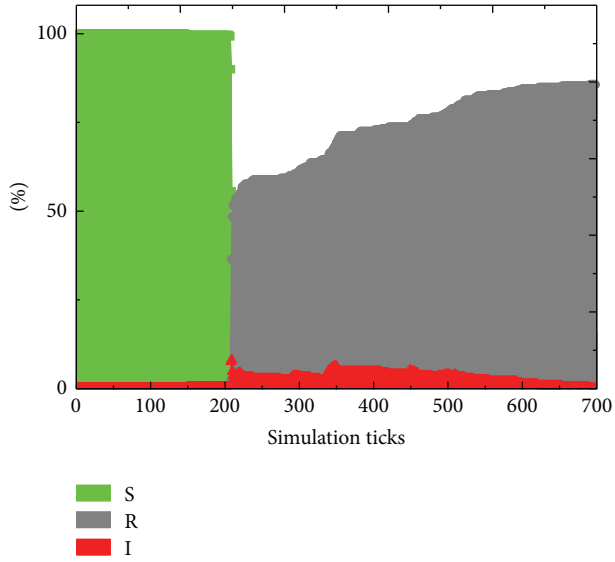
(2) Scene 2. In order to illustrate that whether and how the stability of the microblog network affects the information diffusion, we could change κ from 0.005 to 0.1, and other parameters are the same as (1). In this paper, it is assumed that the number of nodes in the microblog network will not increase or decrease. The stability of the microblog network is related to links of the network, and a better stability means fewer links will be deleted in each time.

(3) Scene 3. As one knows, the active level of agents may affect the information diffusion, so in this scene we set these three parameters as $\rho = 0.1$, $\varsigma = 0.3$, and $\sigma = 0.7$, and the other parameters are the same as (1).

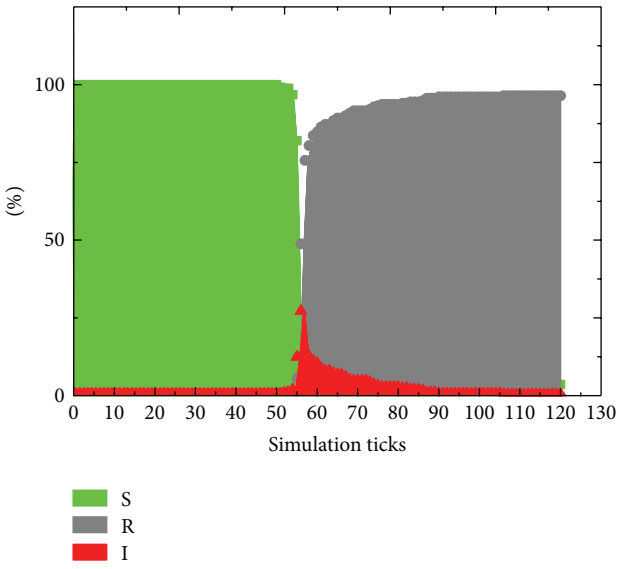
(4) Scene 4. Information disseminates in a random network through "@" mode. In order to depict the characteristics of



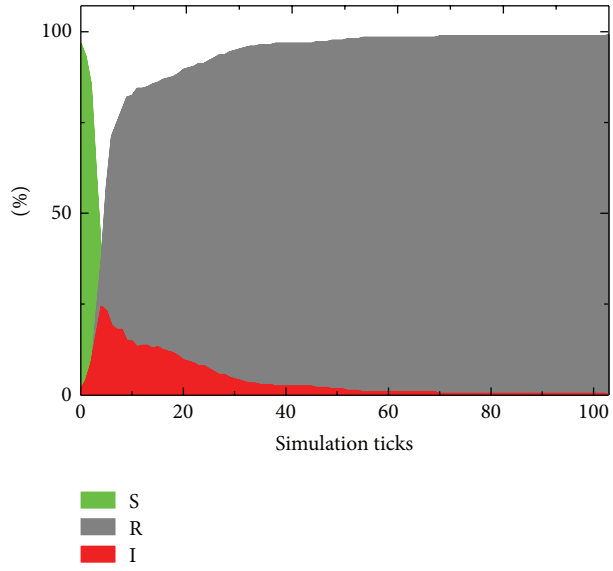
(a) Results of information dissemination in the microblog network with Scene 1



(b) Results of information dissemination in the microblog network with Scene 2



(c) Results of information dissemination in the microblog network with Scene 3



(d) Results of information dissemination in a random network with Scene 4

Figure 16: Results of information dissemination in different scenes.

the information dissemination on the microblog network, we could compare it with information spread in a random network. In this scene, we should set the average degree of this random network as 5. Thus, the conditions are as follows: $N = 250$, average degree = 5, $q = 10$, $\rho = 0.1$, $\varsigma = 0.3$, and $\sigma = 0.7$.

4.2. Results and Analysis. We have implemented the experimental simulations in Netlogo (widely used in simulation of multiagents systems), and these experiments are executed on a workstation with a 4 GB RAM and a 2.60 GHZ Intel Core i5 in Win 8 operation system. To compare the information dissemination in the microblog network with the information spread in the random network, we could define a random directed network with average out-degree = 5. Figure 16(a) presents the information dissemination in the microblog network with the defined parameters: $N = 250$, $\beta = 10$, $p_0 = 0.9$, $\varepsilon = 100$, $\lambda = 0.05$, $p_1 = 0.005$, $\alpha = 10$, $\kappa = 0.005$, $\psi = 0.7$, $\mu = 5$, $\rho = 0.05$, $\varsigma = 0.2$, and $\sigma = 0.5$. Figure 16(b) depicts the information dissemination in the microblog with only changing the parameter κ to 0.1. Figure 16(c) shows the information diffusion in the microblog with changing ρ, ς, and σ to 0.1, 0.3, and 0.7, respectively. Figure 16(d) illustrates the information dissemination in the random network with average degree = 5. x-axis denotes the simulation ticks, and y-axis indicates the percent of three roles that agents play.

Table 1: Comparisons of information spread in different scenes.

	Time cost	ϑ	ϱ
Scene 1	99 ticks (49~147 ticks)	8%	22.4%
Scene 2	514 ticks (148~691 ticks)	14.4%	6.4%
Scene 3	47 ticks (50~96 ticks)	4.4%	27.6%
Scene 4	104 ticks (0~103 ticks)	0.8%	24.4%

Table 1 quantitatively analyzes and compares the effectiveness of information spread in these four scenes. Time cost means the time consumption from the first time of posting information to the end time when no informed users have interest to spread the information. The time cost expresses the speed of information dissemination, and a smaller time cost represents higher speed of information spread. The metrical parameter ϑ (min percent of S) means the minimum percent of agents who play the role S (i.e., agents who are ignorant to the information). A larger ϑ indicates that a greater number of agents have not known the information. In addition, the metrical parameter ϱ (max percent of I) denotes the maximum percent of agents who play the role I (i.e., agents who know the information and do not lose interest to retweet the information). A larger ϱ means more agents are informed and interested in retweeting the information.

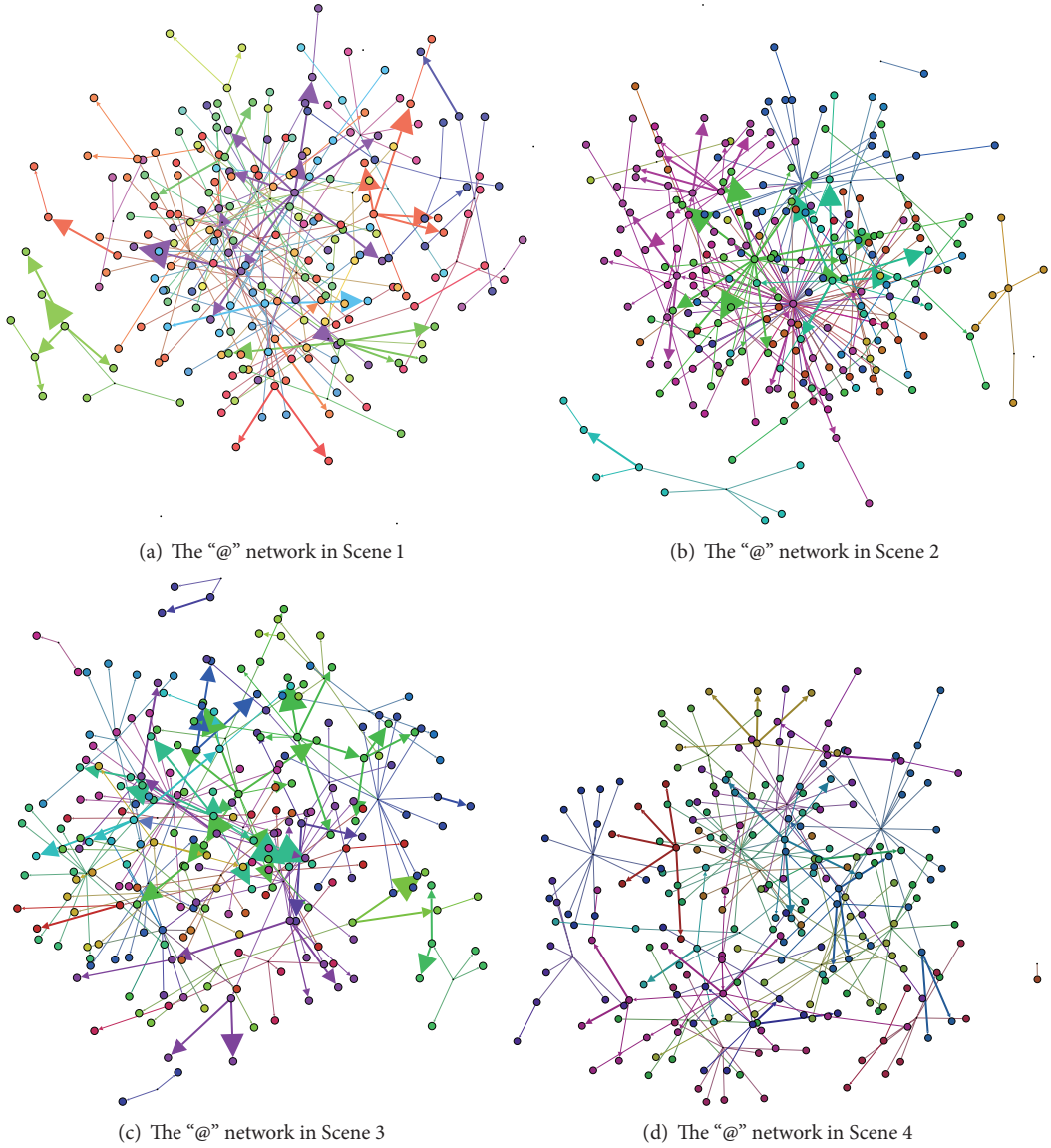
Compared with Scene 1, Scene 2 only has increased the value of parameter κ from 0.005 to 0.1, which is related to the stability of the social network. A larger κ denotes the worse stability of the social network. As shown in Table 1, in Scene 2, the time cost is 514 ticks, ϑ is 14.4%, and ϱ is only 6.4%. However, in Scene 1, the time cost is 99 ticks, ϑ is 8%, and ϱ is only 22.4%. Hence, information spread more widely and quickly in Scene 1; that is, a better stability of the social network is beneficial to improve information dissemination. Compared with Scene 1, Scene 3 only has adjusted the values of parameters ρ, ς, and σ from 0.05, 0.2, and 0.5 to 0.1, 0.3, and 0.7, respectively. We could see that in Scene 3 information spreads more quickly and widely than in Scene 1, which means the higher active level of agents will promote information spread. Scene 4 describes information dissemination in a random social network, and the values of parameters ρ, ς, and σ in Scene 4 are the same with the values of the corresponding parameters in Scene 3. Meanwhile, the average degree of the random network is 5. As presented in Table 1, we could get a conclusion that information spreads quickly in Scene 3 but spreads widely in Scene 4. That is, with a similar average degree, information could spread more quickly in a microblog network, and information may spread more widely in a random network.

We could collect the data of the "@" networks, which are also the information diffusion networks. Figure 17 shows the networks of information dissemination in different scenes. Figures 17(a), 17(b), 17(c), and 17(d) illustrate the information diffusion networks according to Scenes (1), (2), (3), and (4), respectively. This network also reflects the "@" network, because information diffusion in this case is based on the "@" mode. Modularity of the network (a) is 0.924, modularity of network (b) is 0.897, modularity of network (c) is 0.925, and modularity of network (d) is 0.922. All these information dissemination networks have the characteristic of community. The "@" mode of information dissemination determines the bigger modularity of the information diffusion networks. In this case, we only consider the "@" mode for information diffusion. For the real data, we could see that the modularity of information diffusion network (i.e., "@" network) of the real microblog network is 0.848, as seen in Section 2.3.4. Hence, the modularity of these information dissemination networks coincides to the real data. When we adjusted the values of ρ, ς, and σ in Scene 4 from 0.1, 0.3, and 0.7 to 0.05, 0.2, and 0.5, the modularity of "@" network became 0.877, which is less than 0.922. The higher active level of users may make a larger modularity. However, the smaller values of ρ, ς, and σ also bring a high modularity due to the information dissemination mode—the "@" mode.

Above these data and analyses, we could get some conclusions. (1) A random network is more conducive for diffusing the information widely; (2) information spread more quickly in a stable microblog network; (3) the decay rate of the relationships will have an effect on information dissemination; that is, information spreads more quickly and widely with a lower decay rate; (4) the higher active level of users in microblog could also promote information spread; (5) the "@" mode of information dissemination makes a high modularity of the information diffusion network.

4.3. Discussion. Why the random network is more conducive for widely diffusing the information? As shown in Figure 18, (a) shows the process of the information dissemination in a random network, and (b) presents the process of the information dissemination in a microblog network constructed by our model. The average out-degree of this random network and the dynamic microblog network both are 1. For this random network, in Figure 18(a), each node has some following or followed nodes. This results in better connectivity of this random network, which is beneficial to promote information diffusion on the network. For this microblog network in Figure 18(b), the microblog network has a bigger clustering coefficient. However, these nodes a, b, and c of the microblog network have not connected with nodes e, f, and g. Information could not be disseminated to nodes of e, f, and g. Better connectivity of the random network makes information spread widely in the random network.

Why does information spread more quickly in the microblog network? As presented in Figure 18, both nodes "a" in these two figures are original promoters of posting information. As one knows, the max distance between node a and other nodes in the random network is 3, that is, the distance between node "a" and node "c" or the distance between node "a" and node "d", as shown in Figure 18(a). However, the max distance between node "a" and other nodes is 1, that is, the distance between node "a" and node "b" or the distance between node "a" and node "f." This max distance affects the speed of information spread. Smaller max distance



(a) The "@" network in Scene 1

(b) The "@" network in Scene 2

(c) The "@" network in Scene 3

(d) The "@" network in Scene 4

Figure 17: The "@" networks of information dissemination in four scenes.

of the microblog network makes information spread more quickly.

These models and conclusions in this paper could also be used to explain some phenomena of social media. For example, the Sina Weibo is more popular than Renren (a social media platform like Facebook), and we could explain this phenomenon with our model. The reason may be the Matthew effect described in our model. Because many famous stars or persons do not have Renren accounts but have Sina Weibo accounts. For example, Kun Chen (陈坤) is a very famous movie star in China, and he does not have a Renren account but the number of his fans in Sina Weibo is more than 70 million (the number of Sina Weibo users over the world is about 500 million), which means that about 14% users in Sina Weibo have followed Kun Chen. As shown in Figure 19, Kun Chen looks like the node "s", some persons like node "1" and node "2" want to follow the node "s." Hence, both node "1" and node "2" will register in Sina Weibo, and then an increasing number of people will register as the users in Sina Weibo for Matthew effect.

5. Conclusion

Borrowing the ideas of social computing and artificial society, we could make several experiments in artificial societies that are the mappings of the microblogs or real societies. As we know, the social network plays the key role in the information dissemination. This paper has proposed a dynamic growing microblog network based upon the characteristics of the real microblog such as the Sina Weibo. The microblog network is a directed network with the characteristics of higher clustering coefficient, higher modularity, community, and power-law degree distribution. Meanwhile, to facilitate simulations, we have provided another microblog model, which is similar

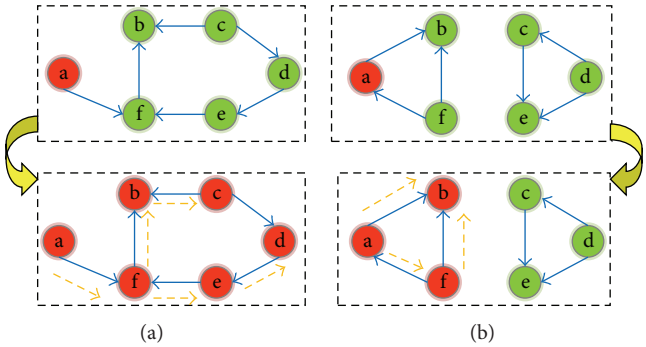


Figure 18: (a) The information dissemination in the random network (average out-degree is 1); (b) The information dissemination in the microblog network (average out-degree is 1).

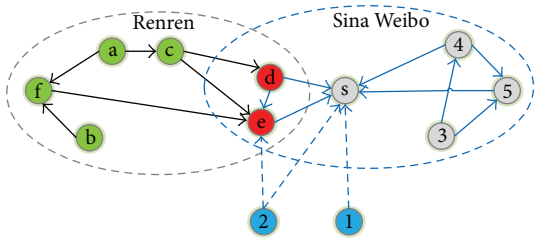


Figure 19: Why Sina Weibo is more popular than Renren.

to the former model. Information could be transmitted through several ways or modes, and the "@" mode is an important and stronger way for information dissemination in the microblog. In order to research into information dissemination with "@" mode in the microblog, aid in modeling artificial societies, and quantitatively and qualitatively analyzing the information dissemination and emerging phenomenon, we have introduced a proposed general artificial society model—zombie-city. Based on the microblog network and the zombie-city model, this paper has modeled the artificial microblog with the zombie-city and analyzed the information dissemination in "@" mode with different scenes. Therefore, through these experimental simulations and analysis, we have acquired some general and interesting conclusions. (1) A random network is more conducive for diffusing the information widely; (2) information spread more quickly in a stable microblog network; (3) the decay rate of the relationships will have an effect on information dissemination; that is, information spreads more quickly and widely with a lower decay rate; (4) the higher active level of users in microblog could also promote information spread; (5) the "@" mode of information dissemination makes a high modularity of the information diffusion network.

In addition, some works should be done or improved. Our future works are foreseen as follows: (1) collecting some real data about social network and information dissemination in Sina Weibo to verify the models proposed in this paper; (2) studying other modes of information dissemination in social media, and (3) synthetically considering all these modes to research into information dissemination in social media based on the zombie-city model research into control and management of information dissemination in the microblog.

Conflict of Interests

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

Acknowledgments

This work is supported by National Nature and Science Foundation of China under Grants nos. 61379051 and 61133001, Program for New Century Excellent Talents in University under Grant no. NCET-10-0898, and Open Fund State Key Laboratory of Software Development Environment under Grant no. SKLSDE-2012KF-0X. Thanks also go to the two reviewers and the editor for their assistances in perfecting this paper.

References

[1] CNNIC, "Statistics Report of Chinese Internet Development Status," 2014, http://www.cnnic.cn/hlwfzyj/hlwxzbg/hlwtjbg/201403/t20140305_46240.htm (in Chinese).

[2] L. Anderson, "Demystifying the Arab spring: parsing the differences between Tunisia, Egypt, and Libya," *Foreign Affairs*, vol. 90, no. 3, p. 2, 2011.

[3] S. Johnstone and J. Mazo, "Global warming and the Arab spring," *Survival*, vol. 53, no. 2, pp. 11–17, 2011.

[4] J. M. Epstein and R. Axtell, *Growing Artificial Societies—Social Science from the Bottom up*, Brooking Institution Press, Washington, DC, USA, The MIT Press, London, UK, 1996.

[5] M. Tang, X. Mao, and H. Zhou, "Zombie-city: a new artificial soceity model," *Journal of Computational Information Systems*, vol. 9, no. 12, pp. 4989–4996, 2013.

[6] M. Tang, X. Mao, and Z. Guessoum, "Research on an infectious disease transmission by flocking birds," *The Scientific World Journal*, vol. 2013, Article ID 196823, 7 pages, 2013.

[7] M. Tang, X. Mao, Z. Guessoum, and H. Zhou, "Rumor diffusion in an interests-based dynamic social network," *The Scientific World Journal*, vol. 2013, Article ID 824505, 10 pages, 2013.

[8] E. M. Jin, M. Girvan, and M. E. J. Newman, "Structure of growing social networks," *Physical Review E*, vol. 64, no. 4, Article ID 046132, 2001.

[9] M. E. J. Newman, "Properties of highly clustered networks," *Physical Review E*, vol. 68, no. 2, Article ID 026121, 2003.

[10] Q. Yan, L. Wu, and L. Zheng, "Social network based microblog user behavior analysis," *Physica A: Statistical Mechanics and Its Applications*, vol. 392, no. 7, pp. 1712–1723, 2013.

[11] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[12] F. Pengyi, W. Hui, J. Zhihong, and L. Pei, "Measurement of Microbloging network," *Journal of Computer Research and Development*, vol. 49, no. 4, pp. 691–699, 2012.

[13] D. J. Watts and S. H. Strogatz, "Collective dynamics of "small-world" networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[14] S. E. Ahnert and T. M. A. Fink, "Clustering signatures classify directed networks," *Physical Review E*, vol. 78, no. 3, Article ID 036112, 2008.

[15] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, Article ID P10008, 2008.

[16] R. Lambiotte, J. C. Delvenne, and M. Barahona, "Laplacian dynamics and multiscale modular structure in networks," http://arxiv.org/abs/0812.1770.

[17] A. Guille and H. Hacid, "A predictive model for the temporal dynamics of information dissemination in online social networks," in *Proceedings of the International World Wide Web Conference Committee (IW3C2)*, pp. 1145–1152, Lyon, France, 2012.

[18] E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic, "The role of social networks in information dissemination," in *Proceedings of the International World Wide Web Conference Committee (IW3C2)*, pp. 512–528, Lyon, France, 2012.

[19] J. Wei, B. Bua, and L. Liang, "Estimating the diffusion models of crisis information in micro blog," *Journal of Informetrics*, vol. 6, pp. 600–610, 2012.

[20] F. Xiong, Y. Liu, Z.-j. Zhang, J. Zhu, and Y. Zhang, "An information diffusion model based on retweeting mechanism for online social media," *Physics Letters A*, vol. 376, no. 30, pp. 2103–2108, 2012.

[21] F. A. Rihan, M.-N. Anwar, M. Sheek-Hussein, and S. Denic, "SIR model of swine influenza epidemic in Abu Dhabi: estimation of vaccination requirement," *Journal of Public Health Frontier*, vol. 1, no. 4, pp. 85–89, 2012.

[22] J. Zhou, Z. Liu, and B. Li, "Influence of network structure on rumor propagation," *Physics Letters A*, vol. 368, no. 6, pp. 458–463, 2007.

[23] D. Gruhl, D. Liben-Nowell, R. Guha, and A. Tomkins, "Information diffusion through blogspace," in *Proceedings of the 13th International World Wide Web Conference Proceedings (WWW '04)*, pp. 491–501, May 2004.

[24] Z. Liu, Y.-C. Lai, and N. Ye, "Propagation and immunization of infection on general networks with both homogeneous and heterogeneous components," *Physical Review E*, vol. 67, no. 3, Article ID 031911, 2003.

[25] X. Mao, L. Shan, H. Zhu, and J. Wang, "An adaptive casteship mechanism for developing multi-agent systems," *International Journal of Computer Applications in Technology*, vol. 31, no. 1-2, pp. 17–34, 2008.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 920829, 13 pages
http://dx.doi.org/10.1155/2014/920829

Research Article

Dynamic Request Routing for Online Video-on-Demand Service: A Markov Decision Process Approach

Jianxiong Wan,[1] Limin Liu,[1] and Jianwei Guo[2]

[1] *Inner Mongolia University of Technology, Hohhot 010051, China*
[2] *Liaoning Technical University, Fuxin 123000, China*

Correspondence should be addressed to Jianxiong Wan; jxwan@csnet1.cs.tsinghua.edu.cn

Received 22 January 2014; Accepted 4 May 2014; Published 1 June 2014

Academic Editor: Guoqiang Hu

We investigate the request routing problem in the CDN-based Video-on-Demand system. We model the system as a controlled queueing system including a dispatcher and several edge servers. The system is formulated as a Markov decision process (MDP). Since the MDP formulation suffers from the so-called "the curse of dimensionality" problem, we then develop a greedy heuristic algorithm, which is simple and can be implemented online, to approximately solve the MDP model. However, we do not know how far it deviates from the optimal solution. To address this problem, we further aggregate the state space of the original MDP model and use the bounded-parameter MDP (BMDP) to reformulate the system. This allows us to obtain a suboptimal solution with a known performance bound. The effectiveness of two approaches is evaluated in a simulation study.

1. Introduction

The advancements of Internet technology have remarkably improved the capacity of both core networks and access networks and enable complex bandwidth-consuming applications which are far beyond simple text-based web page browsing. Among all these applications, Video-on-Demand (VoD) service has gained great popularity over the past decade. According to the report of the world's largest content delivery network (CDN) provider, Akamai, the global average connection speed experienced over 20% growth during 2010 [1], partially due to the surging video content transferring via the Internet.

Nowadays, commercial VoD services on the Internet like YouTube [2] and Hulu [3] are typically supported by CDNs. In RFC 3466, Internet Engineering Task Force (IETF) defined CDN as a type of content network which is a virtual network on top of a generic IP network. CDNs contain many infrastructures (edge servers) distributed across vast geographical areas. Clients can fetch the video streaming data from edge servers instead of the source server, avoiding the network congestion and source server overloaded efficiency.

Modern CDN providers often build their system over cloud infrastructures to reduce the deployment and maintenance cost while extending the system scalability. From this sense, Lenk et al. [4] categorized Amazon's CDN service CloudFront and Akamai's EdgePlatform as higher infrastructure services. There is vast number of literatures concentrated on cloud-based CDN system. The recent AirLift system [5] and the CALMS system [6] demonstrated the feasibility of deploying video streaming applications in cloud-based CDN. Li et al. [7] regarded network congestion at cloud egress and latency to client as one of the major challenges for the current cloud system. They proposed an architecture which integrates CDN with cloud where global load balancing can be achieved with CDN. An experiment in the Microsoft Windows Azure public cloud demonstrated the effectiveness of their idea. Tran et al. [8] developed a content distribution network cloud architecture (CDNCA) based on quality of service (QoS) and quality of experience (QoE) criterions. Jin et al. [9] proposed a content-delivery-as-a-service (CoDaaS) framework to enable on-demand virtual content delivery service (vCDS) for user-generated content (UGC). Chia-Feng et al. [10] presented a cloud-based CDN (CCDN) platform which provides content

delivery features with cloud storage and platform as a service in cloud computing field. Chen et al. [11] investigated a joint problem of building distribution paths and placing Web server replicas in CDNs driven by storage cloud to minimize the cost. Clearly, the techniques to migrate CDN service from traditional distributed computing systems to the cloud are an active research area.

There are also many research efforts in industry focused on this area. A number of major enterprises develop their own cloud-based CDN systems. For example, Amazon CloudFront [12] is a web service for content delivery. It uses Amazon S3 (a cloud storage service) or EC2 (an infrastructure-as-a-service) as the underlying storage servers and edge servers. Akamai NetStorage [13] is another example of cloud storage service built for Akamai's CDN. It is a geographically distributed cloud storage system which provides multiple terabytes of storage capacity. Limelight Networks [14] also builds many data centers by itself all over the world to support its CDN service.

One of the key challenges for CDN providers is to design an appropriate request routing strategy so that CDN providers can obtain as much profit as possible while preserving good performance. In this paper, we investigate the request routing problem in the CDN-based VoD system. The system of our interest is shown in Figure 1. It contains a dispatcher and a number of edge servers. Like Amazon CloudFront, our system can be implemented on a cloud infrastructure. CDN providers can rent virtual machines (VMs) from geographically distributed cloud providers or build their own distributed data centers (DCs). These VMs or DCs act as edge servers which directly transfer VoD streaming data to the clients. The load balance component from cloud system and the geo-aware request routing component from CDN system are integrated into the dispatcher. We assume the dispatcher works in a time-slotted fashion.

Request routing problem is one of the most fundamental issues in the distributed online service providing system. It has received much attention in various environments over the years [15–18]. Among these works, some focused on probing for the design principles behind proprietary request routing algorithm of commercial enterprise and evaluating their effectiveness by using a measurement methodology [19, 20] and others preferred to treat the problem in a more analytical way [21–24]. Xu and Li [25, 26] used decomposition techniques in optimization theory to investigate the request routing problem in various settings. Tran et al. [27] developed a request routing scheme based on the QoE criterion. Qian and Rabinovich [28] developed a heuristic algorithm called permutation prefix clustering to solve the joint application placement and request routing problem. Dealer system [29] abstracts multilayer application into a directed graph where the vertices are application components and the edges are communications between vertices. Dealer tries to minimize response time by searching for a better combination approach of application components. CloudGPS system [30] and NetPaaS system [31] deal with the request routing problem in ISP-P2P and ISP-CDN environments. The highlight of the distributed cooperative request routing algorithms behind these systems is that they can benefit all entities (CDN, ISP, and P2P end user) within the system. In our work, we use the Markov decision process (MDP) [32] rather than static optimization techniques [33, 34] to model the system, since the dynamic nature of MDP can potentially yield remarkable performance improvements [21].

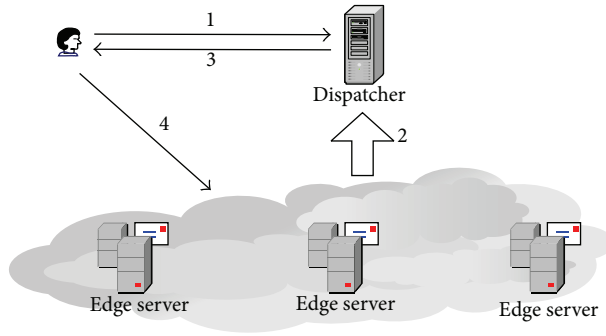


Figure 1: A typical CDN-based VoD service system. The request routing task is fulfilled by the dispatcher. The request routing contains 4 steps: (1) client requests arrive at the dispatcher; (2) the dispatcher samples the state of the servers and computes the request routing strategy; (3) the dispatcher returns the destination edge server information to the clients; (4) client requests are redirected to appropriate edge servers.

Load balance factors must be considered while designing a request routing strategy; it has been intensively studied by the researchers [35–40]. An unbalanced load allocation may cause network congestion and deplete server computational resources and thus greatly degrade user experience. However, starting from the perspective of load balance alone is not enough, since the final goal of the VoD provider is to earn as much profit as possible, rather than only to guarantee the user experience. In a practical CDN-based VoD service system, local costs in different geographical regions, for example, the bandwidth rental cost and the server maintenance cost, are not equal. How to select appropriate server for a request so that the profit is maximized is our focus.

There are many researches concentrated on reward-based dynamic request routing in web-based applications [22–24]. Unfortunately, neither of them can be directly applied to the VoD service for following reasons: (1) a video request will take up the resources of VoD servers throughout the whole viewing period, which is several orders of magnitude longer than a web-based request; (2) as long as a client begins to receive data from an edge server, it cannot be redirected to other edge servers arbitrarily, since this will disturb the continuity of the playback process and degrade the user experience.

In this paper, we investigate the request routing problem in the VoD service by formulating it as a standard MDP, which can be theoretically solved by the classical algorithms like value iteration, policy iteration, or their variations. However, this MDP formulation is practically intractable due to the so-called the curse of dimensionality; that is, the problem size grows exponentially with the underling parameters. To address this issue, we propose two alternative approaches to approximately solve the problem. The first approach is to

use a greedy strategy, that is, to focus on maximizing the reward which can be earned in current time slot, instead of the accumulated reward in the long run. The advantage of the greedy approach is its simplicity; for example, it can be implemented online. The drawback of this approach is that the performance deviation between the optimal strategy and the greedy strategy is unknown. To overcome it, we turn to the second approach—bounded-parameter Markov decision processes (BMDP). The BMDP was proposed by Givan et al. [41, 42] to approximately solve the MDP with a large state space. We use state aggregation technique to partition the overall state space and reformulate the request routing problem as a BMDP. The BMDP formulation has two advantages. (1) we can derive a B-optimal strategy aiming at maximizing the attainable reward in a BMDP model; (2) the upper bound value of the B-optimal strategy is the highest reward a feasible strategy can achieve in the original MDP model, so it can be served as a cornerstone to evaluate other strategies, for example, the greedy strategy. We further conduct experimental study to evaluate the effectiveness of these two algorithms. Simulation results show that the B-optimal strategy is a close-to-optimal request routing solution.

This paper proceeds as follows. Section 2 depicts the model in detail; Section 3 presents the MDP formulation of the request routing problem; in Section 4 we first provide a greedy algorithm to approximately solve the MDP and then reformulate the problem as a BMDP. The interval value iteration (IVI) algorithm is presented thereafter, and a comparison of the IVI algorithm with the traditional value iteration algorithm from the perspective of computational complexity is conducted; a numerical analysis is given in Section 5 followed by conclusions in Section 6.

2. System Model

We use a discrete time controlled queueing model to describe the VoD service system. The model is shown in Figure 2. It consisted of a dispatcher and several edge servers located across a geographic region (e.g., a country). The dispatcher is a centralized controller which collects video requests and redirects them to appropriate edge servers. At a given time slot, the VoD edge servers transmit the video streaming data to each of its clients. At the end of a time slot, some clients close their streaming sessions and leave the system, while others prefer to stay and wait for receiving video stream in the next time slot. Upon each accepted request, the CDN provider can get rewards. Our focus is to find the optimal strategy which maximizes such rewards. Notations are summarized in Notation section.

2.1. Customer Model. Customer categorization is a common approach for studying the online service providing system. In this paper, we regard the partition by the video file that the clients are asking for as impractical since there may be hundreds of thousands of video files in a VoD system. As explained in the following section, the increase of customer types will dramatically raise the state space and action space of the problem, making it hard to solve. We use an alternative

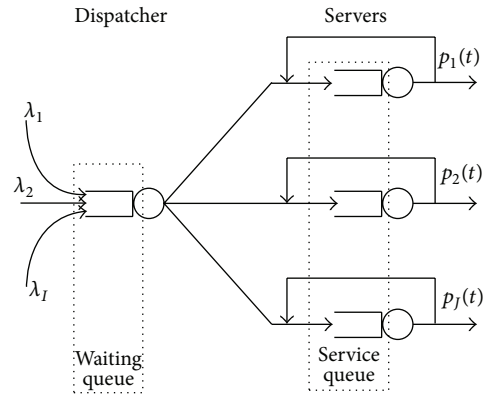


Figure 2: System model for a CDN-based VoD service.

approach, that is, categorizing the customer according to (1) the expected viewing time length and (2) the average bitrates. The first standard ensures that all requests in a given time have the same departure probability (as we can see later), and the second one implies that they consume the same amount of resources. This categorization can be achieved by the following steps.

(i) Measure the expected viewing time for each video file.

(ii) Group video files which have (approximately) the same expected viewing time and bitrates.

(iii) If a customer asks for a video file which belongs to the ith group, mark this request as type i.

The usage of average bitrate is not an exact model since VoD providers sometimes employ a variable bit rate (VBR) approach like MPEG-4 to encode video files for reduction of the file size. This may induce the system to run out of resources if all requests achieve their peak rate at the same time. However, we believe that this impreciseness does not affect our overall model that much for two reasons: (1) the video streaming itself can tolerate packet losses or delay jitters to some extent and (2) the alignment of the peak rate periods of the video streams is rare [43]; we can use some moderate resource reservation strategies to effectively avoid the resource depletion in most cases.

In each time slot, the number of type i video requests arriving at the dispatcher, that is, λ_i, is a random variable, which can be reasonably viewed as subject to some stationary probability distribution in VoD service. To avoid infinite state space problem, we assume that λ_i is a bounded nonnegative integer random variable.

2.2. Server Model. Each VoD edge server is modeled by a discrete time queueing system. The resource capacity of edge server j, that is, output bandwidth, and so forth, is C_j. We assume that a type i request consumes ω_i unit of bandwidth. The resource consumption of all requests in a server cannot exceed its capacity limitation.

In the VoD service, customers' requests tend to "reside in" the system since the playback time of a video file is several orders of magnitude longer than the one in web-based

application. We assume each accepted request must stay in the system for at least one time slot, and all departures happen at the end of a time slot. Each edge server does not log the state of an individual request, that is, the time a request has spent in the system, because this scheme is too resource consuming and not scalable as the number of requests grows. Therefore, we consider the probability that a request leaves the system p_i is equal in every time slot. We have the following result.

Lemma 1. *Suppose a type i request can stay in the system for at most $\overline{T}_i$ of time or $K = \lceil \overline{T}_i/\Gamma \rceil$ of time slot; then*

$$\frac{1}{p_i} - \frac{(Kp_i + 1)(1 - p_i)^K}{p_i} = \left\lceil \frac{T_i}{\Gamma} \right\rceil, \tag{1}$$

where Γ is the slot length and T_i is the average sojourn time of type i request in the system.

Proof. The proof is simply a calculation of expected sojourn time provided that a type i request can stay for at most K slots in the system. Consider

$$\begin{aligned} &\sum_{k=1}^{K} \left\{ p_i \cdot (1 - p_i)^{k-1} \cdot k\Gamma \right\} \\ &\quad = p_i\Gamma \left\{ \sum_{k=1}^{\infty} k(1 - p_i)^{k-1} - \sum_{k=K+1}^{\infty} k(1 - p_i)^{k-1} \right\} \\ &\quad = p_i\Gamma \left\{ \frac{1}{p_i^2} - \frac{(Kp_i + 1)(1 - p_i)^K}{p_i^2} \right\} \\ &\quad = T_i, \end{aligned} \tag{2}$$

completing the proof. □

It is not easy to get the analytical solution of p_i from (1). We propose the following methods to get numerical value of p_i.

(i) When K is small, we can plot the left hand side of (1) with p_i as a variable and locate p_i directly from the figure, as illustrated in Figure 3.

(ii) When K is large, the second term in the left hand side of (1) tends to 0 and can be omitted, yielding an approximation of

$$p_i \approx \left\lceil \frac{\Gamma}{T_i} \right\rceil. \tag{3}$$

3. Problem Formulation

The above model can be formulated as a standard discrete time Markov decision process (MDP). We illustrate the system dynamics in Table 1. During $(t - 1, t)$ the dispatcher buffers the incoming requests in its waiting queue. In time point t, the dispatcher samples the system state including

(1) the arrival vector $\boldsymbol{\lambda}(t) = \{\lambda_i(t)\}$, $i \in I$, which describes the number of type i requests in the waiting queue;

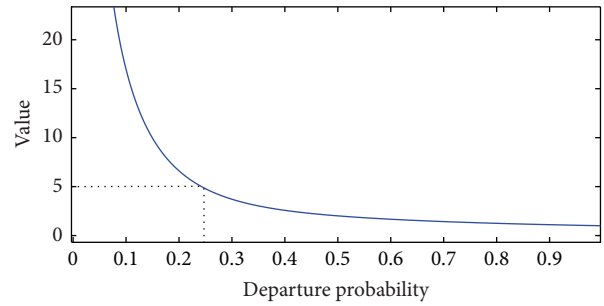


Figure 3: An example to find departure probability with a small $K = 10$. $T_i = 50$ s, $\Gamma = 10$ s. The upper bound of sojourn time is 100 s. We obtain that $p_i \approx 0.25$ in this scenario.

Table 1: System dynamics.

Time	Event
t	The dispatcher samples the system state $(\lambda(t), \mathbf{N}(t))$, and make the decision $a(t)$ forwarding the buffered requests to appropriate edge servers (or rejecting some requests).
$(t, t + 1)$	Requests arrive and are buffered in the dispatcher's waiting queue.
$(t + 1)^-$	Some requests in the service queue leave the system, and the system transits to the next state.

(2) the server load matrix $\mathbf{N}(t) = \{n_{ij}(t)\}$, $i \in I$ and $j \in J$, which describes the number of type i requests in edge server j,

and then it makes a decision. The action is a matrix $a(t) = \{a_{ij}(t)\}$ denoting the number of type i requests forwarded to edge server j. The system then proceeds to the time $(t + 1)^-$, where a reward of $R(\mathbf{N}(t), a(t))$ is received and some requests leave the system. The system transits to another state. The MDP formulation of the problem is given as follows.

States. Denote the state space by $\mathbf{S}$ and denote one element in $\mathbf{S}$ by $(\mathbf{N}, \boldsymbol{\lambda})$. We view the actual state is a function of time, that is, $(\mathbf{N}(t), \boldsymbol{\lambda}(t))$. Note that $\boldsymbol{\lambda}$ can be fully observed by the dispatcher.

Decision Epoch. Decision epoch t is at the start of a slot with length Γ; namely, $t \in \{0, \Gamma, 2\Gamma, \ldots, n\Gamma, \ldots\}$.

Actions. At the beginning of the tth slot, the dispatcher can make a deterministic action $a(t)$ subject to the following constraints:

$$\sum_{j \in J} a_{ij}(t) \le \lambda_i(t), \quad \forall i \in I \tag{4}$$

$$\sum_{i \in I} \omega_i \left(a_{ij}(t) + n_{ij}(t) \right) \le C_j, \quad \forall j \in J, \tag{5}$$

where (4) is the flow conservation constraint representing the forwarded requests which cannot exceed the arrived (and if

forwarded requests are less than the arrived, some requests must be rejected) and (5) implies the bandwidth constraints.

Transition Probability. Since the arrival vector is a stochastic vector and is independent with the server load matrix, we focus mainly on the changes of the latter one. The following dynamic equation can be obtained immediately:

$$\mathbf{N}(t+1) = \mathbf{N}(t) + a(t) - \mathbf{y}(t), \tag{6}$$

where $\mathbf{y}(t) = \{y_{ij}(t)\}$, with $0 \le y_{ij}(t) \le N_{ij}(t) + a_{ij}(t)$, $i \in I$, and $j \in J$, denotes the number of type i departures in server j at the end of tth time slot. Let $g_{ij}^t(n)$ be the probability distribution function of $y_{ij}(t)$; namely, $g_{ij}^t(n) = \Pr(y_{ij}(t) = n)$, and the explicit expression of $g_{ij}^t(n)$ is

$$g_{ij}^t(n) = \begin{cases} \binom{n_{ij}(t) + a_{ij}(t)}{n} (1-p_i)^{n_{ij}(t)+a_{ij}(t)-n} p_i^n, \\ \quad n \le n_{ij}(t) + a_{ij}(t) \\ 0, \\ \quad \text{otherwise.} \end{cases} \tag{7}$$

Let λ_i be subject to some discrete probability distribution function $f_i(x)$; that is, $\Pr(\lambda_i = n) = f_i(n)$; then the transition probability of the system is given by

$$P(\mathbf{N}(t+1), \boldsymbol{\lambda}(t+1) \mid \mathbf{N}(t), \boldsymbol{\lambda}(t), a(t)) = \begin{cases} 0, \\ \quad \text{if } \forall j,\ n_{ij}(t+1) > n_{ij} + a_{ij}(t) \\ \prod_{i \in I} f_i(\lambda_i(t+1)) \\ \quad \cdot \prod_{j \in J} \prod_{i \in I} g_{ij}^t\left(n_{ij}(t) + a_{ij}(t) - n_{ij}(t+1)\right), \\ \quad \text{otherwise.} \end{cases} \tag{8}$$

Rewards. The reward is defined by the VoD service provider, often in the form of $r-c$, where r and c are the revenue and the cost for serving a request, respectively. These two parameters can be defined from different perspectives. For example, from per request perspective, they can be earned right after a request is accepted (pay-per-view); from temporal perspective, they depend upon the sojourn time in the system of each request. In this paper we adopt the latter one. This is reasonable since the more time the client spends in the system, the more profits the VoD provider can potentially earn (e.g., by means of periodically popping up the embedded advertisements). The rewards earned in the tth slot are

$$R(\mathbf{N}(t), \boldsymbol{\lambda}(t), a(t)) = \sum_{i \in I} \sum_{j \in J} \left(n_{ij}(t) + a_{ij}(t)\right) \times \left(r_i - c_{ij}\right), \tag{9}$$

where $n_{ij}(t) + a_{ij}(t)$ is the number of type i clients in server j during the tth slot and $r_i - c_{ij}$ is the net gain for a type i request being served in server j.

Optimization Objective. The optimization objective is to maximize the expected accumulated long term system rewards. In practice the system is actually a finite horizon MDP since the arrival distribution is stationary only in a certain period (let us call it *stationary period*). However, we could use an infinite horizon MDP with discounted object function

$$\mathbb{E}\left\{\sum_{t=0}^{\infty} \delta^t R(\mathbf{N}(t), \boldsymbol{\lambda}(t), a(t))\right\} \tag{10}$$

to approximate it, where $0 \le \delta \le 1$ is the discounted factor, because the length of the stationary period (often in the order of several hours) is much longer than the length of the time slot (often in the order of seconds). This form of objective function is also used in [22]. Consider

$$V(\mathbf{N}(t), \boldsymbol{\lambda}(t)) = \max_{a(t)} \Big\{ R(\mathbf{N}(t), \boldsymbol{\lambda}(t), a(t)) + \delta \sum_{(\mathbf{N}(t+1), \boldsymbol{\lambda}(t+1)) \in \mathbf{S}} P(\mathbf{N}(t+1), \boldsymbol{\lambda}(t+1) \mid \mathbf{N}(t), \boldsymbol{\lambda}(t), a(t)) \cdot V \times (\mathbf{N}(t+1), \boldsymbol{\lambda}(t+1)) \Big\}. \tag{11}$$

The value function can be established as (11), and classical algorithms like value iteration or policy iteration can be used to solve this MDP. However, the above MDP model obviously suffers from the so-called state space explosion problem. To see that, suppose a system with p kinds of requests, q edge servers, and capacity r for each edge server; the total number of states with respect to $\mathbf{N}$ is

$$\left(\sum_{i=0}^{r} \binom{i+p-1}{i}\right)^q. \tag{12}$$

The computational cost of traditional algorithm, in which all states must be visited in each iteration, will be prohibitive. Therefore we need alternative approaches to obtain approximate solutions.

4. Solving the MDP Model Approximately

In this section we present two approaches to approximately solve the above MDP model of request routing problem. The first approach is the greedy strategy, which is an integer linear programming problem aiming at maximizing the reward in current time slot. The second approach is the bounded-parameter MDP (BMDP) strategy, which aggregates the state

space of the original MDP model and applies the interval value iteration algorithm on the aggregated model to obtain the B-optimal solution. We finally discuss the computational complexity of the BMDP strategy.

4.1. A Greedy Approximation Strategy. One of the simplest approximations is the greedy strategy, which focuses on maximizing the reward in each current slot instead of the cumulative reward in the long run. The problem can be summarized as an integer linear programming as follows:

$$\max_{a(t)} \sum_{i\in I} \sum_{j\in J} a_{ij}(t)\left(r_i - c_{ij}\right) \tag{13}$$

subject to constraints (4) and (5).

The idea behind the greedy strategy is straightforward: ignore the arrival pattern of requests and accept as many profitable requests as possible in a current slot. In fact, the greedy strategy does not require the server state **N**. Instead, it only needs to keep track of the total load of each individual server as

$$\mathbf{L} = \left\{L_j\right\}, \quad j \in J, \text{ where } L_j = \sum_{i\in I} n_{ij}(t). \tag{14}$$

We will evaluate the greedy strategy in Section 5.

The greatest advantage of the greedy strategy is its simplicity. Since it involves no iteration operations as in the classical iterative algorithm, its computational time is much smaller and thus can be implemented online. However, the biggest flaw of greedy strategy is that we do not know how far the profits of the greedy strategy deviate from the profits of the optimal strategy. This motivates us to find another approach which can provide some bounding information. In the following subsection, we provide the BMDP approach, which satisfies this property.

4.2. The Bounded-Parameter MDP (BMDP) Approximation Strategy. The BMDP was introduced by Givan et al. [41, 42] to provide an approximate approach for solving the MDP with a large state space; it can be categorized into a more general class known as Markov decision processed with imprecisely known transition probabilities (MDPIPs).

4.2.1. BMDP Preliminaries. A BMDP $M_\updownarrow$ is a four-tuple $\{S, A, R_\updownarrow, P_\updownarrow\}$. It is different from traditional MDP (also called *exact MDP*) in the sense that the reward function in each state $R_\updownarrow(s)$ and the transition probability $P_\updownarrow(s' \mid s, a)$ are specified by closed intervals $[R_\downarrow(p), R_\uparrow(p)]$ and $[P_\downarrow(s' \mid s, a), P_\uparrow(s' \mid s, a)]$, respectively, rather than exact point values. An exact MDP $M = \{S', A', R', P'\}$ is said to be contained in a BMDP $M_\updownarrow$ ($M \in M_\updownarrow$) as long as $S' = S, A' = A, R' \in R_\updownarrow$, and $P' \in P_\updownarrow$.

Given a policy π, the interval value function $V_{\updownarrow\pi}$ is defined by

$$V_{\updownarrow\pi}(s) = \left[\min_{M\in M_\updownarrow} V_{M,\pi}(s), \max_{M\in M_\updownarrow} V_{M,\pi}(s)\right], \tag{15}$$

where $V_{M,\pi}(s) = R_M(s) + \delta \sum_{s'\in S} P_M(s' \mid s, a) V_{M,\pi}(s')$ is the traditional value function for a specific exact MDP $M \in M_\updownarrow$. It can be proved that (see reference [41]) there exists a MDP $M \in M_\updownarrow$ which maximizes/minimizes $V_{M,\pi}(s)$ for all $s \in S$ simultaneously. We call such MDP *π-maximizing/π-minimizing* MDP.

The interval value functions cannot be compared using traditional MDP standard, which focuses on point values. To evaluate how "good" a policy π can achieve, we must define a scheme to compare interval value functions. In this paper we define the interval greater operator $\gtrsim$ given by

$$V_{\updownarrow 1} \gtrsim V_{\updownarrow 2} \Longleftrightarrow V_{\uparrow 1} \geq V_{\uparrow 2} \vee \left(V_{\uparrow 1} = V_{\uparrow 2} \wedge V_{\downarrow 1} \geq V_{\downarrow 2}\right). \tag{16}$$

The interval value iteration (IVI) algorithms with provable convergence property can be used to obtain the optimal solution to the BDMP; that is,

$$\begin{aligned} &\mathrm{IVI}_{\updownarrow \mathrm{opt}}\left(V_\updownarrow\right)(s) \\ &\quad = \max_{a\in A(s)} \left[\min_{M\in M_\updownarrow} \mathrm{VI}_{M,a}\left(V_\downarrow\right)(s), \max_{M\in M_\updownarrow} \mathrm{VI}_{M,a}\left(V_\uparrow\right)(s)\right] \end{aligned} \tag{17}$$

with $\mathrm{VI}_{M,a}(V)(s) = R_M(s) + \delta \sum_{s'\in S} P_M(s' \mid s, a) V(s')$.

The IVI algorithm (17) starts with an arbitrary interval value function $V_\updownarrow = [V_\downarrow, V_\uparrow]$ with $V_\downarrow \leq V_\uparrow$. The operation of finding exact MDPs $M \in M_\updownarrow$ inside the square bracket is equivalent to searching for order-maximizing *MDPs* with respect to state order sequences of decreasing $V_\uparrow$ and increasing $V_\downarrow$. Formal definition of the order-maximizing MDP for a specific state order sequence $O = s_1 s_2 \cdots s_n$ is given by the following definition.

Definition 2. The order-maximizing index r for state s and action a with respect to order O is

$$\arg\max_{1\leq r\leq n} \left\{\sum_{i=1}^{r-1} P_\uparrow(s_i \mid s, a) + \sum_{i=r}^{n} P_\downarrow(s_i \mid s, a)\right\}. \tag{18}$$

The order-maximizing MDP is an exact MDP $M_O \in M_\updownarrow$ satisfying

$$\begin{aligned} P_{M_O}(s_i \mid s, a) &= \begin{cases} P_\uparrow(s_i \mid s, a), & \text{if } i < r, \\ P_\downarrow(s_i \mid s, a), & \text{if } i > r, \end{cases} \\ P_{M_O}(s_r \mid s, a) &= 1 - \sum_{i=1, i\neq r}^{i=n} P_{M_O}(s_i \mid s, a). \end{aligned} \tag{19}$$

Note that the max operator in (17) uses (16) to compare interval value functions. It can be proved (see [41]) that the IVI algorithm will finally converge to an interval value function $[V_{\downarrow\mathrm{opt}}, V_{\uparrow\mathrm{opt}}]$ and an associated solution which we call B-optimal policy. The upper bound of interval value function $V_{\uparrow\mathrm{opt}}$ for the B-optimal policy is the possible best reward one can get from the BMDP $M_\updownarrow$; therefore, it can serve as a cornerstone to evaluate how far the value of a given strategy deviates from the one of possible optimal strategies for any exact MDP $M \in M_\updownarrow$. $V_{\downarrow\mathrm{opt}}$ is the possible worst case performance of the B-optimal strategy.

In our application, states in the BMDP $M_\updownarrow$ can be viewed as aggregations of states in the exact MDP M. The parameter

intervals in $M_{\updownarrow}$ represent the parameter ranges of states in M which belong to the same BMDP state. From this viewpoint $M_{\updownarrow}$ is a "smaller" approximation to the original M.

4.2.2. Model Reduction. The intuition behind our aggregation scheme is to overlook the request type in the server load matrix $\mathbf{N}$. We can construct an aggregate state space $\mathbf{S}'$ with an element $(\mathbf{L}, \boldsymbol{\lambda})$, where $\mathbf{L}$ is the server load vector specified in (14). The system dynamics equation becomes

$$\mathbf{L}(t+1) = \mathbf{L}(t) + a(t) \cdot \mathbf{e} - \mathbf{Y}(t), \tag{20}$$

where $\mathbf{e}$ is a j-dimension unit vector and $\mathbf{Y}$ is the total departures vector. Equation (20) means that the change of server state only depends upon the number of requests assigned to and departing from the server regardless of their types. With this abstraction, the number of state spaces with respect to $\mathbf{L}$ is r^q, a great reduction compared to (12).

4.2.3. The BMDP Reformulation. First we present the following lemma which is used to obtain the interval transition function of the BMDP formulation.

Lemma 3. *Suppose the system is in a particular aggregated state $(\mathbf{L}(t), \boldsymbol{\lambda}(t))$ at the tth slot; the probability of n clients leaving edge server j in this slot, namely, $Pr(Y_j(t) = n)$, can be bounded by*

$$\left[\binom{L_j(t) + \sum_{i \in I} a_{ij}(t)}{n} p_{\min}^n (1 - p_{\max})^{L_j(t) + \sum_{i \in I} a_{ij}(t) - n}, \binom{L_j(t) + \sum_{i \in I} a_{ij}(t)}{n} p_{\max}^n (1 - p_{\min})^{L_j(t) + \sum_{i \in I} a_{ij}(t) - n} \right] \tag{21}$$

provided that $n \le L_j(t) + \sum_{i \in I} a_{ij}(t)$, where $p_{\max} = \max_{i \in I} p_i$ and $p_{\min} = \min_{i \in I} p_i$.

Proof. We show how to derive the upper bound; the lower bound can be obtained using the same idea. Pick any exact MDP state $(\mathbf{N}(t), \boldsymbol{\lambda}(t))$ which belongs to the aggregated state $(\mathbf{L}(t), \boldsymbol{\lambda}(t))$; that is, $\sum_{i \in I} n_{ij}(t) = L_j(t)$. For server j in state $(\mathbf{N}(t), \boldsymbol{\lambda}(t))$, there are a total $\binom{L_j(t) + \sum_{i \in I} a_{ij}(t)}{n}$ number of cases such that n clients leave server j. Choose a particular case where the number of type i requests that leave server j is k_{ij}; we have $\sum_{i \in I} k_{ij}(t) = n$. The probability of the occurrence of this case is

$$\begin{aligned} &\prod_{i \in I} p_i^{k_{ij}} \times \prod_{i \in I} (1 - p_i)^{n_{ij}(t) + a_{ij}(t) - k_{ij}} \\ &\quad \le p_{\max}^{\sum_{i \in I} k_{ij}} (1 - p_{\min})^{\sum_{i \in I} n_{ij}(t) + a_{ij}(t) - k_{ij}} \\ &\quad = p_{\max}^n (1 - p_{\min})^{L_j(t) + \sum_{i \in I} a_{ij}(t) - n}. \end{aligned} \tag{22}$$

The results can be followed immediately. □

We can now reformulate the problem with the BMDP.

States. The aggregated state space is $\mathbf{S}'$, with an element denoted by $(\mathbf{L}, \boldsymbol{\lambda})$.

Decision Epoch and Actions. Decision epoch and actions are the same as in the original MDP model.

Interval Transition Probability. Note that, in the BMDP formulation, the transition probability is a closed interval. First observe that

$$\begin{aligned} &P(\mathbf{L}(t+1), \boldsymbol{\lambda}(t+1) \mid \mathbf{L}(t), \boldsymbol{\lambda}(t), a(t)) \\ &\quad = \prod_{i \in I} f_i(\lambda_i(t+1)) \cdot \prod_{j \in J} P\left(L_j(t+1) \mid L_j(t), \boldsymbol{\lambda}(t), a(t)\right) \\ &\quad = \prod_{i \in I} f_i(\lambda_i(t+1)) \\ &\qquad \cdot \prod_{j \in J} P\left(Y_j(t) = L_j(t) + \sum_{i \in I} a_{ij}(t) - L_j(t+1)\right). \end{aligned} \tag{23}$$

Combining with Lemma 3, the transition probability in (23) can be bounded by the interval transition shown in (24) with $\Theta_j(t) = L_j(t) + \sum_{i \in I} a_{ij}(t)$ and $Y_j(t) = L_j(t) + \sum_{i \in I} a_{ij}(t) - L_j(t+1)$ denoting the number of requests and the number of departures in tth slot in server j regardless of their types, respectively. Consider

$$\begin{aligned} &P_{\updownarrow}(\mathbf{L}(t+1), \boldsymbol{\lambda}(t+1) \mid \mathbf{L}(t), \boldsymbol{\lambda}(t), a(t)) \\ &\quad = \begin{cases} [0, 0], \\ \quad \text{if } L_j(t) + \sum_{i \in I} a_{ij}(t) \le L_j(t+1) \\ \left[p_{\min}^{\sum_{j \in J} Y_j(t)} (1 - p_{\max})^{\sum_{j \in J} L_j(t+1)} \right. \\ \quad \times \prod_{i \in I} f_i(\lambda_i(t+1)) \prod_{j \in J} \binom{\Theta_j(t)}{Y_j(t)}, \\ \quad p_{\max}^{\sum_{j \in J} Y_j(t)} (1 - p_{\min})^{\sum_{j \in J} L_j(t+1)} \\ \quad \left. \times \prod_{i \in I} f_i(\lambda_i(t+1)) \prod_{j \in J} \binom{\Theta_j(t)}{Y_j(t)} \right], \\ \quad \text{otherwise.} \end{cases} \end{aligned} \tag{24}$$

Interval Rewards. The reward consists of two parts. The first part is the reward for the action $a(t)$, which is a fixed value given by $\sum_{j \in j} \sum_{i \in I} a_{ij}(t)(r_i - c_{ij})$. The second part is a closed interval $[mL(t), ML(t)]$, with

$$\begin{aligned} M &= \max_{i \in I, j \in J} \{r_i - c_{ij}\}, \\ m &= \min_{i \in I, j \in J} \{r_i - c_{ij}\}. \end{aligned} \tag{25}$$

The interval reward function can then be expressed by

$$R_{\updownarrow}(\mathbf{L}(t), \boldsymbol{\lambda}(t), \mathbf{a}(t)) = \left[mL(t) + \sum_{j\in j}\sum_{i\in I} a_{ij}(t)\left(r_i - c_{ij}\right), \; ML(t) + \sum_{j\in j}\sum_{i\in I} a_{ij}(t)\left(r_i - c_{ij}\right) \right]. \quad (26)$$

4.2.4. The Interval Value Iteration (IVI) Algorithm. We illustrate the IVI algorithm in detail in Algorithms 1 and 2.

Algorithm 1 takes the above BMDP model $M_{\updownarrow}$ as input. It first chooses an initial interval value function $V_{\updownarrow}$ and then iteratively calls function IVI. The algorithm finishes with an interval value function $V_{\updownarrow}$ and a corresponding policy π.

Algorithm 2 captures the essence of (17). The increasing state order of $V_{\uparrow}$ and the decreasing state order of $V_{\downarrow}$ are stored in O_{up} and O_{down}. For each state s and action a, the algorithm computes the order-maximizing indices r_{up} and r_{down} for order sequences O_{up} and O_{down} and forms two order-maximizing MDPs with transition probabilities P'_{up} and P'_{down}. Using P'_{up}, the set of actions $\mathbf{a}$ which maximizes the upper bound $V'_{\uparrow}$ is identified. If $\mathbf{a}$ contains only one element, for example, $\mathbf{a} = \{a\}$, then a is the B-optimal action in this state and the lower bound can be obtained immediately. Otherwise, an action $a \in \mathbf{a}$ which maximizes the lower bound $V'_{\downarrow}$ is chosen as the B-optimal action.

4.2.5. A Word on Computational Complexity

(i) *Space Complexity*. Although the memory consumption of a single state in the BMDP model is a bit larger than the one in the MDP model due to the interval transition probability and the interval reward, but since the aggregation scheme in our BMDP formulation dramatically reduces the state space of edge servers from $(\sum_{i=0}^{r}\binom{i+p-1}{i})^q$ to r^q, the total memory space needed for storing the BMDP model is significantly less than the MDP model. On the other hand, the IVI algorithm only needs additional O_{up} and O_{down} to store order sequences in BMDP model, r_{up} and r_{down} to store order-maximizing indices, and P'_{up} and P'_{down} to store transition probabilities for the π-maximizing MDP and the π-minimizing MDP, respectively. In all, the memory space needed to store the BMDP model and implement the IVI algorithm decreases greatly compared with the memory space needed for the MDP model.

(ii) *Time Complexity*. In the IVI algorithm, the first optimization problem $(*)$ is much easier than problem (11) in traditional value iteration algorithm since the computation of the expectation in the Bellman equation involves much less states. There are three extra computational burdens in the IVI algorithm: (1) sorting states to obtain O_{up} and O_{down} (lines 5 and 6 in Algorithm 2), which take $O(2n^2)$ in worst case using Quicksort algorithm; (2) finding the order-maximizing indices and computing the transition probabilities for the π-maximizing MDP and the π-minimizing MDP (lines 9, 10, and 12 in Algorithm 2), which take $O(4n)$ in worst case; (3) finding a unique solution $a \in \mathbf{a}$ to the problem $(**)$ which maximizes the lower bound $V_{\downarrow}$ (line 20 in Algorithm 2). In this step we use the exhaustive search, so the time complexity is $O(|\mathbf{a}|)$. However, regarding $\mathbf{a}$ which is usually a small set, it will not take too long to complete the search. Thus, the total extra time complexity is $O(2n^2 + 4n + |\mathbf{a}|)$.

Input: a BMDP $M_{\updownarrow} = \{S, A, P_{\updownarrow}, R_{\updownarrow}\}$
Output: $V_{\updownarrow}$ and π
(1) Create π; {π holds the strategy in each iteration.}
(2) Create $V_{\updownarrow}$; {$V_{\updownarrow}$ is an initial interval value function.}
(3) **loop**
(4) $V_{\updownarrow} = \text{IVI}_{\updownarrow\text{opt}}(M_{\updownarrow}, V_{\updownarrow}, \&\pi)$;
(5) **end loop**

Algorithm 1: Solutions to the BMDP formulation.

5. Numerical Analysis

In this section we study an illustrative system with two video request types and two edge servers. To alleviate the computational and simulation burden, we use a small scale system parameterization. We initially set the server capacity $C_1 = C_2 = 5$. The expected sojourn time of type 1 and type 2 requests (T_1 and T_2) is 2000 s and 1000 s, respectively. According to (3), the departure probabilities can be computed as $p_2 = 0.0025$ and $p_1 = 0.005$ if we set the length of the time slot to 5 s. Other parameters are summarized in Table 2.

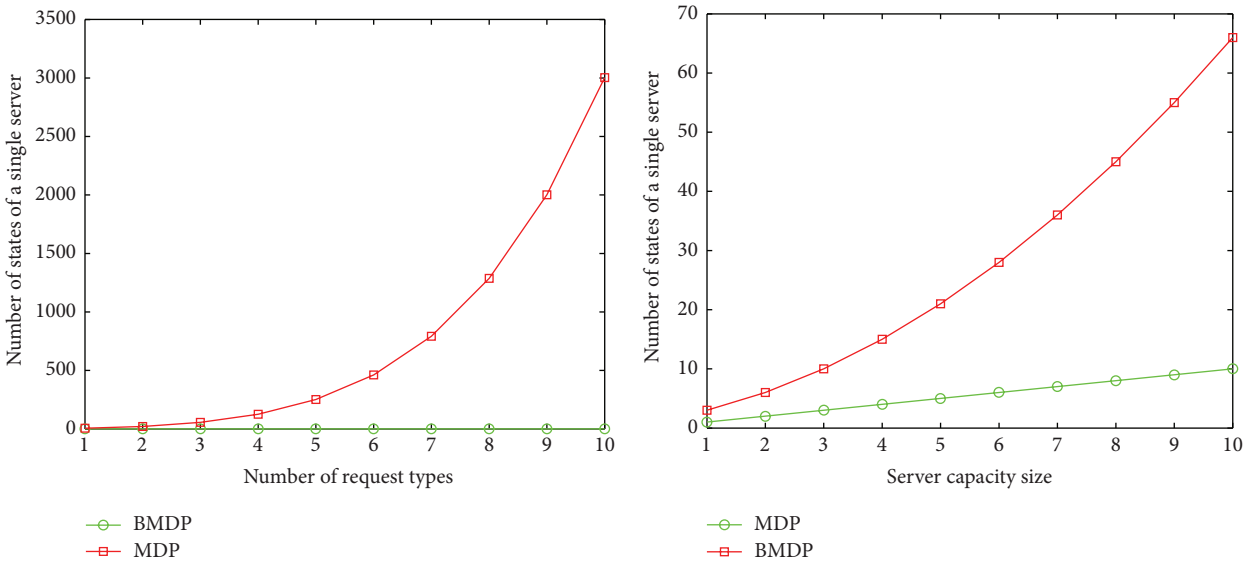
5.1. Problem State Space. First we will see the effectiveness of the state aggregation scheme of the proposed BMDP formulation. We compare the state space of a single server here with respect to MDP formulation and BMDP formulation. The results are shown in Figure 4.

In Figure 4(a), we fix the server capacity to 5 and vary the number of request types. In Figure 4(b), we fix the number of request types to 2 and vary the server capacity. The state space of the BMDP formulation does not change with the number of request types and grows linearly with the server capacity. In contrast, the state space of the MDP formulation increases exponentially with both the number of request types and the server capacity. There are over 3000 states when the number of request type reaches 10, demonstrating that the MDP model is practically intractable.

On the other hand, although the state space of a single server in the BMDP formulation is greatly reduced compared to the counterpart in the MDP formulation, the state space of the whole system in the BMDP formulation still grows

Input: a BMDP $M_{\updownarrow}$, a value function $V_{\updownarrow}$, and a place π for holding the policy in current iteration
Output: $V'_{\updownarrow \text{opt}}$ and π_{opt}
(1) Create O_{up}, O_{down}; {O_{up} and O_{down} hold order sequence of states in $M_{\updownarrow}$, i.e., $s_1 s_2, \ldots, s_n$.}
(2) Create P'_{up}, P'_{down}; {P'_{up} and P'_{down} hold the transition probabilities for the order-maximizing MDP with respect to order O_{up} and O_{down}, respectively.}
(3) Create r_{up}, r_{down}; {r_{up} and r_{down} is the order-maximizing index for order sequences O_{up} and O_{down}, respectively.}
(4) Create i; {i is the index into an ordering O.}
(5) O_{up} = Sort_Decreasing_Order($V_{\uparrow}$);
(6) O_{down} = Sort_Increasing_Order($V_{\downarrow}$);
(7) **for all** $s \in S$ **do**
(8) **for all** $a \in A$ **do**
(9) r_{up} = Order_Maximizing_Ind($M_{\updownarrow}, O_{\text{up}}, s, a$);
(10) r_{down} = Order_Maximizing_Ind($M_{\updownarrow}, O_{\text{down}}, s, a$);
{find order-maximizing index for transition probability in state s under action a according to (18).}
(11) **for** $i = 1$ to n **do**
(12) Update $P'_{\text{up}}(s_{O_{\text{down}}(i)} \mid s, a)$ and $P'_{\text{down}}(s_{O_{\text{up}}(i)} \mid s, a)$ according to (19);
(13) **end for**
(14) **end for**
(15) $V'_{\uparrow} = \max_{\mathbf{a} \subseteq A(s)} R_{\uparrow}(s, a) + \delta \sum_{s' \in S} P'_{\text{up}}(s's, a) V_{\uparrow}(s')$;(∗)
(16) **if** $|\mathbf{a}| = 1$ and $\mathbf{a} = \{a\}$ **then**
(17) $V'_{\downarrow} = R_{\downarrow}(s, a) + \delta \sum_{s' \in S} P'_{\text{down}}(s's, a) V_{\downarrow}(s')$;
(18) $\pi(s) = a$;
(19) **else**
(20) $V'_{\downarrow} = \max_{a \in \mathbf{a}} R_{\downarrow}(s, a) + \delta \sum_{s' \in S} P'_{\text{down}}(s's, a) V_{\downarrow}(s')$;(∗∗)
(21) $\pi(s) = a$;
(22) **end if**
(23) **end if**

Algorithm 2: Interval value iteration (IVI$_{\updownarrow}$) algorithm.



(a) State size versus number of request types, with server capacity = 5 (b) State size versus server capacity, with number of request types = 2

Figure 4: Problem state space size of MDP and BMDP formulations.

exponentially as the number of edge servers increases. This problem can be addressed by further aggregation of the state space of a single server. However, the greater the degree of aggregation is, the more the transition probability information is lost, which may degrade the performance of the generated solution. In practice, the CDN provider must make tradeoff between the quality of the generated solution and the computational cost. We leave this issue in our future works.

5.2. Convergence and Application of the IVI Algorithm. We initially set the interval value function for all states to $[0, 0]$ and other parameters are defined in Tables 2 and 3. The interval value function of state $[\mathbf{0}, \mathbf{0}]$ in each iteration is plotted in Figure 5. It converges after about 600 iterations.

The IVI algorithm is not suitable for online scheduling since it involves time-consuming iterative steps, which cannot cope with variations of online request arrival pattern. However, like the Internet traffic, the Internet video streaming also demonstrates a strong *temporal pattern* [44]. We could divide a day into several parts according to the request arrival patterns obtained by network measurement and compute a particular strategy for each part offline. The dispatcher can adopt the associated strategy in a given time interval to yield close-to-optimal rewards.

5.3. Performance for the Greedy Strategy and the B-Optimal Strategy. We evaluate the greedy strategy and the B-optimal strategy in two ways.

(1) The value function V_{greedy} for the greedy strategy in the MDP model and the interval value function $[V_{\downarrow\text{opt}}, V_{\uparrow\text{opt}}]$ for the B-optimal strategy in the BMDP model, which is plotted in Figure 6.

(2) The ratio of $V_{\downarrow\text{opt}}/V_{\uparrow\text{opt}}$ and $V_{\text{greedy}}/V_{\uparrow\text{opt}}$, which is plotted in Figure 7.

In both Figures 6 and 7 we vary the arrival probability of two request types from 0.01 to 0.2.

The first observation from Figure 6 is that as the arrival traffic grows heavier, the upper bound and the lower bound of the B-optimal strategy, as well as the value of the greedy strategy, all tend to a steady state. This is because the system approaches to a saturated state. Another phenomenon is that the B-optimal strategy always outperforms the greedy strategy since even the lower bound of the B-optimal strategy is greater than the greedy strategy, meaning that the B-optimal algorithm is effective in this system.

We can have a clearer view of the quality of the proposed strategies in Figure 7. The lower bound of performance $V_{\downarrow\text{opt}}$ in the worst case for B-optimal strategy can attain around 80% to over 95% of the upper bound $V_{\uparrow\text{opt}}$. For the greedy strategy, V_{greedy} can attain around 65% to 80% of the upper bound $V_{\uparrow\text{opt}}$. An interesting finding is that there exists a conspicuous "jumping line" which divides $V_{\downarrow\text{opt}}/V_{\uparrow\text{opt}}$ into two parts (a light load part and a heavy load part). The ratio $V_{\downarrow\text{opt}}/V_{\uparrow\text{opt}}$ grows in the light load part, surges to the peak at the jumping line, and begins to decline slowly in the heavy load part, as the request arrival probabilities increase.

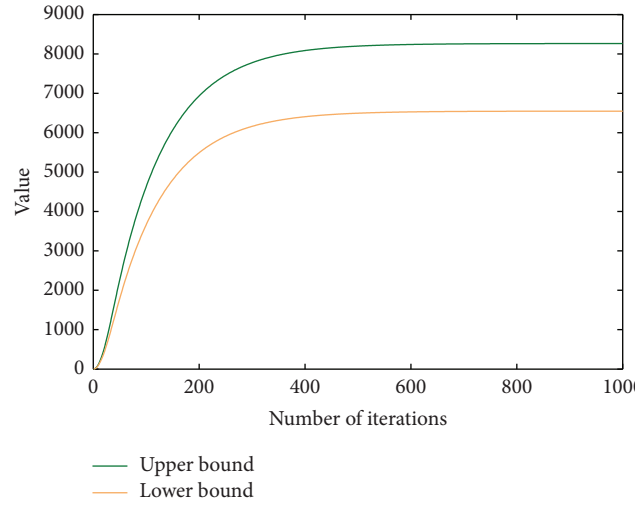


Figure 5: Convergence of IVI algorithm.

Table 2: Parameters setting.

	Type 1 request		Type 2 request	
	Server 1	Server 2	Server 1	Server 2
Cost	2	2.5	1	2
Reward	12		10	
Expected sojourn time	2000		1000	
Upper bound of sojourn time	2500		1200	

Table 3: Arrival and departure distribution.

	Type 1 request		Type 2 request	
Number of arrivals	0	1	0	1
Arrival probability	0.8	0.2	0.9	0.1
Departure probability	0.0025		0.005	

The heavy load part of $V_{\downarrow\text{opt}}/V_{\uparrow\text{opt}}$ still stands beyond 90%, suggesting that the B-optimal strategy has an outstanding performance under heavy load even in the worst case. On the other hand $V_{\text{greedy}}/V_{\uparrow\text{opt}}$ reaches the trough at two positions: (1) extreme light load; (2) nearly the same load region of the jumping line of $V_{\downarrow\text{opt}}/V_{\uparrow\text{opt}}$, indicating that the performance of the greedy strategy may degrade at these load regions. However, in most of cases, $V_{\text{greedy}}/V_{\uparrow\text{opt}}$ remains flat (around 80%).

6. Concluding Remarks

In this paper we consider the dynamic request routing issue in the Video-on-Demand system. We classify video requests by the expected viewing time and the average bitrates of the video files. The system can be abstracted into a controlled queueing system containing one dispatcher with its waiting queue and several VoD edge servers with their service queues. Our goal is to find the decision policy of the dispatcher

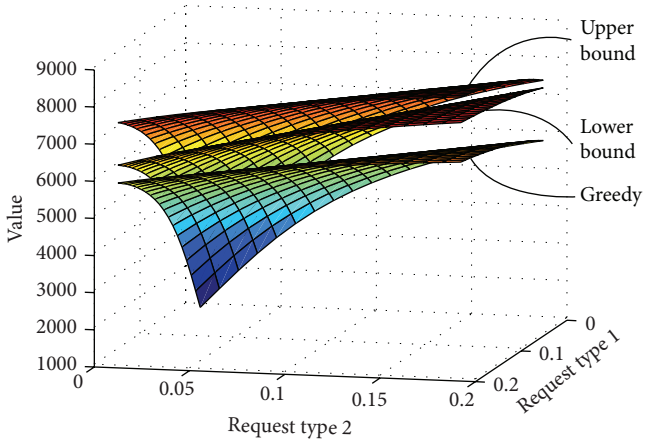


Figure 6: Upper bound and lower bound versus arrival probability.

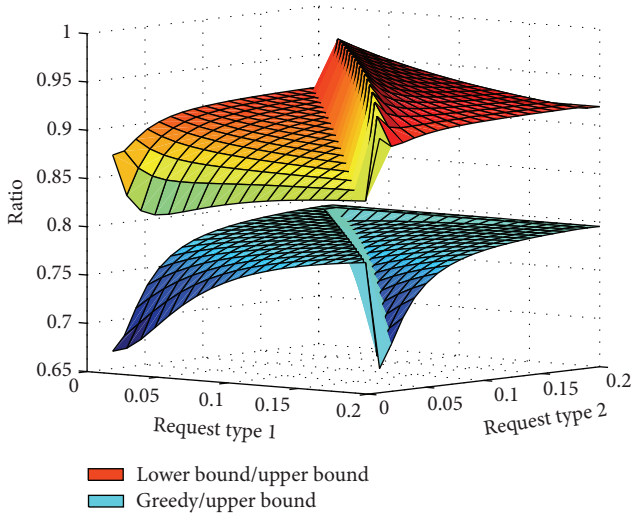


Figure 7: $V_{\downarrow opt}/V_{\uparrow opt}$ = lower bound/upper bound and $V_{greedy}/V_{\uparrow opt}$ = greedy/upper bound versus arrival probability.

which yields the highest reward. The dynamic request routing problem can be formulated as a Markov decision process, and classical iterative algorithm can be used to obtain the optimal solution.

However, the MDP formulation has its intrinsic drawback of the curse of dimensionality, which makes the problem intractable in practical scenario. To address this issue, we present two alternative approaches, that is, the greedy strategy and the bounded-parameter MDP reformulation, to approximately compute the suboptimal solution. These two approximation schemes start from different points: the greedy strategy ignores the request arrival patterns in the future and the BMDP reformulation overlooks the types of request in the server load. Although the greedy strategy is much simpler, the numerical results show that the B-optimal strategy can generate a higher reward than the greedy strategy. Our future research will concentrate on the distributed implementation of dynamic request routing strategy for VoD service.

Notation

I : Set of request types
J : Set of edge servers
$\mathbf{S}$: State space of the original MDP formulation
$\mathbf{S}'$: State space of the BMDP formulation
$\mathbf{N}$: System state matrix, with an element n_{ij} denoting the number of type i requests in edge server j
$\mathbf{L}$: System state vector, with an element L_j denoting the number of requests in edge server j
$\boldsymbol{\lambda}$: Arrival vector, with an element λ_i denoting the number of type i requests arrived at the dispatcher in one slot
$\mathbf{y}$: Request departure matrix, with element y_{ij} denoting the number of departures for type i request in edge server j
$\mathbf{Y}$: Request departure vector, with element Y_j denoting the number of departures in edge server j
Γ : Length of a time slot
T_i : Expected sojourn time in the system for type i request
$\overline{T}_i$: Upper bound of sojourn time in the system for type i request
p_i : Departure probability of type i request
C_j : Bandwidth capacity of edge server j
ω_i : Amount of bandwidth consumed by a type i request
c_{ij} : Costs of edge server j for serving a type i request in one slot
r_i : Rewards for serving a type i request in one slot
a : Action matrix at time t, with an element a_{ij} denoting the number of type i requests forwarded to edge server j in one slot.

Conflict of Interests

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

Acknowledgments

The work is funded in part by the National Natural Science Foundation of China (NSFC) under Grant no. 61363052, the Inner Mongolia Provincial Natural Science Foundation under Grant nos. 2010MS0913, and 2013MS0920 and the Science Research Project for Inner Mongolia College under Grant nos. NJZY14064 and NJZY13120.

References

[1] Akamai, "State of the Internet," http://www.akamai.com.
[2] YouTube, http://www.youtube.com.
[3] Hulu, http://www.hulu.com.

[4] A. Lenk, M. Klems, J. Nimis, S. Tai, and T. Sandholm, "What's inside the cloud? An architectural map of the cloud landscape," in *Proceedings of the ICSE Workshop on Software Engineering Challenges of Cloud Computing (CLOUD '09)*, pp. 23–31, May 2009.

[5] Y. Feng, B. Li, and B. Li, "Airlift: video conferencing as a cloud service using inter-datacenter networks," in *Proceeding of the 20th IEEE International Conference on Network Protocols (ICNP '12)*, pp. 1–11, Austin, Tex, USA, November 2012.

[6] F. Wang, J. Liu, and M. Chen, "CALMS: cloud-assisted live media streaming for globalized demands with time/region diversities," in *Proceeding of the IEEE Conference on Computer Communications (INFOCOM '12)*, pp. 199–207, Orlando, Fla, USA, March 2012.

[7] Y. Li, Y. Shen, and Y. Liu, "Utilizing content delivery network in cloud computing," in *Proceeding of the International Conference on Computational Problem-Solving (ICCP '12)*, pp. 137–143, Leshan, China, October 2012.

[8] H. A. Tran, A. Mellouk, and S. Hoceini, "QoE content distribution network for cloud architecture," in *Proceeding of the 1st IEEE Symposium on Network Cloud Computing and Applications (NCCA '11)*, pp. 14–19, Toulouse, France, November 2011.

[9] Y. Jin, Y. Wen, G. Shi, G. Wang, and A. V. Vasilakos, "CoDaaS: an experimental cloud-centric content delivery platform for user-generated contents," in *Proceeding of the International Conference on Computing, Networking and Communications (ICNC '12)*, pp. 934–938, Maui, Hawaii, USA, February 2012.

[10] L. Chia-Feng, L. Muh-Chy, C. Chih-Wei, and Y. Shyan-Ming, "The study and methods for cloud based CDN," in *Procedding of the 3rd International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC '11)*, pp. 469–475, Sanya, China, October 2011.

[11] F. Chen, K. Guoy, J. Liny, and T. La Porta, "Intra-cloud lightning: building CDNs in the cloud," in *Proceeding of the IEEE INFOCOM*, pp. 433–441, Orlando, Fla, USA, March 2012.

[12] Amazon CloudFront, http://aws.amazon.com/cloudfront/.

[13] Akamai NetStorage, http://www.akamai.com/html/technology/products/netstorage.html.

[14] http://www.limelight.com/.

[15] Z. Zhang, M. Zhang, A. Greenberg, Y. C. Hu, R. Mahajan, and B. Christian, "Optimizing cost and performance in online service provider networks," in *Proceeding of the 7th USENIX Symposium on Networked Systems Design and Implementation (NSDI '10)*, 2010.

[16] M. Andrews, B. Shepherd, A. Srinivasan, P. Winkler, and F. Zane, "Clustering and server selection using passive monitoring," in *Proceeding of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02)*, vol. 3, pp. 1717–1725, New York, NY, USA, June 2002.

[17] O. Ardaiz, F. Freitag, and L. Navarro, " Improving the service time of web clients using server redirection," *ACM SIGMETRICS Performance Evaluation Review*, vol. 29, no. 2, pp. 39–44, 2001.

[18] P. Wendell, J. W. Jiang, M. J. Freedman, and J. Rexford, "DONAR: decentralized server selection for cloud services," in *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '10)*, New Delhi, India, September 2010.

[19] R. Torres, A. Finamore, J. R. Kim, M. Mellia, M. M. Munafo, and S. Rao, "Dissecting video server selection strategies in the YouTube CDN," in *Proceeding of the 31st International Conference on Distributed Computing Systems (ICDCS '11)*, pp. 248–257, Minneapolis, Minn, USA, July 2011.

[20] S. Ao-Jan, D. R. Choffnes, A. Kuzmanovic, and F. E. Bustamante, "Drafting behind Akamai (travelocitybased detouring)," in *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '06)*, pp. 435–446, Pisa, Italy, September 2006.

[21] N. Carlsson and D. L. Eager, "Server selection in large-scale video-on-demand systems," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 6, no. 1, article 1, 2010.

[22] L. Liu and Y. Lu, "Dynamic traffic controls for web-server networks," *Computer Networks*, vol. 45, no. 4, pp. 523–536, 2004.

[23] C. Lin, Y. Jun, P. Jianping, S. Xuemin (Sherman), and J. W. Mark, "Dynamic server selection using fuzzy inference in content distribution networks," *Computer Communications*, vol. 29, no. 8, pp. 1026–1038, 2006.

[24] Z. Fei, M. H. Ammar, and E. W. Zegura, "Optimal allocation of clients to replicated multicast servers," in *Proceedings of the 7th International Conference on Network Protocols (ICNP '99)*, pp. 69–76, October 1999.

[25] H. Xu and B. Li, "Joint request mapping and response routing for geo-distributed cloud services," in *Proceedings of IEEE INFOCOM*, pp. 854–862, Turin, Italy, April 2013.

[26] H. Xu and B. Li, "A general and practical datacenter selection framework for cloud services," in *Proceeding of the IEEE 5th International Conference on Cloud Computing (CLOUD '12)*, pp. 9–16, Honolulu, Hawaii, USA, June 2012.

[27] H. A. Tran, A. Mellouk, J. Perez, S. Hoceini, and S. Zeadally, "QoE-based server selection for content distribution networks," *IEEE Transactions on Computers*, 2013.

[28] H. Qian and M. Rabinovich, "Application placement and demand distribution in a global elastic cloud: a unified approach," in *Proceeding of the 10th International Conference on Autonomic Computing (ICAC '13)*, San Jose, Calif, USA, June 2013.

[29] M. Hajjat, P. N. Shankaranarayanan, and D. Maltz, "Dealer: application-aware request splitting for interactive cloud applications," in *Proceeding of the 8th ACM International Conference on Emerging Networking Experiments and Technologies (CoNEXT '12)*, pp. 157–168, Nice, France, December 2012.

[30] C. Ding, Y. Chen, T. Xu, and X. Fu, "CloudGPS: a scalable and ISP-friendly server selection scheme in cloud computing environments," in *Proceeding of the IEEE 20th International Workshop on Quality of Service (IWQoS '12)*, Coimbra, Portugal, June 2012.

[31] B. Frank, I. Poese, Y. Lin et al., "Pushing CDN-ISP collaboration to the Limit," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 3, pp. 34–44, 2013.

[32] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, Wiley-Interscience, New York, NY, USA, 1994.

[33] Z. Liu, M. S. Squillante, and J. L. Wolf, "On maximizing service-level-agreement profits," in *Proceedings of the 3rd ACM Conference on Electronic Commerce (EC '01)*, pp. 223–213, Tampa, Fla, USA, October 2001.

[34] L. Zhang and D. Ardagna, "SLA based profit optimization in autonomic computing systems," in *Proceedings of the 2nd International Conference on Service Oriented Computing (ICSOC '04)*, pp. 173–182, New York, NY, USA, November 2004.

[35] E. Casalicchio and M. Colajanni, "A client-aware dispatching algorithm for web clusters providing multiple services," in *Proceedings of the 10th ACM International Conference on World Wide Web (WWW '01)*, pp. 535–544, Hong Kong, May 2001.

[36] M. Colajanni and P. S. Yu, "Adaptive TTL schemes for load balancing of distributed web servers," *ACM SIGMETRICS Performance Evaluation Review*, vol. 25, pp. 36–42, 1997.

[37] M. Colajanni, P. S. Yu, and V. Cardellini, "Dynamic load balancing in geographically distributed heterogeneous web servers," in *Proceedings of the 18th International Conference on Distributed Computing Systems (ICDCS '98)*, pp. 295–302, Amsterdam, The Netherlands, May 1998.

[38] M. Conti, E. Gregori, and F. Panzieri, "Load distribution among replicated web servers: a QoS-based approach," *ACM SIGMETRICS Performance Evaluation Review*, vol. 27, no. 4, pp. 12–19, 2000.

[39] J. Cao, Y. Sun, X. Wang, and S. K. Das, "Scalable load balancing on distributed web servers using mobile agents," *Journal of Parallel and Distributed Computing*, vol. 63, no. 10, pp. 996–1005, 2003.

[40] G. Ciardo, A. Riska, and E. Smirni, "EquiLoad: a load balancing policy for clustered web servers," *Performance Evaluation*, vol. 46, no. 2-3, pp. 101–124, 2001.

[41] R. Givan, S. Leach, and T. Dean, "Bounded-parameter Markov decision processes," *Artificial Intelligence*, vol. 122, no. 1, pp. 71–109, 2000.

[42] R. Givan, S. Leach, and T. Dean, "Bounded parameter Markov decision processes," in *Computer Science*, S. Steel and R. Alami, Eds., vol. 1348 of *Lecture Notes*, pp. 234–246, Springer, Berlin, Germany, 1997.

[43] F. Thouin and M. Coates, "Video-on-demand networks: design approaches and future challenges," *IEEE Network*, vol. 21, no. 2, pp. 42–48, 2007.

[44] H. Yu, D. Zheng, B. Y. Zhao, and W. Zheng, "Understanding user behavior in large-scale video-on-demand systems," in *Proceedings of the 1st ACM SIGOPS/EuroSys European Conference on Computer Systems (EuroSys '06)*, pp. 333–344, Leuven, Belgium, April 2006.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 874543, 6 pages
http://dx.doi.org/10.1155/2014/874543

Research Article

Formation Control of Multirobot Based on I/O Feedback Linearization and Potential Function

Jie Dong, Sen Liu, and Kaixiang Peng

Key Laboratory of Advanced Control of Iron and Steel Process of Ministry of Education, School of Automation and Electrical Engineering, University of Science and Technology, Beijing 100083, China

Correspondence should be addressed to Kaixiang Peng; kaixiang@ustb.edu.cn

Received 24 January 2014; Revised 30 March 2014; Accepted 3 May 2014; Published 14 May 2014

Academic Editor: Housheng Su

Standard techniques of I/O linearization are widely applied to leader-follower approach for multirobot formation control. However general leader-follower approach cannot adapt to the environment with obstacles. Concerning that issue, a formation control method of multirobot system based on potential function is proposed in this paper, and a new control law is designed by choosing a proper potential function and employing Lyapunov stability theory, which stabilizes the formation of the multirobot system. We combine the method with a leader-follower approach to solve the problem that the latter cannot avoid obstacles. Simulation results are given to validate the method.

1. Introduction

Multirobot formation control is a common cooperation problem of multirobot system. In recent years, as one of the basic coordination and cooperation issues, it has been gradually applied to military reconnaissance, search, demining, aircraft formation flying, space exploration, and many other fields and attracted a large number of researchers. It has become a very active research direction of multirobot system [1]. At present, formation control methods mainly include leader-follower approach [2–5], artificial potential field approach [6–9], behavior-based approach [10], and virtual structural approach [11]. To achieve a reasonable and efficient formation, the key is to use a proper control method that meets the system requirements. No matter what kind of control method is used, the stability of the system must be analyzed. Lyapunov stability theory [12–14] and graph theory [15, 16] are usually used in stability analysis, and these control theory researches contribute to the improvements of multirobot system. Lyapunov method plays a very important role in a multirobot system's stability analysis and controller design. In [17], the idea of relative-position-based formation stability was proposed and the Lyapunov method was also used to design the decentralized controllers, along with an extended linear matrix inequality (LMI) to analyze the conditions required for formation stability. In [18], Lyapunov method was used to analyze the stability of multiagent coordination control. For each individual robot, there is a control Lyapunov function, and the Lyapunov function of the system is the weighted sum of all the agents' Lyapunov functions. However, that paper presents a method based on Lyapunov stability theory, which gives a stable formation control law of the multirobot system by selecting an appropriate artificial potential field function. We combine this method with $l - \varphi$ control based on input-output feedback linearization [19] to solve the problem that $l - \varphi$ control cannot adapt to obstacles environment.

2. Mathematical Model

In this section, the kinematics and dynamics of mobile robots are described in Figure 1. The robot consists of two driving rear wheels and a supporting roller, and the nonholonomic constraint that the driving wheels allow only pure rolling without slipping is satisfied. The kinematic and dynamic equations of motion of the ith individual robot are given as

$$\dot{x}_i = v_i \cos\theta_i,$$

$$\dot{y}_i = v_i \sin\theta_i,$$

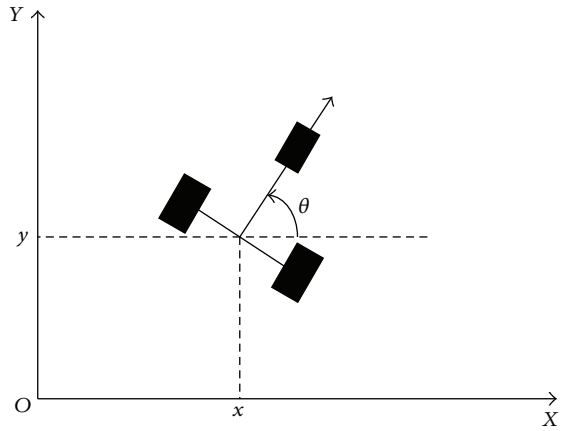


Figure 1: Mathematical model of the robot.

$$\begin{aligned}\dot{\theta}_i &= \omega_i,\\ \dot{v}_i &= \alpha_i,\\ \dot{\omega}_i &= \beta_i,\end{aligned} \tag{1}$$

where the position of the robot is given by $(x_i, y_i)^T$. θ_i is the heading direction of the robot in the global coordinates. $q_i = (x_i, y_i, \theta_i)^T$ is the pose vector of robot i. v_i and ω_i are the linear and angular velocities at the center of the axle of each robot. $u_i = (\alpha_i, \beta_i)^T$ is control input (acceleration).

3. Control of Formations

3.1. $l - \varphi$ Control Based on Input-Output Feedback Linearization. $l - \varphi$ control is a kind of leader-follower approach. A group of two nonholonomic mobile robots is shown in Figure 2. The aim of the controller is to make the follower (Robot2) follow the leader (Robot1) by keeping the relative distance l_{12} and the relative orientation φ_{12} constant. As a result, both the values can be controlled to reach the desired value $(l_{12}^d, \varphi_{12}^d)$ and maintain the desired formation.

We introduce $\gamma_1 = \theta_1 + \varphi_{12} - \theta_2$; according to the geometrical relationship, the differential equation of l_{12} and φ_{12} is given by

$$\begin{aligned}\dot{l}_{12} &= v_2 \cos\gamma_1 - v_1 \cos\varphi_{12} + d\omega_2 \sin\gamma_1,\\ \dot{\varphi}_{12} &= \frac{1}{l_{12}}\left(v_1 \sin\varphi_{12} - v_2 \sin\gamma_1 + d\omega_2 \cos\gamma_1 - l_{12}\omega_1\right),\\ \dot{\theta}_{12} &= \omega_2,\end{aligned} \tag{2}$$

where v_i and ω_i $(i = 1, 2)$ are the linear and angular velocities at the center of the axle of each robot. The distance between the supporting roller and the center of the axis of the driving wheels of each robot is denoted by d. In order to avoid collisions between robots, we will require that $l_{12} > d$.

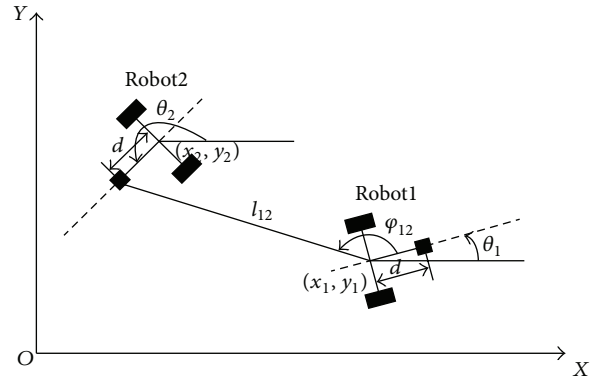


Figure 2: Notation for $l - \varphi$ control.

We use standard techniques of I/O linearization to generate a control law [12], which is given by

$$\begin{aligned}\omega_2 = \frac{\cos\gamma_1}{d}\Big\{&a_2 l_{12}\left(\varphi_{12}^d - \varphi_{12}\right)\\ &-v_1 \sin\varphi_{12} + l_{12}\omega_1 + \rho_{12}\sin\gamma_1\Big\},\\ v_2 = \rho_{12} - d\omega_2 \tan\gamma_1,\end{aligned} \tag{3}$$

where

$$\rho_{12} = \frac{a_1\left(l_{12}^d - l_{12}\right) + v_1\cos\varphi_{12}}{\cos\gamma_1}. \tag{4}$$

That leads to dynamics in the $l - \varphi$ variables of the form

$$\begin{aligned}\dot{l}_{12} &= a_1\left(l_{12}^d - l_{12}\right),\\ \dot{\varphi}_{12} &= a_2\left(\varphi_{12}^d - \varphi_{12}\right),\end{aligned} \tag{5}$$

where a_1 and a_2 are constants. By selecting appropriate values, the relative distance and angle between the robots can converge to $(l_{12}^d, \varphi_{12}^d)$, and the system becomes stable.

3.2. Control Based on Artificial Potential Field Function. An artificial potential field approach is a virtual force method proposed by Khatib. Its basic idea is that the robots move in a virtual force field. Obstacles are surrounded by repulsive potential field, which produces a repulsive force that decreases rapidly with the closer distance between the robot and the obstacle. The target is surrounded by attractive potential field, which produces an attractive force that decreases as the robot is close to the target. Under the role of the resultant force, the robots move along the direction of minimizing potential energy. For multirobot formation control, we need to design a suitable potential function to maintain the formation while reaching the target and avoiding obstacles. Potential function is defined as follows.

Definition 1 (potential function). Potential function V_{ij} is a differentiable, nonnegative, and unbounded function of

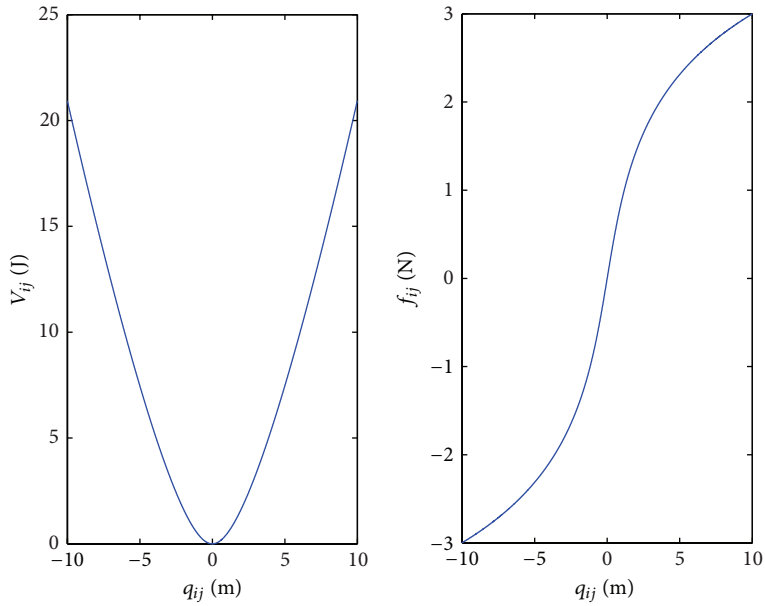


Figure 3: Potential energy and force.

the distance $\|r_{ij}\|$ between robot i and j, and the following conditions are satisfied.

(1) When $\|r_{ij}\| \to \infty$, $V_{ij}(\|r_{ij}\|) \to \infty$.

(2) When the distance $\|r_{ij}\|$ achieves the desired distance, that is, $\|r_{ij}\| = d_{ij}$, V_{ij} gets the only minimum value.

The potential function of neighbor robot i and j is chosen as follows:

$$V_{ij}\left(q_{ij}\right) = K_{ij}\left(q_{ij}\ln\left(q_{ij} + \sqrt{q_{ij}^2 + 1}\right) - \sqrt{q_{ij}^2 + 1}\right) + 1, \quad (6)$$

where $q_{ij} = \|r_{ij}\| - d_{ij}$, K_{ij} is a positive constant, and then the force is the negative gradient of the potential energy:

$$f_{ij}\left(r_i, r_j\right) = -\text{grad}\left(V_{ij}\right) = K_{ij}\text{arsh}\left(q_{ij}\right)\frac{r_j - r_i}{\left\|r_{ij}\right\|}, \quad (7)$$

where r_i, r_j are the position vectors of robots i and j.

The potential energy V_{ij} and force f_{ij} described by (6) and (7) are shown in Figure 3. It indicates that when $q_{ij} > 0$, that is, $\|r_{ij}\| > d_{ij}$, $f_{ij} > 0$, the robots are attractive, when $q_{ij} < 0$, that is, $\|r_{ij}\| < d_{ij}$, $f_{ij} < 0$, the robots are repulsive, and when $q_{ij} = 0$, that is, $\|r_{ij}\| = d_{ij}$, $f_{ij} = 0$, the force between the robots achieves balance, and V_{ij} is minimized.

4. Control Based on I/O Feedback Linearization and Potential Function

4.1. Design of a Stable Formation Control Law. Considering a system of n robots, the dynamic of each robot is described by (1). The total potential energy and force acting on robot i contributed from all the other robots are given as

$$V_i\left(r_i\right) = \sum_{j\neq i} V_{ij}\left(q_{ij}\right), \quad (8)$$

$$F_i\left(r_i\right) = \sum_{j\neq i} f_{ij}\left(r_i, r_j\right). \quad (9)$$

When the formation moves toward the target in an environment with obstacles, $F_i(r_i)$ also includes the attractive force of the target and the repulsive force of the obstacles. The magnitude of $F_i(r_i)$ projected in the heading direction of the robot is used to control the translational acceleration of the robot. The angle is used as the desired orientation for the heading direction of the robot. Let ϕ_i be the direction of the resultant force of robot i, and the orientation error can be described as

$$\begin{aligned} e_i &= \theta_i - \phi_i, \\ \dot{e}_i &= \dot{\theta}_i - \dot{\phi}_i, \\ \ddot{e}_i &= \ddot{\theta}_i - \ddot{\phi}_i. \end{aligned} \quad (10)$$

Theorem 2. *For a multirobot system described by* (1)*, the potential force shown by* (9) *acts on each robot, and the angle of the resultant force is defined as the desired orientation of them. Then, the following control law stabilizes the formation of the system:*

$$\begin{aligned} \alpha_i &= -\sqrt{f_{ix}^2 + f_{iy}^2} - k_1 v_i, \\ \beta_i &= -k_2\left(\theta_i - \phi_i\right) - k_3\left(\dot{\theta}_i - \dot{\phi}_i\right) + \ddot{\phi}_i, \end{aligned} \quad (11)$$

where k_1, k_2, k_3 are positive constants and f_{ix}, f_{iy} are the components of $F_i(r_i)$ in the x and y directions, respectively.

Proof. (1) $v_i \geq 0$. Consider the following Lyapunov candidate:

$$W = \sum_{i=1}^{n} V_i + \frac{1}{2}\sum_{i=1}^{n} k_2 e_i^2 + \frac{1}{2}\sum_{i=1}^{n} \dot{e}_i^2 + \frac{1}{2}\sum_{i=1}^{n} v_i^2. \quad (12)$$

The time derivative of W is computed as

$$\dot{W} = \sum_{i=1}^{n} \left\{ \left(\nabla_{r_i} V_i\right)^T \dot{r}_i + k_2 e_i \dot{e}_i + \dot{e}_i \ddot{e}_i + v_i \dot{v}_i \right\}. \quad (13)$$

Insert (11) into (13):

$$\begin{aligned} \dot{W} &= \sum_{i=1}^{n} \Big\{ v_i f_{ix} \cos\theta_i + v_i f_{iy} \sin\theta_i \\ &\qquad - k_3 \dot{e}_i^2 - v_i \sqrt{f_{ix}^2 + f_{iy}^2} - k_1 v_i^2 \Big\} \quad (14) \\ &= \sum_{i=1}^{n} \left\{ v_i \left|F_i\right| \left(\cos\left(\theta_i - \phi_i\right) - 1\right) - k_3 \dot{e}_i^2 - k_1 v_i^2 \right\}, \end{aligned}$$

where $|F_i| = \sqrt{f_{ix}^2 + f_{iy}^2}$, due to $\cos(\theta_i - \phi_i) \leq 1$, so $\dot{W} \leq 0$.

(2) $v_i < 0$. Consider the following Lyapunov candidate:

$$W = 2\sum_{i=1}^{n} V_i + \frac{1}{2}\sum_{i=1}^{n} k_2 e_i^2 + \frac{1}{2}\sum_{i=1}^{n} \dot{e}_i^2 + \frac{1}{2}\sum_{i=1}^{n} v_i^2. \quad (15)$$

Insert (11) into (15), and then calculate the derivative of W:

$$\dot{W} = \sum_{i=1}^{n} \left\{ v_i \left|F_i\right| \left(2\cos\left(\theta_i - \phi_i\right) - 1\right) - k_3 \dot{e}_i^2 - k_1 v_i^2 \right\}. \quad (16)$$

From (10),

$$\begin{aligned} e_i = \Bigg\{ & c_1 \cos\left(\frac{\sqrt{4k_2 - k_3^2}}{2} t \right) \\ & + c_2 \sin\left(\frac{\sqrt{4k_2 - k_3^2}}{2} t \right) \Bigg\} e^{-(k_3/2)t}. \end{aligned} \quad (17)$$

Let $c_1 = c_2 = 0.5$, and then $|e_i| < 1$, that is, $2\cos(\theta_i - \phi_i) - 1 = 2\cos e_i - 1 > 0$, so $\dot{W} \leq 0$.

Consequently, the stability of the formation is proved with Lyapunov stability theory. □

4.2. Combination Control Method. $l - \varphi$ control based on I/O feedback linearization is simple, that we just control the follower robot to follow the trajectory of the leader robot. In this way, the formation control can be simplified to an independent tracking problem. Each robot only needs to obtain the status information of its leader, which greatly simplifies the issue of cooperation among robots. However, this feedback control cannot avoid obstacles, while the artificial potential field approach is effective in obstacle avoidance. For this reason, if we combine those two methods, the problem will be solved.

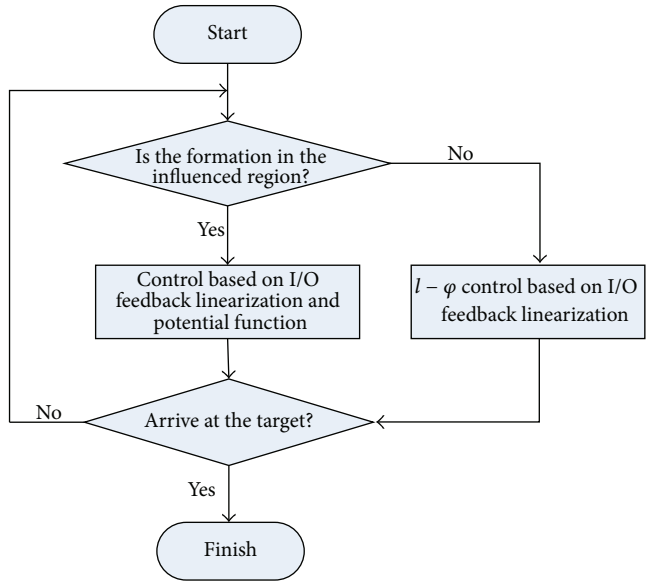


Figure 4: The algorithm flow chart of combination method.

The strategy is as follows. The leader adopts the artificial potential field approach for path planning to reach the target while avoiding obstacles. The follower uses I/O feedback linearization combination with potential function to maintain formation in the obstacles environment; that is, when the follower is in the region influenced by obstacles, the combination method is applied, and the control performance can be improved by adjusting the weight, and then when out of the influenced region, $l - \varphi$ control based on I/O feedback linearization is only used. The algorithm flow chart is shown in Figure 4.

5. Simulation

In this section, computer simulation is used to verify the previous conclusions. For a group of 3 robots, the robots maintain triangular formation to reach the target while avoiding obstacles, which were idealized as circular objects just as described in [20].

(1) Formation Control Based on Potential Function. The initial positions of the robots are $[-2\ \ 0]$, $[0\ \ 3]$, and $[0\ \ 0]$. The position of the target is $[25\ \ 20]$. The obstacle is a circle of radius 1 centered at $[7\ \ 9]$. Under the role of the potential energy described by (6) and applying the control law (11), the robots form an equilateral triangle formation with the side length of 3, avoid the obstacle, keep the same speed and direction, and reach the target at last just as shown in Figure 5. Figure 6 illustrates the change of errors e_i. It is observed that the formation is stable.

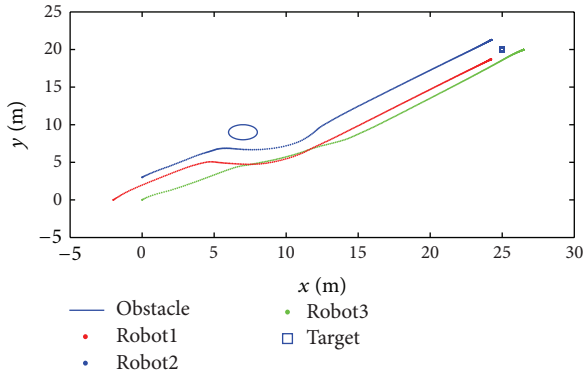


Figure 5: Formation control based on potential function.

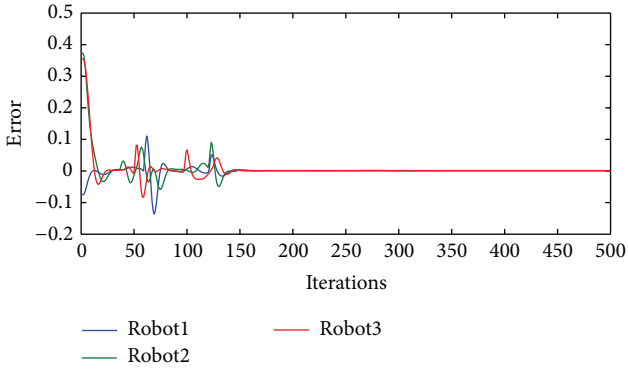


Figure 6: The orientation error of robots.

(2) Control Based on $l-\varphi$ and Potential Function. The initial positions of the robots are $[0\ \ 1]$, $[-3\ \ -2]$, and $[-2\ \ 0]$. The position of the target is $[20\ \ 16]$. The obstacle is a circle of radius 1 centered at $[9\ \ 7]$. Robot1 is the leader, and Robot2, Robot3 are the followers keeping a distance of 2, angles of $\pi/3$, $-\pi/3$, respectively, from Robot1. When the formation enters into the influenced region of the obstacle, use the approach of control based on $l-\varphi$ and potential function. Otherwise, $l-\varphi$ control based on I/O feedback linearization is applied. Figure 7 illustrates that single $l-\varphi$ control cannot avoid obstacles, while by using the proposed combination control method, the formation can successfully round the obstacle and move to the target keeping a stable formation, as shown in Figure 8.

6. Conclusions

Formation control of multirobot system was studied in the paper. A new artificial potential field function and corresponding formation controller were designed, and the stability analysis was given. In addition, we proposed a combination control method based on I/O feedback linearization and potential function, of which feature is choosing the appropriate control strategy according to obstacles environment, solving the problem that $l-\varphi$ control cannot avoid obstacles.

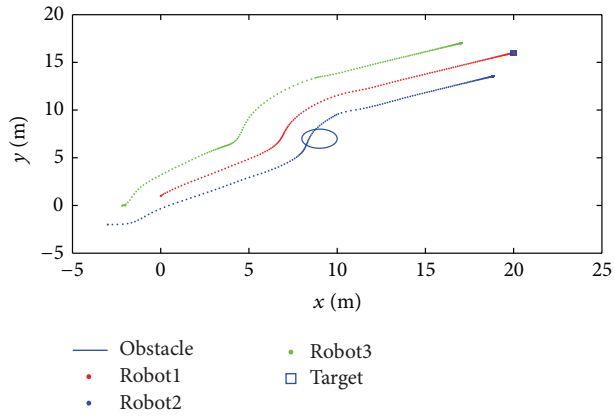


Figure 7: $l-\varphi$ control cannot avoid obstacles.

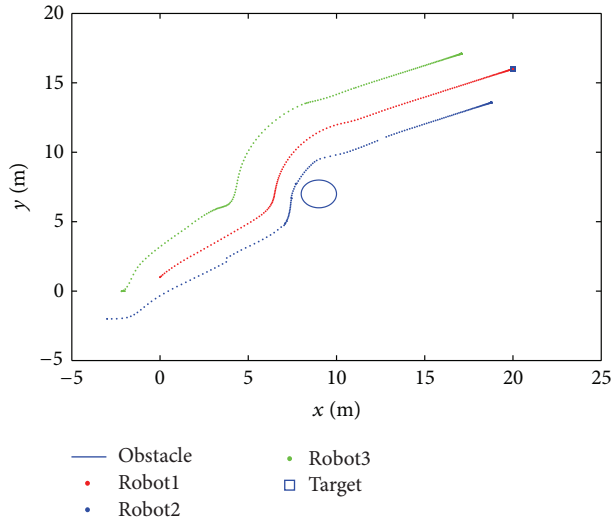


Figure 8: Control based on $l-\varphi$ and potential function.

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 Natural Science Foundation of China (NSFC) under Grant no. 61074085 and by Beijing Natural Science Foundation (4142035), China.

References

[1] L. Lei, *Formation and Coordination Control of Multiple Mobile Robots*, Huazhong University of Science and Technology, Wuhan, China, 2009.

[2] X. Chen, A. Serrani, and H. Ozbay, "Control of leader-follower formations of terrestrial UAVs," in *Proceedings of the 42nd IEEE Conference on Decision and Control*, vol. 1, pp. 498–503, December 2003.

[3] K. H. Kowdiki, R. K. Barai, and S. Bhattacharya, "Leader-follower formation control using artificial potential function: a kinematic approach," in *Proceeding of the IEEE-International Conference on Advances in Engineering, Science and Management*, pp. 500–505, Tamil Nadu, India, March 2012.

[4] 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.

[5] H. Su, X. Wang, and G. Chen, "A connectivity-preserving flocking algorithm for multi-agent systems based only on position measurements," *International Journal of Control*, vol. 82, no. 7, pp. 1334–1343, 2009.

[6] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *The International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986.

[7] J. Sfeir, M. Saad, and H. Saliah-Hassane, "An improved artificial potential field approach to real-time mobile robot path planning in an unknown environment," in *Proceeding of the 9th IEEE International Symposium on Robotic and Sensors Environments*, pp. 208–213, Montreal, Canada, September 2011.

[8] R. Olfati-Saber and R. M. Murray, "Distributed cooperative control of multiple vehicle formations using structural potential functions," in *Proceeding of the IFAC World Congress*, pp. 346–352, 2002.

[9] R. Olfati-Saber, "Flocking with obstacle avoidance," Tech. Rep. CIT-CDS03-006, California Institute of Technology, Control and Dynamical Systems, Pasadena, Calif, USA, 2003.

[10] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 926–939, 1998.

[11] W. Ren and R. W. Beard, "Decentralized scheme for spacecraft formation flying via the virtual structure approach," *Journal of Guidance, Control and Dynamics*, vol. 27, no. 1, pp. 73–82, 2004.

[12] H. Su, N. Zhang, M. Z. Q. 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.

[13] A. Yang, W. Naeem, G. W. Irwin, and K. Li, "Stability analysis and implementation of a decentralized formation control strategy for unmanned vehicles technology," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 2, pp. 706–720, 2014.

[14] X. Luo, N. Han, and X. Guan, "Leader-following consensus protocols for formation control of multi-agent network," *Journal of Systems Engineering and Electronics*, vol. 22, no. 6, pp. 991–997, 2011.

[15] 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.

[16] M. M. Zavlanos, M. B. Egerstedt, and G. J. Pappas, "Graph-theoretic connectivity control of mobile robot networks," *Proceedings of the IEEE*, vol. 99, no. 9, pp. 1525–1540, 2011.

[17] D. Xue, J. Yao, G. Chen, and Y. L. Yu, "Formation control of networked multi-agent systems," *IET Control Theory and Applications*, vol. 4, no. 10, pp. 2168–2176, 2010.

[18] P. Ogren, M. Egerstedt, and X. Hu, "A control Lyapunov function approach to multiagent coordination," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 847–851, 2002.

[19] J. P. Desai, J. Ostrowski, and V. Kumar, "Controlling formations of multiple mobile robots," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 4, pp. 2864–2869, Leuven, Belgium, May 1998.

[20] Y. Liang and H. Lee, "Decentralized formation control and obstacle avoidance for multiple robots with nonholonomic constraints," in *Proceedings of the American Control Conference*, pp. 5596–5601, Minneapolis, Minn, USA, June 2006.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 803743, 11 pages
http://dx.doi.org/10.1155/2014/803743

Research Article

Moving Target Positioning Based on a Distributed Camera Network

Long Zhao,[1,2] Zhen Liu,[1] Tiejun Li,[1] Baoqi Huang,[2] and Lihua Xie[2]

[1] *School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China*
[2] *School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798*

Correspondence should be addressed to Long Zhao; buaa_dnc@buaa.edu.cn

Received 14 February 2014; Accepted 18 April 2014; Published 13 May 2014

Academic Editor: Guoqiang Hu

We propose a systematic framework for moving target positioning based on a distributed camera network. In the proposed framework, low-cost static cameras are deployed to cover a large region, moving targets are detected and then tracked using corresponding algorithms, target positions are estimated by making use of the geometrical relationships among those cameras after calibrating those cameras, and finally, for each target, its position estimates obtained from different cameras are unified into the world coordinate system. This system can function as complementary positioning information sources to realize moving target positioning in indoor or outdoor environments when global navigation satellite system (GNSS) signals are unavailable. The experiments are carried out using practical indoor and outdoor environment data, and the experimental results show that the systematic framework and inclusive algorithms are both effective and efficient.

1. Introduction

The theory of navigation and positioning has been used in various application fields such as positioning equipment to monitor its working state and positioning a car or people to guide them to a certain place. In these applications, it is required that targets are positioned and tracked, which can be realized using the global navigation satellite system (GNSS) or GNSS aided inertial navigation system (INS). However, the GNSS system is subject to various limitations, the most critical one of which is jamming. GNSS signals are not always available due to the blockage of high buildings, canyons, and forests, among others. For this reason, a number of alternative technologies, including optical [1], radio [2–4], RFID [5], and acoustic [6], have been proposed for indoor and outdoor positioning systems over the years. Most efforts were focused on WiFi based localization which takes the advantage of WiFi infrastructures. By using the user's smartphone to measure the signal strength from multiple WiFi access points, the user's location can be constrained within a relatively small region in a large indoor environment. However, these systems rely on prepared infrastructures installed beforehand, and their accuracy critically depends on the number of available access points, which results in certain restrictions [5]. In addition, many results related to the application of the camera based positioning systems have been reported in the last few years [1], including simultaneous localization and mapping (SLAM) [7] and visual odometry [8]. Although SLAM is becoming a standard technique for indoor robotic applications, it is still challenging to apply SLAM in large outdoor environments.

In recent years, the size of camera networks grows quickly with the development of building safe and smart city. These cameras can provide complementary positioning information for moving target positioning in both indoor and outdoor environments when GNSS signals are unavailable. In this paper, we will deal with the moving target positioning based on a distributed camera network in a three-dimensional space. In existing camera networks, static cameras are generally deployed to cover a large region, and moving targets can only be detected and tracked via certain algorithms running in a central supervisor unit, but their positions are not determined [9–12]. In some practical applications, the total number of cameras is usually restricted by various factors such as the cost and placement of cameras. To address this problem, multiple pan, tilt, zoom (PTZ)

cameras or the combination of PTZ cameras and static cameras can be deployed to fulfill some practical tasks [13–18].

The target detection, target tracking, and camera calibration are key to the moving target positioning process. To extract moving targets from a video frame of a static camera, background subtraction is the most widely used approach [19, 20]. When the camera is stationary, the background scene is unchanging, such that it is convenient to construct a background model [21, 22]. The capability of efficiently and accurately estimating background images is critical for any robust background subtraction algorithm. A well-known method presented by Stauffer and Grimson [21] uses an adaptive strategy for modeling background. Therein, each pixel is modeled using a separate Gaussian mixture, which is continuously learnt based on online approximations. Target detection at the current frame is then performed at pixel level by comparing its value against the most probable background Gaussians. However, the adaptive Gaussian mixture algorithm suffers from a low convergence speed in the learning process, especially in complicated environments. For this reason, an improved adaptive Gaussian mixture learning algorithm was introduced in [23].

Moving target tracking is an important component in the field of computer vision and has been widely used in many applications, such as video surveillance [16], intelligent transportation [24], and multiagent systems tracking and control [25]. Target tracking aims to estimate the position and the shape of a target or a region in subsequent frames. During target tracking, a target is continuously tracked by correctly associating a target detected in subsequent frames with the same identified track. These methods and their variations commonly make use of the one-to-one assumption in the sense that a target can only generate at most one measurement in each frame and a measurement can only originate from at most one target. However, the one-to-one assumption rarely holds in practical applications due to the splitting and merging processes as well as multitargets existing in a common scene. In order to overcome these shortcomings, several approaches have been proposed for multitargets tracking in recent years [26–29].

Camera calibration is an essential procedure in distributed multitarget positioning and determines the mapping between 3D world coordinates and 2D image coordinates in practical applications. The basic task of camera calibration is to compute the camera extrinsic and intrinsic parameters which determine the imaging model and the relationship between multiple camera coordinates. With respect to different applications, the corresponding calibration algorithms include the direct linear transformation (DLT) algorithm [9], Tsai algorithm [10], vanishing point algorithm [11], and Zhang algorithm [12]. These algorithms have respective advantages and disadvantages in various practical applications. In this paper, we will focus on a fast calibration algorithm based on the vanishing point theory, which overcomes the defects of traditional measurements.

The paper is organized as follows. The system framework is presented in Section 2. Section 3 focuses on the target detection and tracking. The fast calibration algorithm is presented in Section 4. The test results of target positioning based on a distributed camera network are reported in Section 5. Finally, we draw some conclusions and shed light on future work in Section 6.

2. Systematic Framework and Problem

The work presented in this paper originates from a research project of moving target tracking and positioning in the Digital Navigation Center (DNC) at Beihang University. The primary goal of the project is to develop a target positioning platform to realize monitoring and positioning targets in a large region. The systematic framework of the moving target positioning based on a distributed camera network is shown in Figure 1. Due to the field of view and price limitations, a mass of static cameras are installed in a practical application environment. In order to realize moving target positioning in a large region, it is necessary that the system supports targets detection and tracking. Since a target can no longer be detected, because either it leaves the field of view, it stops and becomes static, or it can no longer be distinguished from the background, it is reasonable to take target splitting and merging into account or detect multiple targets. Therefore, the performance of the target detection, tracking, and association algorithms will influence the reliability of the target positioning, and it is necessary that the target positioning results between cameras are fused into a world coordinate system $X_wY_wZ_w$.

In this paper, we tackle several problems including the targets detection, tracking, and association, as well as the fast camera calibration and target positioning in the moving target positioning system based on a distributed camera network in practical applications.

3. Target Detection and Tracking

3.1. Target Detection. Target detection is the basis of target tracking, target positioning, target recognition, action recognition, and so forth. There are some common algorithms such as the optical flow algorithm [30], the frame difference algorithm [31], and the background subtraction algorithm [32] in practical applications. The most well-known and the most widely used one is background subtraction for static cameras because it is convenient to construct a background model and extract moving targets.

The background modeling techniques can be divided into two categories: the parametric techniques that use a parametric model for each pixel location and the samples-based techniques that build their model by aggregating previously observed values for each pixel location [33]. The most popular parametric technique is based on the Gaussian mixture model (GMM) presented by Stauffer and Grimson [21]. This algorithm relies on the principle that the pixel value for the same location in the image sequences satisfies a Gaussian distribution, as illustrated in Figure 2.

While updating the background model, each pixel of a scene image is independently modelled by a mixture of at most K Gaussian distributions and employs an adaptive

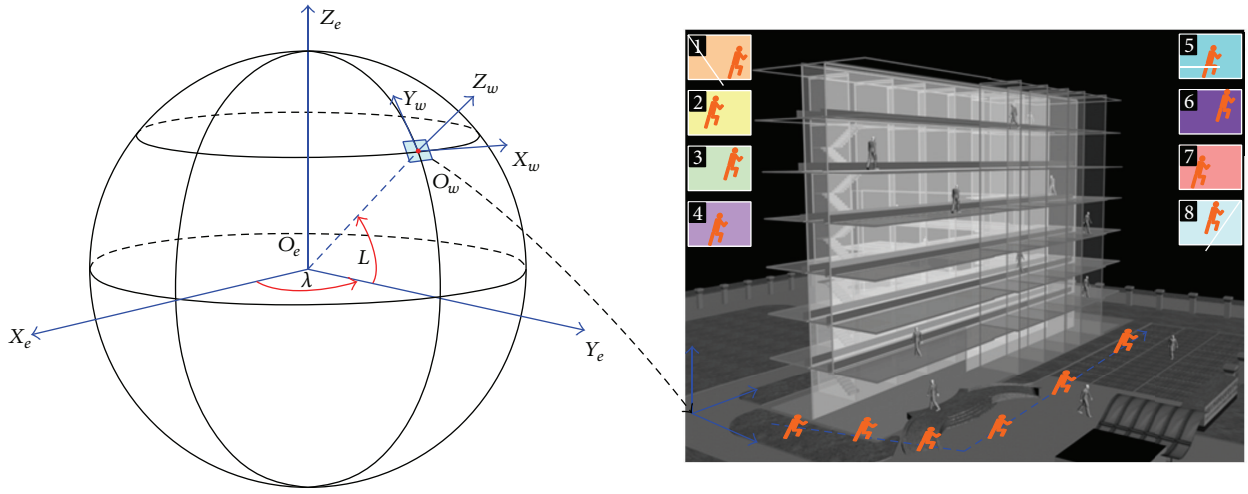


Figure 1: The systematic framework of the moving target positioning based on a distributed camera network.

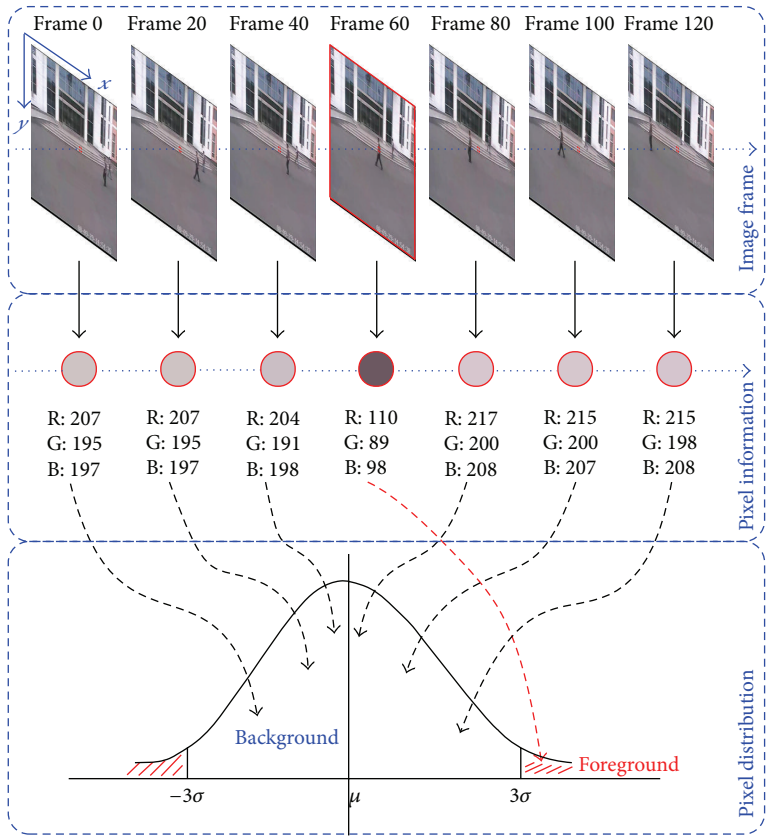


Figure 2: The model of pixel distribution.

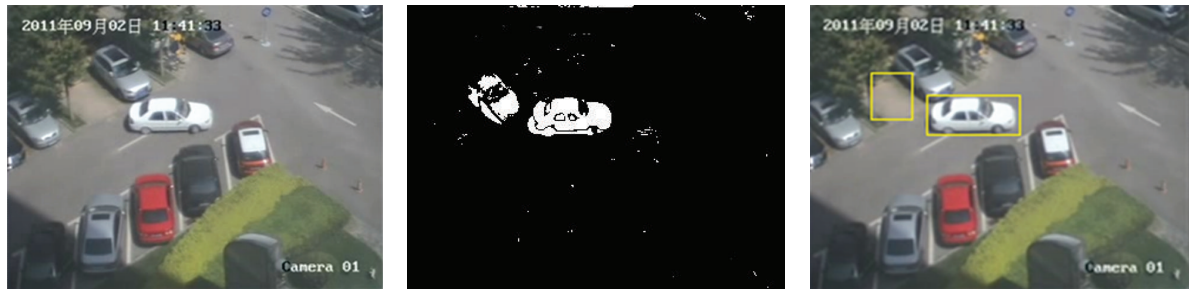
strategy, with the result that the algorithm is adaptive and able to deal with multimodal backgrounds in a dynamic environment (e.g., changing time of day, clouds, swaying tree leafs, and etc.). However, since its sensitivity cannot be properly tuned, its ability to successfully handle high- and low-frequency changes in the background is debatable. To overcome these shortages, samples-based techniques [34] circumvent a part of the parameter estimation step by building their models from observed pixel values and enhance their robustness to noises. They provide fast responses to high-frequency events in the background by directly including newly observed values in their pixel models. However, since they update their pixel models in a first-in first-out manner, their ability to successfully handle concomitant

(a) Original video (b) Gaussian mixture model (c) Random background model

Figure 3: The target detection results of video data set 1.

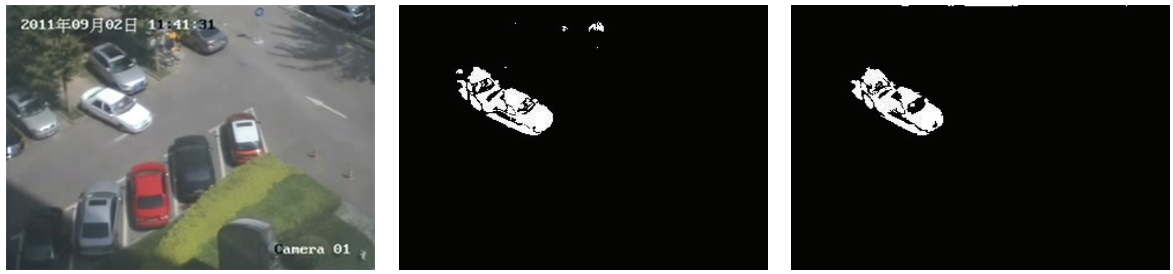
(a) Original video (b) Gaussian mixture model (c) Random background model

Figure 4: The target detection results of video data set 2.



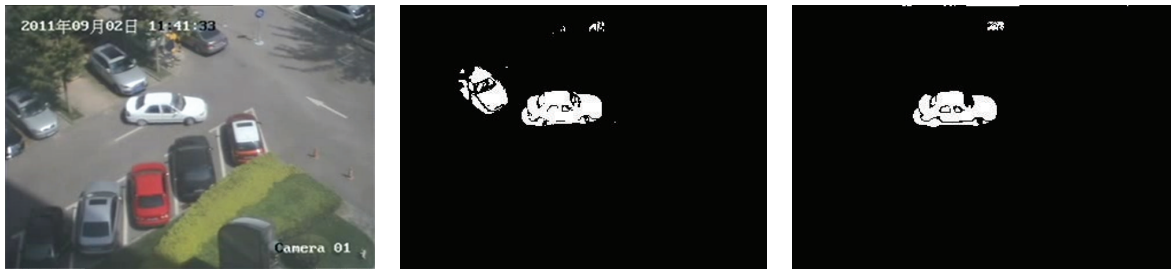
(a) Original video (b) Foreground image with ghost (c) The target detection result

Figure 5: The target detection results included the ghost.



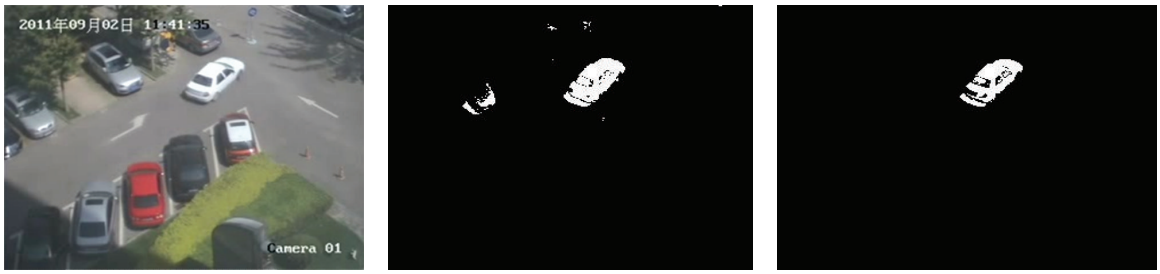
(a) Original video (b) Gaussian mixture model (c) Our method

Figure 6: The target detection results for frame 80.



(a) Original video (b) Gaussian mixture model (c) Our method

Figure 7: The target detection results for frame 105.



(a) Original video (b) Gaussian mixture model (c) Our method

Figure 8: The target detection results for frame 145.

events evolving at various speeds is limited, similarly to the limitation in its adaptive ability of dealing with the concurrent events with different frequency. In order to address this issue, random background modeling that is intuitively an improved samples-based algorithm is found in [33]. This algorithm assumes $p_t(x)$ to be the value of the pixel x at time t and imposes the constraint that the influence of a value on the polychromatic space is restricted within the local neighborhood. Then, a set of sample values is used as a pixel model to classify a value $p_t(x)$ to be either a background or a foreground pixel value.

An experiment was carried out using video data and compared with the Gaussian mixture model presented by Stauffer and Grimson [21], in which the video data set 1 comes from the evaluating data from Performance Evaluation of Tracking and Surveillance (PETS) database with the video image resolution of 768 × 576 pixels and the frame rate of 25 frames per second (f/s); the video data set 2 comes from a practical surveillance system data in the DNC of Beihang University, with the video image resolution of 352 × 288 pixels and the frame rate of 25 f/s. The experimental results are shown in Figures 3 and 4. As can be seen from Figures 3 and 4, since trees were swinging in the wind, such movements were classified as foreground motions by the Gaussian mixture model, while the random background model effectively detected the trees as the background. However, both algorithms do not take into account the shadow of the target and thus are severely damaged in terms of the reliability and robustness of the target detection and tracking, as illustrated in Figure 5, in which the used video data set 3 comes from a practical surveillance system in the DNC of Beihang University and is with the video image resolution of 352 × 288 pixels and the frame rate of 25 f/s.

In order to remove the damage from the shadow on target detection and tracking, we propose an algorithm by combining the random background model and the frame difference algorithm, and the mathematical model is described as follows:

$$\text{Mask}(x, y) = \begin{cases} 1 & \left|\text{Dilate}\left(D(x, y)\right) - \text{Erode}\left(D(x, y)\right)\right| = 1 \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

where $D(x, y)$ denotes the mask image of the background differencing; Dilate$(*)$ denotes the dilation operation of the target region block; Erode$(*)$ denotes the erosion operation of the target region; Mask(x, y) denotes the mask image of the difference between the dilated and eroded image operations.

Suppose that the number of the pixels with their values equal to 1 in Mask(x, y) is N_1 and the number of the pixels that are detected as foreground from differencing image and the values of which at (x, y) in the template Mask(x, y) equal to 1 is N_2. If $N_1/N_2 > T$, where T denotes a threshold, then the target region block is the foreground target; otherwise, it is the shadow of the target.

An experiment is carried out using the video data set 3 and compared with the GMM. The experimental results are shown in Figures 6, 7, and 8. As can be seen, the algorithm presented in this paper is effective to remove the shadow of the target.

Figure 9: The multitarget tracking results for video data set 4.

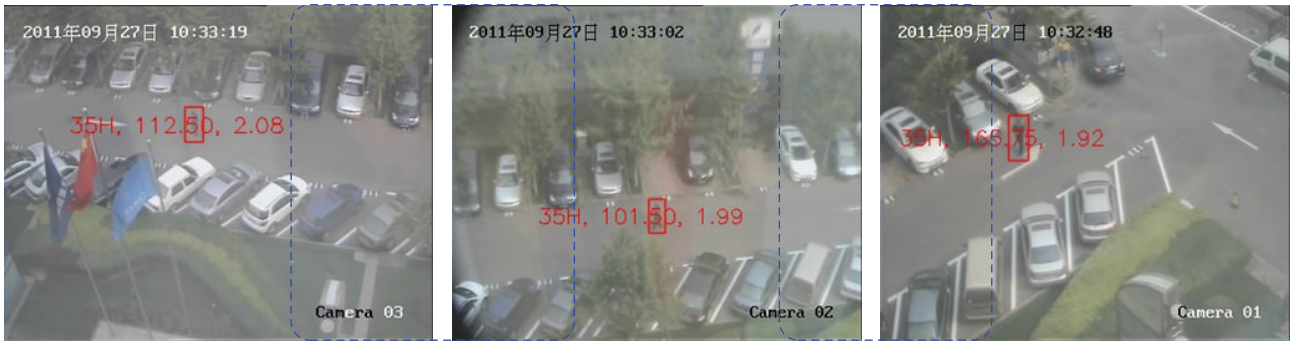


Figure 10: The target tracking results across multicamera for video data set 3.

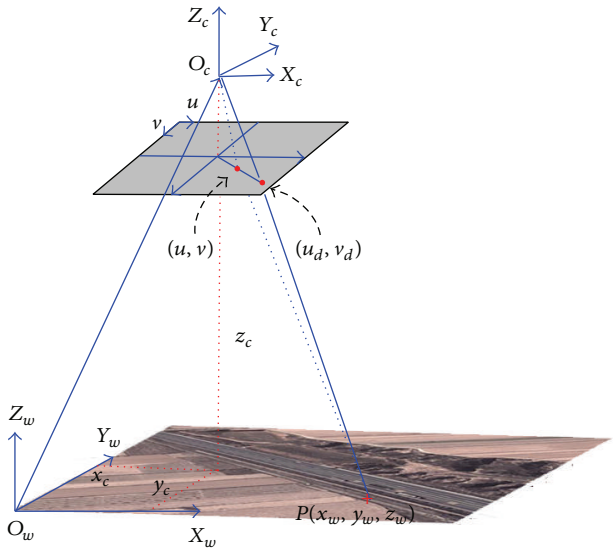


Figure 11: The relationship between the world coordinates and pixel coordinates.

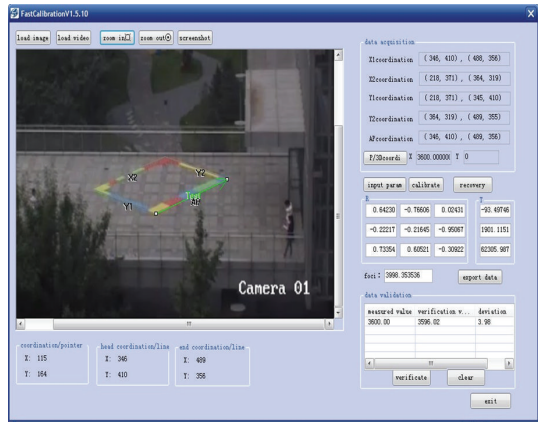


Figure 12: The fast calibrating camera tool software.

3.2. Target Tracking. Once moving targets are detected, the track initialization event is triggered, such that the moving targets can be continuously tracked by the tracking algorithm in the living period of a track (which starts from its initialization to its termination [35]). The termination of a track occurs when a target can no longer be detected because it leaves the field of view, it stops and becomes static, or it can no longer be distinguished from the background. Detected targets are not confirmed to be true moving targets until they have been consistently tracked for a period of time before their target tracks are initialized. We create a dynamic list of potential tracks using all detected targets. Associations will be established between targets detected in a new image frame and potential tracking targets. When a potential target is tracked in several continuous frames, it is recognized as a true moving target and a track will be initialized.

Compared to a single target tracking, the multitarget problem poses additional difficulties: data association needs to be solved; that is, it has to be decided which observation

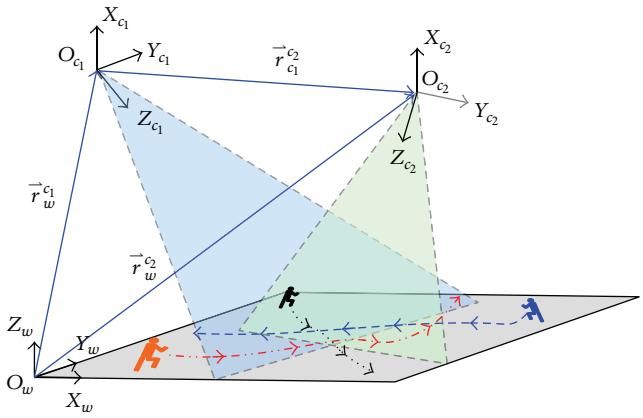


FIGURE 13: The schematic diagram of the target positioning.

corresponds to which target; constraints between targets need to be taken into account. Multitarget tracking algorithms can be roughly divided into two categories: the recursive algorithms and the nonrecursive algorithms. The recursive algorithms base their estimate only on the state of the previous frame such as Kalman filtering [36] and particle filtering [27], in which different strategy is used to obtain an optimal solution over multiple frames and can thus better cope with ambiguous, multimodal distributions. The nonrecursive algorithms seek optimality over an extended period of time [37, 38].

In practical applications, target tracking takes target splitting and merging into account because of the factors such as illumination changes and occlusion. Since the one-to-one tracking assumption rarely holds, multitarget tracking problem is still challenging. In order to realize multitarget tracking, we propose a solution by combining the pyramid Lucas-Kanade feature tracker [39] and Kalman filter. The mathematical model of Kalman filtering is described as follows:

$$X(k+1) = A(k+1,k)\,X(k) + W(k),$$
$$Z(k) = H(k)\,X(k) + V(k), \tag{2}$$

where $X(k) = [x(k), y(k), w(k), h(k), v_x(k), v_y(k), v_w(k), v_h(k)]^T$ denotes the state; $A(k+1,k)$ denotes the state transition matrix; $W(k)$ denotes the system noise; $H(k)$ denotes the measurement matrix; $Z(k) = [x(k), y(k), w(k), h(k)]^T$ denotes the measurement value; $V(k)$ denotes the measurement noise; $x(k)$ and $y(k)$ denote the horizontal and vertical ordinates of the target centroid; $w(k)$ and $h(k)$ denote the width and height of the target envelope rectangle; $v_x(k)$, $v_y(k)$, $v_w(k)$, and $v_h(k)$ denote the speeds of the $x(k)$, $y(k)$, $w(k)$, and $h(k)$, respectively.

To realize multitarget tracking across multiple cameras, we construct a similarity function of target matching to realize target association and target tracking across multiple cameras. The similarity function is described as follows:

$$f(a,b) = \alpha M_d(a,b) + \beta M_s(a,b) + \gamma M_h(a,b),$$
$$M_d(a,b) = \frac{(W_a + W_b)}{(W_a + W_b) + D_x} \cdot \frac{(H_a + H_b)}{(H_a + H_b) + D_y},$$
$$M_s(a,b) = \frac{2W_aH_aW_bH_b}{(W_aH_a)^2 + (W_bH_b)^2}, \tag{3}$$
$$M_h(a,b) = \frac{2h_ah_b}{h_a^2 + h_b^2},$$

where a and b denote targets which will be matched; $M_d(a,b) \in [0,1]$ denotes their position similarity (the larger the $M_d(a,b)$ is, the closer their positions are); $M_s(a,b) \in [0,1]$ denotes the similarity of their sizes (the larger the $M_s(a,b)$ is, the closer their sizes are); $M_h(a,b) \in [0,1]$ denotes the similarity of their heights (the larger the $M_h(a,b)$ is, the closer their heights are); $\alpha \in [0,1]$, $\beta \in [0,1]$, and $\gamma \in [0,1]$ denote the weight coefficients and satisfy $\alpha + \beta + \gamma = 1$; X_a and Y_a denote the horizontal and vertical ordinates of the target a in the world coordinate system; X_b and Y_b denote the horizontal and vertical ordinates of the target b in the world coordinate system; W_a and W_b denote half of the widths of the targets a and b, respectively; H_a and H_b denote half of their heights, respectively; $D_x = \text{fabs}(X_a - X_b)$ denotes the absolute difference between X_a and X_b; $D_y = \text{fabs}(Y_a - Y_b)$ denotes the absolute difference between Y_a and Y_b; h_a and h_b denote their heights in the world coordinate system; $f(a,b) \in [0,1]$ (the larger the $f(a,b)$ is, the higher their matching similarity is), and vice versa. In practical applications, α, β, and γ can be adjusted according to the accuracy of the $M_d(a,b)$, $M_s(a,b)$, and $M_h(a,b)$.

An experiment is carried out by using the video data sets 3 and 4 which come from a practical surveillance system in the DNC of Beihang University, with the video image resolution of 352 × 288 pixels and the frame rate of 25 f/s. The experimental results are shown in Figures 9 and 10, in which the rectangle with the dotted blue lines denotes the overlapping area between two cameras. As can be seen, the target tracking algorithm presented in this paper is effective and can track multiple targets across the multiple cameras.

4. Fast Camera Calibration and Target Positioning

4.1. Fast Camera Calibration. Camera calibration is a key technology in determining the mapping between 3D world coordinates and 2D image coordinates for various computer vision applications. A schematic diagram describing the mapping between 3D world coordinates and 2D image coordinates is shown in Figure 11. (u_d, v_d) is the image coordinate of (x_w, y_w, z_w) if a perfect pinhole camera model is used. (u, v) is the actual image coordinate which deviates from (u_d, v_d) due to lens distortion. The distance between them is termed as the radial distortion. Therefore, the mathematical

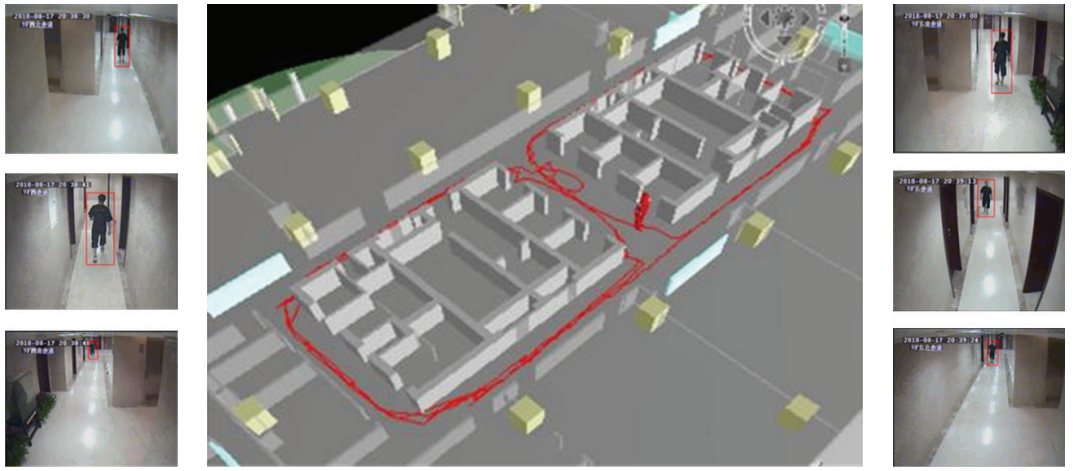


Figure 14: The experimental results in indoor environment.

model from 3D world coordinates to 2D image coordinates is expressed by [40]

$$u + \delta u = f_x \frac{r_1 (x_w - x_c) + r_2 (y_w - y_c) + r_3 (z_w - z_c)}{r_7 (x_w - x_c) + r_8 (y_w - y_c) + r_9 (z_w - z_c)}$$
$$v + \delta v = f_y \frac{r_4 (x_w - x_c) + r_5 (y_w - y_c) + r_6 (z_w - z_c)}{r_7 (x_w - x_c) + r_8 (y_w - y_c) + r_9 (z_w - z_c)}, \quad (4)$$

where r_i $(i = 1, 2, \ldots, 9)$ denotes the element of the rotation matrix from the world coordinate frame to the pixel coordinate frame; f_x and f_y denote the focal lengths of the camera in the x and y directions; δu and δv denote the photogrammetric distortions; $O_c(x_c, y_c, z_c)$ denotes the coordinate of the camera in the world coordinate frame; $P(x_w, y_w, z_w)$ denotes the coordinate of the target in the world coordinate frame.

Traditional calibration algorithms, for example, DLT algorithm [9] and Tsai algorithm [10], utilize a series of mathematical transformations and algorithms to obtain parameters of the camera model. They have been widely used because of their simple mathematical model and theory. However, these algorithms require a large amount of work to record and check calibration points during calibration and thus are inefficient in practical applications. For this reason, a fast calibration algorithm based on the vanishing point theory was introduced in [11, 41], in which the photogrammetric distortions δu and δv are unconsidered.

According to the mathematical model of the fast camera calibration presented in [11], we develop software to calibrate cameras fast and can promptly check the accuracy of camera parameters by (4). The calibration process and results of a practical camera are shown in Figure 12. As can be seen, the focal lengths of the camera are 3998.3535 pixels, the calibration error of the line segment AP is 3.98 mm, and the rotation matrix R and translation vector T (unit: mm) are as the following:

$$R = \begin{bmatrix} 0.642 & -0.766 & 0.024 \\ -0.222 & -0.216 & -0.951 \\ 0.734 & 0.605 & -0.309 \end{bmatrix}, \quad T = \begin{bmatrix} -93.497 \\ 1901.11 \\ 62305.937 \end{bmatrix}. \quad (5)$$

4.2. Target Positioning. Once targets are continuously tracked, the space coordinates of the targets in the camera coordinate frame can be computed by imaging model with camera parameters as follows:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = Z_c \begin{bmatrix} \frac{d_x}{f_x} & 0 & 0 \\ 0 & \frac{d_y}{f_y} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} + \begin{bmatrix} \frac{u_0 Z_c d_x}{f_x} \\ \frac{v_0 Z_c d_y}{f_y} \\ 0 \end{bmatrix}, \quad (6)$$

where u_0 and v_0 denote the principle point in the pixel frame and dx and dy denote the pixel sizes of the camera in the x and y directions.

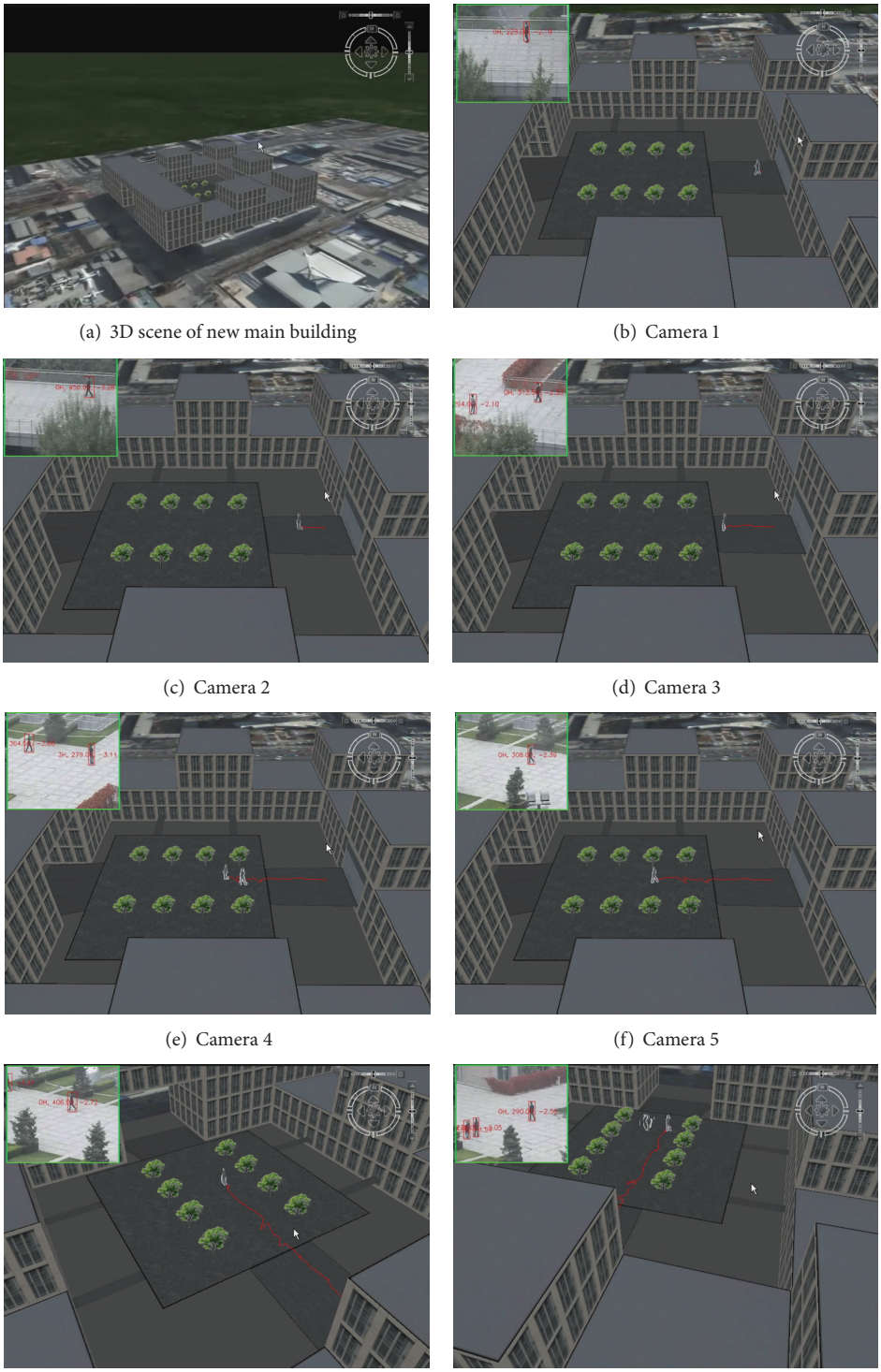
When the targets are across multiple cameras, the space coordinates of the targets in each camera coordinate frame are computed, respectively, by (6) and then are unified into the world coordinated system, as illustrated in Figure 13. The mathematical model is described as follows:

$$\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} = R_i \left(\begin{bmatrix} X_{c_i} \\ Y_{c_i} \\ Z_{c_i} \end{bmatrix} - T_i \right), \quad i = 1, 2, \ldots, N, \quad (7)$$

where i denotes the ith camera and N denotes the number of cameras.

5. Experiment Test

The target positioning system presented in this paper is tested in indoor and outdoor environments, in which all the low-cost static cameras are calibrated, and the coordinates of



(a) 3D scene of new main building (b) Camera 1

(c) Camera 2 (d) Camera 3

(e) Camera 4 (f) Camera 5

(g) Camera 6 (h) Camera 7

Figure 15: The experimental results in outdoor environment.

cameras are unified into the world coordinate system. Targets detection and tracking are done by the target detection and tracking algorithm. As a result, the targets are positioned by imaging model and camera parameters in real time, and their trajectories are displayed in a three-dimensional scene.

The test results are, respectively, shown in Figures 14 and 15 for indoor and outdoor environments. As can be seen from Figure 14, when a target is continuously moving within an indoor corridor, the positioning system consisting of six distributed cameras is able to position this target in real

time and display its trajectory in a three-dimensional space model. Likewise, as can be seen from Figure 15, when a target is continuously moving outdoors, the positioning system consisting of seven distributed cameras is able to position this target and display its trajectory in a three-dimensional space model in real time as well. The experimental results confirm that the systematic framework and inclusive algorithms presented in this paper are both effective and efficient.

In this paper, we assume that the ground is flat, which rarely holds in practical applications given a large region. In order to solve this problem, it is necessary to use digital elevation model (DEM) to describe topographic relief in large regions.

6. Conclusion and Future Work

This paper presented the comprehensive design and implementation of a moving target positioning system based on a distributed camera network. The system is composed of low-cost static cameras, which provide complementary positioning information for moving target positioning in indoor and outdoor environments when GNSS signals are unavailable. In this system, static cameras can cover a large region, moving targets are detected and then tracked using corresponding algorithms, target positions are estimated by making use of the geometrical relationships among those cameras after calibrating those cameras, and finally, for each target, its position estimates obtained from different cameras are unified into the world coordinate system. The experimental results of the target detection, tracking, and positioning system were reported based on real video data.

Targets positioning and tracking with multiple static cameras were verified in both indoor and outdoor environments. However, the reliability and accuracy of target tracking and positioning suffer from several environment factors. Hence it is necessary to fuse information from various sensors, such as radar, infrared camera, inertial measure unit (IMU), and wireless location system. Regarding future work, it is meaningful to develop and test these algorithms in the practical applications.

Conflict of Interests

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

Acknowledgments

This project is supported by the Key Program of the National Natural Science Foundation of China (Grant no. 61039003), the National Natural Science Foundation of China (Grant no. 41274038), the Aeronautical Science Foundation of China (Grant no. 2013ZC51027), the Aerospace Innovation Foundation of China (CASC201102), and the Fundamental Research Funds for the Central Universities.

References

[1] R. Mautz and S. Tilch, "Survey of optical indoor positioning systems," in *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN '11)*, Guimarães, Portugal, September 2011.

[2] G. Retscher, E. Moser, D. Vredeveld et al., "Performance and accuracy test of a WiFi indoor positioning system," *Journal of Applied Geodesy*, vol. 1, no. 2, pp. 103–110, 2007.

[3] S. J. Ingram, D. Harmer, and M. Quinlan, "Ultra wide band indoor positioning systems and their use in emergencies," in *Proceedings of the Position Location and Navigation Symposium (PLANS '04)*, pp. 706–715, Monterey, Calif, USA, April 2004.

[4] G. Anastasi, R. Bandelloni, M. Conti et al., "Experimenting an indoor bluetooth-based positioning service," in *Proceedings of the 23rd International Conference on Distributed Computing Systems Workshops*, pp. 480–483, Providence, RI, USA, 2003.

[5] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man and Cybernetics C: Applications and Reviews*, vol. 37, no. 6, pp. 1067–1080, 2007.

[6] H.-S. Kim and J.-S. Choi, "Advanced indoor localization using ultrasonic sensor and digital compass," in *Proceedings of the International Conference on Control, Automation and Systems (ICCAS '08)*, pp. 223–226, Seoul, Korea, October 2008.

[7] R. Karlsson, T. B. Schön, D. Törnqvist, G. Conte, and F. Gustafsson, "Utilizing model structure for efficient simultaneous localization and mapping for a UAV application," in *Proceedings of the IEEE Aerospace Conference Digital Object Identifier (AC '08)*, Rome, Italy, March 2008.

[8] D. Nistér, O. Naroditsky, and J. Bergen, "Visual odometry," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, pp. 652–659, Washington, DC, USA, July 2004.

[9] Y. I. Abdel-Aziz and H. M. Karara, "Direct linear transformation into object space coordinates in close-range photogrammetry," in *Proceedings of the Symposium on Close-Range Photogrammetry*, pp. 1–18, Urbana, Ill, USA, 1971.

[10] R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3D Machine vision metrology using off-the-shelf TV camera and lenses," *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 4, pp. 323–344, 1987.

[11] J. Long, X. Zhang, and L. Zhao, "A fast calibration algorithm based on vanishing point for scene camera," *Applied Mechanics and Materials*, vol. 58–60, pp. 1148–1153, 2011.

[12] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[13] P. D. Z. Varcheie and G.-A. Bilodeau, "People tracking using a network-based PTZ camera," *Machine Vision and Applications*, vol. 22, no. 4, pp. 671–690, 2011.

[14] C.-H. Chen, Y. Yao, D. Page, B. Abidi, A. Koschan, and M. Abidi, "Heterogeneous fusion of omnidirectional and PTZ cameras for multiple object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 8, pp. 1052–1063, 2008.

[15] X. Clady, F. Collange, F. Jurie, and P. Martinet, "Object tracking with a Pan-tilt-zoom camera: application to car driving assistance," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '01)*, pp. 1653–1658, Seoul, Korea, May 2001.

[16] N. Bellotto, E. Sommerlade, B. Benfold et al., "A dstributed camera system for multi-resolution surveillance," in *Proceedings*

of the 3rd ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC '09), Como, Italy, September 2009.

[17] C. Ding, B. Song, A. Morye et al., "Collaborative sensing in a distributed ptz camera network," *IEEE Transactions on Image Processing*, vol. 21, no. 7, pp. 3282–3295, 2012.

[18] E. B. Ermis, P. Clarot, P.-M. Jodoin, and V. Saligrama, "Activity based matching in distributed camera networks," *IEEE Transactions on Image Processing*, vol. 19, no. 10, pp. 2595–2613, 2010.

[19] D.-M. Tsai and S.-C. Lai, "Independent component analysis-based background subtraction for indoor surveillance," *IEEE Transactions on Image Processing*, vol. 18, no. 1, pp. 158–167, 2009.

[20] J. Migdal and W. E. L. Grimson, "Background subtraction using Markov thresholds," in *Proceedings of the IEEE Workshop on Motion and Video Computing (MOTION '05)*, pp. 58–65, Breckenridge, Colo, USA, January 2005.

[21] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 747–757, 2000.

[22] D.-S. Lee, "Effective Gaussian mixture learning for video background subtraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 827–832, 2005.

[23] L. Zhao and X. He, "Adaptive Gaussian mixture learning for moving object detection," in *Proceedings of the 3rd IEEE International Conference on Broadband Network and Multimedia Technology (IC-BNMT '10)*, pp. 1176–1180, Beijing, China, October 2010.

[24] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: a survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1773–1795, 2013.

[25] G. H. Wen, Z. S. Duan, G. R. Chen, and W. W. Yu, "Consensus tracking of multi-agent systems with lipschitz-type node dynamics and switching topologies," *IEEE Transactions on Circuits and Systems*, vol. 60, no. 9, pp. 1–13, 2013.

[26] S. Oh, S. Russell, and S. Sastry, "Markov chain Monte Carlo data association for multi-target tracking," *IEEE Transactions on Automatic Control*, vol. 54, no. 3, pp. 481–497, 2009.

[27] H. Zhang and L. Zhao, "Integral channel features for particle filter based object tracking," in *Proceedings of the 5th International Conference on Intelligent Human-Machine Systems and Cybernetics*, pp. 190–193, Hangzhou, China, 2013.

[28] M. Taj and A. Cavallaro, "Distributed and decentralized multi-camera tracking," *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 46–58, 2011.

[29] N. Krahnstoever, T. Yu, S. Lim et al., "Collaborative real-time control of active cameras in large scale surveillance systems," in *Proceedings of the Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications*, pp. 1–12, Marseille, France, 2008.

[30] C. Braillon, C. Pradalier, J. L. Crowley, and C. Laugier, "Real-time moving obstacle detection using optical flow models," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV '06)*, pp. 466–471, Tokyo, Japan, June 2006.

[31] C. Kim and J. N. Hwang, "Fast and automatic video object segmentation and tracking for content-based applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 2, pp. 122–129, 2002.

[32] R. Jain and H. H. Nagel, "On the analysis of accumulative difference pictures from image sequences of real world scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, no. 2, pp. 206–214, 1978.

[33] O. Barnich and M. Van Droogenbroeck, "ViBE: a powerful random technique to estimate the background in video sequences," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09)*, pp. 945–948, Taibei, Taiwan, April 2009.

[34] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *Proceedings of the the 6th European Conference on Computer Vision-Part II*, pp. 751–767, London, UK, 2000.

[35] H. W. Mao, C. H. Yang, G. P. Abousleman, and J. Si, "Automated multiple target detection and tracking in UAV videos," in *Airborne Intelligence, Surveillance, Reconnaissance (ISR) Systems and Applications VII*, vol. 7668 of *Proceedings of SPIE*, Orlando, Fla, USA, 2010.

[36] H. Medeiros, J. Park, and A. C. Kak, "Distributed object tracking using a cluster-based Kalman filter in wireless camera networks," *IEEE Journal on Selected Topics in Signal Processing*, vol. 2, no. 4, pp. 448–463, 2008.

[37] A. Andriyenko and K. Schindler, "Multi-target tracking by continuous energy minimization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 1265–1272, Colorado Springs, Colo, USA, June 2011.

[38] A. Andriyenko and K. Schindler, "Globally optimal multi target tracking on a hexagonal lattice," in *Proceedings of the 11th European Conference on Computer Vision*, pp. 466–479, Crete, Greece, 2010.

[39] J. Y. Bouguet, "Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm," *Intel Corporation*, pp. 1–9, 2001.

[40] Y. Wang, G. Hu, and Z. Chen, "Calibration of CCD camera in image matching experimental equipment," in *Proceedings of the 2nd International Symposium on Instrumentation Science and Technology*, p. 3/146, Jinan, China, August 2002.

[41] B. Caprile and V. Torre, "Using vanishing points for camera calibration," *International Journal of Computer Vision*, vol. 4, no. 2, pp. 127–139, 1990.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 461635, 8 pages
http://dx.doi.org/10.1155/2014/461635

Research Article

Synchronization Control of Time-Varying Complex Dynamic Network with Nonidentical Nodes and Coupling Time-Delay

Youjian Zhang, Wenjun Yan, and Qiang Yang

College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China

Correspondence should be addressed to Wenjun Yan; yanwenjun_zju@163.com

Received 15 January 2014; Accepted 18 April 2014; Published 7 May 2014

Academic Editor: Guoqiang Hu

This paper addresses the synchronization problem for a class of complex networks with time-varying topology as well as nonidentical nodes and coupling time-delay and presents two efficient control schemes to synchronize the network onto any given smooth goal dynamics. The time-varying network is supposed to be bounded within a certain range, which cannot be controlled. Through the adoption of hybrid control with linear static feedback control and adaptive feedback control, two control schemes are derived to guarantee such complex networks to reach the global synchronization. Finally, a set of numerical simulation experiments are carried out and the results demonstrate the effectiveness of the suggested control solutions.

1. Introduction

In the past decades, complex dynamic networks have been considered to be one of the challenging areas and received increasing attention from research and industry community due to the wide applications in various fields, for example, biological systems [1], wireless communication networks [2], artificial intelligent systems [3, 4], and so forth. In general, a complex network consists of a collection of interconnected nodes, where individual nodes are basic units representing specific contents or dynamics and the connections (edges) represent the interactions amongst them. It is virtually universally agreed that many collective behaviors and complex phenomena observed in the nature and society are related to the issue of network synchronization [1]. As a response, much research effort has been made to exploit the pattern and regulation of the network synchronization, and many control tools, for example, pinning control [5, 6], adaptive control [7], sliding mode control [8], and robust control [9, 10], have been applied to address the technical challenges.

In the literature, the existing solutions are mainly derived for the networks with constant topologies; that is, topologies are not varying over time (e.g., [6–8]). In fact, the studies on networks with time-varying topologies are more realistic as the dynamical characteristics of the complex networks can be better modeled and investigated. In parallel, there are some existing works concerning synchronization in dynamic networks; that is, network coupling structure or strength dynamically changes over time (e.g., [11–13]). Xiao et al. [11] studied the synchronization of complex switched networks subject to network delay as well as switching behaviors exhibited in both nodes and topological structure. The synchronization in a complex dynamical network with randomly switching topology was studied by Lee et al. [12]. Lee et al. [13] addressed the issue in an uncertain complex dynamical network, where the norm-bounded uncertainties imposed on the complex dynamical network in a random fashion.

In addition, the behavior of a complex dynamical network can be determined by both coupling configuration amongst nodes and node dynamics. For the sake of simplicity, previous studies often assume that the dynamics of all network nodes are identical, which makes the control solutions not able to be directly applied to the realistic networks as the individual network nodes often demonstrate diverse physical parameters [14, 15]. The synchronization of the complex networks with nonidentical nodes is still an open problem to be further addressed and some solutions are available [14–20]. In [14], the authors designed a high gain

integral controller for synchronization of complex dynamical networks with unknown nonidentical nodes. In [15], the authors investigated the issue for complex networks with nonidentical nodes. Shi et al. [16] presented an adaptive synchronization control solution for a novel complex dynamical network model with nonidentical nodes and nonderivative and derivative couplings. Fan et al. [17] investigated the synchronization in the context of a class of complex dynamical network with the similar nodes and coupling time-delay. Wang et al. [18] investigated the global bounded synchronization problem of the complex dynamical networks with coupled nonidentical nodes and time-varying topology.

In this paper, we address the synchronization problem with the explicit consideration of the dynamical complex networks with the time-varying topology as well as nonidentical nodes and coupling time-delay simultaneously. To our best knowledge, little research outcome tackling this issue has been reported. The main technical contributions of this paper can be summarized twofold: (1) two synchronization control laws are proposed for time-varying complex networks with nonidentical nodes and coupling time-delay and (2) the assessment verifies that a kind of complex network with nonidentical nodes and time-varying topological structures can be well synchronized through applying the suggested control solution onto any given smooth goal dynamics, for example, a periodic orbit or a chaotic trajectory.

The remainder of this paper is organized as follows: the problem formulation and preliminaries are introduced in Section 2; in Section 3, two synchronization control laws for time-varying complex networks with nonidentical nodes and coupling time-delay are presented, followed by Section 4 carrying out the numerical simulations and presenting a set of key experimental results; finally, the conclusive remarks are given in Section 5.

Notations. In this study, a real symmetric matrix $P > 0 (\geq 0)$ denotes that P is a positive definite (or positive semidefinite) matrix, and $A > (\geq) B$ indicating that $A - B > (\geq) 0$. The superscript "T" represents the transpose of a matrix. The matrices, if their dimensions are not explicitly stated, are assumed to have compatible dimensions for algebraic operations.

2. Problem Formulation and Preliminaries

Consider a class of complex network consisting of N nonidentical nodes with linearly diffusive time-delay couplings, and each node is an n-dimensional dynamic system. The controlled complex network can be described as the following form:

$$\dot{x}_i(t) = f_i(x_i(t)) + c\sum_{j=1}^{N} a_{ij}(t) H x_j(t - \tau(t)) + u_i(t), \quad i = 1, 2, \ldots, N, \tag{1}$$

where $x_i(t) = [x_{i1}(t), x_{i2}(t), \ldots, x_{in}(t)]^T \in R^n$ is the state variable of node i; c is the coupling strength ($c > 0$); $u_i(t)$ is the control action applied to node i; $f_i : [0, \infty) \times R^n \to R^n$ is a vector-valued continuous function which defines the dynamics of node i; $H \in R^{n \times n}$ is a positive definite diagonal matrix which describes the individual couplings between node i and node j; and the coupling configuration matrix $A(t) = [a_{ij}(t)]_{n \times n} \in R^{N \times N}$ represents the time-varying topological structure of network (1) at time t. If there is a connection from node j to node i, $a_{ij}(t) = a_{ji}(t) > 0$ $(i \neq j)$; otherwise, $a_{ij}(t) = a_{ji}(t) = 0$, $(i \neq j)$. Finally, the row sum of $A(t)$ is zero; for example, $a_{ii}(t) = -\sum_{j=1, j \neq i} a_{ij}(t) < 0$, $i, j = 1, 2, \ldots, N$. The term of time-varying delay $\tau(t)$ is subject to

$$\tau(t) \geq 0, \quad \dot{\tau}(t) \leq \mu < 1, \quad \forall t \geq 0. \tag{2}$$

Assumption 1. The weighted connection between any two nodes is bounded; that is,

$$0 \leq \underline{a}_{ij} \leq a_{ij}(t) \leq \overline{a}_{ij} < \infty, \quad i \neq j. \tag{3}$$

Lemma 2 (see [7]). *The eigenvalues of an irreducible matrix $H = (h_{ij})_{N \times N}$ with $h_{ij} \geq 0, 1 \leq i, j \leq N$, and $i \neq j$ and $h_{ii} = -\sum_{j=1, j \neq i} h_{ij}$, $i = 1, 2, \ldots, N$, have the following properties: all the eigenvalues of H are less than or equal to 0; namely $H \leq 0$.*

Corollary 3. *Denote $\underline{A} = \{\underline{a}_{ij}\}_{n \times n}$, $\overline{A} = \{\overline{a}_{ij}\}_{n \times n}$, and $A(t) = \{a_{ij}(t)\}$; then the following properties can be derived:*

$$0 \geq \underline{A} \geq A(t) \geq \overline{A}. \tag{4}$$

Proof. Denote $c_{ij} = \overline{a}_{ij} - a_{ij}(t)$, $C = \{c_{ij}\}$, and based on Assumption 1, we have $c_{ij} \geq 0$, $1 \leq i, j \leq N$, $i \neq j$, and $c_{ii} = \overline{a}_{ii} - a_{ii}(t) = -\sum_{j=1, j \neq i} \overline{a}_{ij} - \sum_{j=1, j \neq i} a_{ij}(t) = -\sum_{j=1, j \neq i} c_{ij}$, $i = 1, 2, \ldots, N$. Thus, from Lemma 2, one can derive that $\underline{A}(t) \leq 0$, $\overline{A} = A(t) + C \leq 0$, and $C \leq 0$, and one can derive $\overline{A} \leq A(t)$. In a similar way, one can obtain $A(t) \leq \underline{A} \leq 0$. This completes the proof. □

Definition 4 (see [15]). Assuming $s(t) \in R^n$ is any smooth dynamics, the controlled complex network described in (1) is considered to be globally asymptotically synchronized onto the homogenous state $s(t)$ if its solution satisfies $\lim_{t \to \infty} \|x_i(t) - s(t)\| = 0$, $i = 1, 2, \ldots, N$, for any initial conditions.

To analyze the synchronization problem of dynamical networks, the problem can be effectively transformed into the stabilization analysis of the corresponding error networks, where the synchronization error vectors are defined as $e_i(t) = x_i(t) - s(t)$, $i = 1, \ldots, N$. As the row sum of the coupling matrix is zero, one can derive $\sum_{j=1}^{N} a_{ij}(t) H s(t - \tau(t)) = 0$; thus the following dynamical error equation can be obtained:

$$\begin{aligned} \dot{e}_i(t) &= \dot{x}_i(t) - \dot{s}(t) \\ &= f_i(x_i(t)) - \dot{s}(t) + u_i(t) + c\sum_{j=1}^{N} a_{ij}(t) H x_j(t - \tau(t)) \\ &= f_i(x_i(t)) - \dot{s}(t) + u_i(t) \end{aligned}$$

$$+ c\sum_{j=1}^{N} a_{ij}(t) H\left(x_j(t-\tau(t)) - s(t-\tau(t))\right)$$

$$= f_i(x_i(t)) - \dot{s}(t) + c\sum_{j=1}^{N} a_{ij}(t) He_j(t-\tau(t)) + u_i(t). \tag{5}$$

For synchronizing the dynamics of network (1) onto the given homogenous state $s(t)$, the suitable control laws $u_i(t)$ need to be designed such that the synchronization errors converge to zero in the norm sense. From the error system expressed in (5), the hybrid control scheme can be designed and described as follows:

$$u_i(t) = u_{i1}(t) + u_{i2}(t), \tag{6}$$

$$\begin{aligned} u_{i1}(t) &= \dot{s}(t) - f_i(s(t)), \\ u_{i2}(t) &= -cd_i(x_i(t) - s(t)). \end{aligned} \tag{7}$$

Remark 5. The control input $u_{i1}(t)$ can be considered as an open-loop control or entrainment control. The entrainment control based approaches provide an efficient tool to entrain nonlinear dynamical systems to any desired goal dynamics [21]. However, the open-loop control cannot guarantee the system stability during the control process. Thus, additional control is required to ensure the globally asymptotical system synchronization. In this paper, the open-plus-closed-loop (OPCL) [15] control method is adopted for the synchronization of complex networks given in (1). As a result, the entrainment control, static feedback control, and adaptive control are incorporated to regulate the network synchronization: the entrainment control $u_{i1}(t)$ provides the basis for developing the synchronization schemes for a network with nonidentical nodes and compensates the nonidentity; and the feedback control and adaptive control $u_{i2}(t)$ are used to ensure the global stability of the synchronization process.

Remark 6. Due to the fact that the network nodes are nonidentical, a common equilibrium solution of all isolated nodes can be hardly determined. Thus, when the complex network (1) has been synchronized onto an expected state $s(t)$, the input control (6) still needs to provide control signal to compensate the nonidentity, implying that the input control (6) is not zero when the network (1) is synchronized. In the case that all nonidentical network nodes have the same equilibrium, the input control (6) will be zero when the network (1) is synchronized.

Before presenting the derivation of the main results, the assumptions and lemmas are introduced as follows.

Assumption 7 (see [15]). For system (1), there exist the constants ω_i, $i = 1, 2, \ldots, N$, such that

$$\begin{aligned} (x-y)^T (f_i(x) - f_i(y)) &\le \omega_i (x-y)^T (x-y), \\ \forall x, y \in R^n, \quad 1 \le i \le N. \end{aligned} \tag{8}$$

It has been recognized that many typical benchmark chaotic systems, such as the Lorenz system, Chen system, Lü system, and the unified chaotic system, satisfy Assumption 7.

Assumption 8. The network (1) is considered to be always a connected network during the network dynamical changes (e.g., coupling strength variation) over time in this study; that is, the network has no isolated clusters. Otherwise, one may consider the synchronization on each connected component of the network separately.

Lemma 9 (see [22]). *Let Q and R be two symmetric matrices, and matrix S has the compatible dimension. Then*

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} < 0 \tag{9}$$

if and only if $R < 0$ and $Q - SR^{-1}S^T < 0$.

3. Main Results

In this section, we design and present two feedback control strategies (7) based on the hybrid control scheme (6). The sufficient conditions are derived to ensure the synchronization of time-varying complex network with nonidentical nodes and constant coupling time-delay. For the sake of clarity, denote $\psi_i^T(t) = [\bar{e}_i^T(t) \bar{e}_i^T(t-\tau)]$ and $\bar{e}_i(t) = [e_{1i}(t), e_{2i}(t), \ldots, e_{Ni}(t)]^T$, which will be used in the rest of this section.

3.1. Synchronization Scheme Combining Static Feedback Control. In the control scheme (7), if $d_1, d_2, \ldots, d_n$ are positive constants, the controller $u_{i2}(t) = -cd_i(x_i(t) - s(t))$ is considered as the linear static feedback controller and the error dynamical system can be written as the following form:

$$\begin{aligned} \dot{e}_i(t) &= f_i(x_i(t)) - f_i(s(t)) \\ &\quad + c\sum_{j=1}^{N} a_{ij}(t) He_j(t-\tau(t)) - cd_i(x_i(t) - s(t)). \end{aligned} \tag{10}$$

Thus, we need to prove that, if $d_1, d_2, \ldots, d_n$ are appropriately designed, the controller (7) can synchronize the network (1) to the given goal state $s(t)$.

Theorem 10. *Suppose Assumptions 1, 7, and 8 and* (2) *hold, if $d_1, d_2, \ldots, d_n$ are nonnegative constants and satisfy the following condition:*

$$2W - 2cD + I + (1-\mu)^{-1} c^2 h^2 \underline{A} \underline{A}^T < 0, \tag{11}$$

where $W = \mathrm{diag}(\omega_1, \ldots, \omega_N)$, $D = \mathrm{diag}(d_1, d_2, \ldots, d_n)$, and $h = \max(h_i)$, $i = 1, \ldots, N$, and the hybrid control controller with static feedback control (7) *guarantees that the controlled network* (1) *can be asymptotically synchronized to the given goal state $s(t)$.*

Proof. We define the Lyapunov-Krasovskii function as

$$V(t) = \sum_{i=1}^{N} e_i^T(t) e_i(t) + \sum_{i=1}^{N} \int_{t-\tau(t)}^{t} e_i^T(s) e_i(s)\, ds. \tag{12}$$

Differentiating the function $V(t)$, we have

$$\begin{aligned}
\dot{V}(t) &= 2\sum_{i=1}^{N} e_i^T(t)\left[f_i(x_i(t)) - f_i(s(t)) + c\sum_{j=1}^{N} a_{ij}(t) He_j(t-\tau(t)) - cd_i e_i(t)\right] \\
&\quad + \sum_{i=1}^{N}\left[e_i^T(t)e_i(t) - (1-\dot{\tau}(t)) \times e_i^T(t-\tau(t))e_i(t-\tau(t))\right] \\
&\le 2\sum_{i=1}^{N} e_i^T(t)\omega_i e_i(t) \\
&\quad + 2c\sum_{i=1}^{N} e_i^T(t)\sum_{j=1}^{N} a_{ij}(t) He_j(t-\tau(t)) \\
&\quad - 2c\sum_{i=1}^{N} e_i^T(t) d_i e_i(t) \\
&\quad + \sum_{i=1}^{N} e_i^T(t)e_i(t) - (1-\mu)\sum_{i=1}^{N} e_i^T(t-\tau(t))e_i(t-\tau(t)) \\
&= 2\sum_{i=1}^{n} \bar{e}_i^T(t) W\bar{e}_i(t) \\
&\quad + 2c\sum_{i=1}^{n} h_i\bar{e}_i^T(t) A(t)\bar{e}_i(t-\tau(t)) - 2c\sum_{i=1}^{n}\bar{e}_i^T(t) D\bar{e}_i(t) \\
&\quad + \sum_{i=1}^{n}\bar{e}_i^T(t)\bar{e}_i(t) - (1-\mu)\sum_{i=1}^{n}\bar{e}_i^T(t-\tau(t))\bar{e}_i(t-\tau(t)) \\
&= \sum_{i=1}^{n}\psi_i^T(t)\Omega_i\psi_i(t),
\end{aligned} \tag{13}$$

where

$$\Omega_i = \begin{bmatrix} 2W - 2cD + I & ch_i A(t) \\ ch_i A^T(t) & -(1-\mu)I \end{bmatrix}. \tag{14}$$

From Corollary 3, one can have $0 \le A(t)A^T(t) \le \underline{A}\underline{A}^T$, and hence the following can be derived

$$\begin{aligned}
&2W - 2cD + I + (1-\mu)^{-1}c^2h_i^2 A(t)A^T(t) \\
&\quad \le 2W - 2cD + I + (1-\mu)^{-1}c^2h^2\underline{A}\underline{A}^T.
\end{aligned} \tag{15}$$

Therefore, if inequality (11) holds, the inequality $2W - 2cD + I + (1-\mu)^{-1}c^2h_i^2A(t)A^T(t) < 0$ holds, which guarantees $\dot{V}(t) < 0$. Based on LaSalle's invariance principle [23], starting from any initial condition, we have $e_i(t) \to 0$ as $t \to \infty$. Therefore, the controlled network (1) is globally asymptotically synchronized; that is, $x_i(t) \to s(t)$ as $t \to \infty$, $i = 1, \ldots, N$. This completes the proof. □

Remark 11. Suppose that the feedback gain matrix is $D = \mathrm{diag}(d_1, d_2, \ldots, d_n)$, $d_i > 0$, $i = 1, 2, \ldots, N$. A Cost Function is defined as $\mathrm{CF} = \sum_{i=1}^{N} d_i$. Then we can get the most efficient control strategy at the meaning of the defined Lyapunov function. The smaller the CF is, the more efficient a control strategy will be to achieve synchronization and the more easily the implementation will be realized. Thus we can get the most efficient control strategy in a certain meaning by minimizing the sum of elements of D, which satisfies condition (11), to make the system individually synchronized, and the pseudooptimal control cost is $\mathrm{CF}_{\mathrm{best}} = \min(\mathrm{sum}(d_i))$.

3.2. Synchronization Scheme Combining Adaptive Feedback Control. In the control scheme (7), if $d_1, d_2, \ldots, d_n$ are defined as

$$\dot{d}_i = \rho_i e_i^T(t) e_i(t), \quad i = 1, 2, \ldots, N, \tag{16}$$

where ρ_i are positive constants, we say that the controller $u_{i2}(t) = -cd_i(x_i(t) - s(t))$ is an adaptive feedback controller. In the following, we will prove that the system (1) would get global synchronized to the given homogenous state $s(t)$ under the controller (6) with adaptive feedback control (17).

Theorem 12. *Suppose that Assumptions 1, 7, and 8 and* (2) *hold, and if $d_1, d_2, \ldots, d_n$ are defined as*

$$\dot{d}_i = \rho_i e_i^T(t) e_i(t), \quad i = 1, 2, \ldots, N, \tag{17}$$

where ρ_i are positive constants, the system can reach the global synchronization under the control scheme (6) *with adaptive control* (17). *Also, the adaptive feedback gains converge to the certain bounded constants; that is, $d_i(t) \to l_i$ as $t \to \infty$, where $l_i \ge 0$, $i = 1, 2, \ldots, N$.*

Proof. Define a Lyapunov-Krasovskii function as follows:

$$\begin{aligned}
V(t) &= \sum_{i=1}^{N} e_i^T(t) e_i(t) + \sum_{i=1}^{N}\int_{t-\tau(t)}^{t} e_i^T(s) e_i(s)\, ds \\
&\quad + \frac{c}{\rho_i}\sum_{i=1}^{N}(d_i(t) - l_i)^2.
\end{aligned} \tag{18}$$

Differentiating the function $V(t)$, we have

$$\begin{aligned}
\dot{V}(t) &= 2\sum_{i=1}^{N} e_i^T(t)\left[f_i(x_i(t)) - f_i(s(t)) + c\sum_{j=1}^{N} a_{ij}(t) He(t-\tau(t)) - cd_i e_i(t)\right] \\
&\quad + \sum_{i=1}^{N}\left[e_i^T(t)e_i(t) - (1-\dot{\tau}(t)) \times e_i^T(t-\tau(t))e_i(t-\tau(t))\right] \\
&\quad + 2c\sum_{i=1}^{N}(d_i(t) - l_i)e_i^T(t)e_i(t)
\end{aligned}$$

$$
\begin{aligned}
&\le 2\sum_{i=1}^{N} e_i^T(t)\,\omega_i e_i(t) \\
&\quad + 2c\sum_{i=1}^{N} e_i^T(t)\sum_{j=1}^{N} a_{ij}(t)\,He(t-\tau(t)) \\
&\quad - 2c\sum_{i=1}^{N} e_i^T(t)\,l_i e_i(t) \\
&\quad + \sum_{i=1}^{N} e_i^T(t)\,e_i(t) - (1-\mu)\sum_{i=1}^{N} e_i^T(t-\tau(t))\,e_i(t-\tau(t)) \\
&= 2\sum_{i=1}^{n} \bar{e}_i^T(t)\,W\bar{e}_i(t) + 2c\sum_{i=1}^{n} h_i\bar{e}_i^T(t)\,A(t)\,\bar{e}_j(t-\tau(t)) \\
&\quad - 2c\sum_{i=1}^{n} \bar{e}_i^T(t)\,L\bar{e}_i(t) \\
&\quad + \sum_{i=1}^{n} \bar{e}_i^T(t)\,\bar{e}_i(t) - (1-\mu)\sum_{i=1}^{n} \bar{e}_i^T(t-\tau(t))\,\bar{e}_i(t-\tau(t)) \\
&= \sum_{i=1}^{n} \psi_i^T(t)\,\Omega_i\psi_i(t),
\end{aligned}
\tag{19}
$$

where $\Omega_i = \begin{bmatrix} 2W-2cL+I & ch_iA(t) \\ ch_iA^T(t) & -(1-\mu)I \end{bmatrix}$.

According to Corollary 3, one can easily get that $0 \le A(t)A^T(t) \le \underline{AA}^T$. Then we can derive the following inequality:

$$
\begin{aligned}
&2W - 2cL + I + (1-\mu)^{-1}c^2h_i^2A(t)\,A^T(t) \\
&\qquad \le 2W - 2cL + I + (1-\mu)^{-1}c^2h^2\underline{AA}^T.
\end{aligned}
\tag{20}
$$

Thus, the diagonal matrix L can be found to make inequality (21) hold,

$$
2W - 2cL + I + (1-\mu)^{-1}c^2h^2\underline{AA}^T < 0, \tag{21}
$$

which can guarantee that inequality $\Omega_i < 0$ holds. Then we have $\dot{V}(t) < 0$, so the controlled network (1) is globally asymptotically synchronized according to Lyapunov stability theory [23]. This completes the proof. □

4. Numerical Simulation

In this section, two numerical simulation experiments are carried out to verify the effectiveness of the proposed synchronization control solutions.

4.1. Synchronizing Network by Hybrid Controller with Static Feedback Control. This numerical example aims to illustrate the effectiveness of the Corollary 3 (Theorem 10) presented in Section 3. Consider a network consisting 5 nonidentical nodes with time-varying network topology and coupling time-delay, as described by Chua's system, Lorenz system, and linear system. The complex network is given by

$$
\dot{x}_i(t) = f_i(x_i(t)) + c\sum_{j=1}^{5} a_{ij}(t)\,Hx_j(t-\tau(t)), \tag{22}
$$

where $c = 1$, $H = \mathrm{diag}(1,1,1)$, $\tau = 0.6 + 0.4\sin(t)$, and the time-varying connection matrix

$$
A(t) = \begin{bmatrix}
-(3+0.5\sin t+2\sin 10t+0.5\cos t) & 1+\sin 10t & 0.5+0.5\sin t & 0.5+0.5\cos t & 1+\sin 10t \\
* & -(1+\sin 10t) & 0 & 0 & 0 \\
* & * & -(0.5+0.5\sin t) & 0 & 0 \\
* & * & * & -(0.5+0.5\cos t) & 0 \\
* & * & * & * & -(1+\sin 10t)
\end{bmatrix}. \tag{23}
$$

The state-space functions and parameters of these nonidentical nodes are chosen as

$$
f_1(x_1) = \begin{cases} 10(x_{12} - x_{11} - F(x_{11})) \\ x_{11} - x_{12} + x_{13} \\ -14.87x_{12}, \end{cases} \tag{24}
$$

where $F(x_{11}) = -0.68x_{11} + 0.5(-1.27+0.68)(|x_{11}+1| + |x_{11}-1|)$,

$$
f_2(x_2) = \begin{cases} 10(x_{22} - x_{21}) \\ 28x_{21} - x_{22} - x_{21}x_{23} \\ x_{21}x_{22} - \left(\dfrac{8}{3}\right)x_{23}, \end{cases}
$$

$$
f_3(x_3) = \begin{cases} -3x_{31} + x_{32} + 2x_{33} \\ x_{32} + 3x_{33} \\ 0.5x_{31} + 4x_{32} - 2x_{33}, \end{cases}
$$

$$
f_4(x_4) = \begin{cases} 5x_{42} - 10x_{41} + 3(|x_{41}+1| + |x_{41}-1|) \\ 2x_{41} - 4x_{42} + 3x_{43} \\ -5x_{42}, \end{cases}
$$

$$
f_5(x_5) = \begin{cases} x_{52} \\ x_{53} \\ -x_{51} - x_{52} - x_{53} + 4\ln\left(x_{53} + \sqrt{1+x_{53}^2}\right). \end{cases} \tag{25}
$$

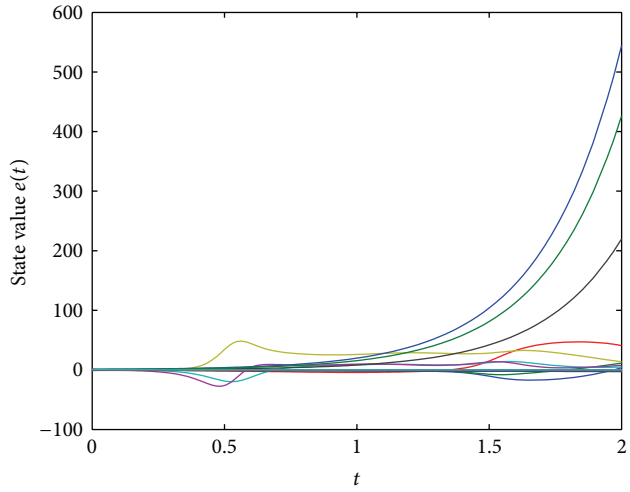


Figure 1: The trajectories of the network state without control.

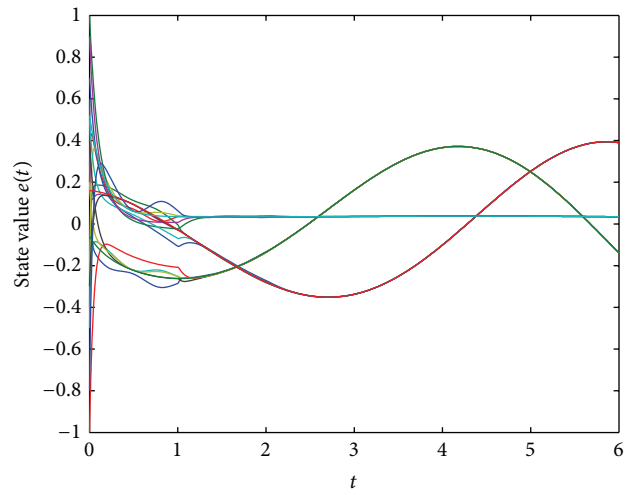


Figure 2: The trajectories of the network state with the feedback control.

To demonstrate the effectiveness of Theorem 10, we use the synchronization scheme (6) which includes static feedback control (7) to achieve the synchronization between network (22) and the solution $s(t)$of the following Rössler system:

$$\begin{aligned} \dot{s}_1(t) &= -(s_2 + s_3), \\ \dot{s}_2(t) &= s_1 + 0.2s_2, \\ \dot{s}_3(t) &= s_3(s_1 - 5.7) + 0.2. \end{aligned} \tag{26}$$

The initial conditions of each node are chosen as follows: $x_1(0) = [0.6, -0.4, 0.2]$, $x_2(0) = [-0.5, 0.3, -0.3]$, $x_3(0) = [0.4, 0.7, 1]$, $x_4(0) = [-1, 0.5, 0.9]$, $x_5(0) = [0.2, -0.3, -0.8]$, and $s(0) = [0.06, 0.16, 0.52]$. In order to derive the static control gain, the parameters need to be determined to satisfy Assumption 1. Based on the work in [15], we can derive that $\omega_1 = 9.1$, $\omega_2 = 27.5$, $\omega_3 = 3.5$, $\omega_4 = 0.4$, and $\omega_5 = 3.1$. By using MATLAB LMI Toolbox, it can be calculated that $D = [86.3, 54.7, 15.7, 12.6, 30.3]$ and $\mathrm{CF}_{\mathrm{best}} = 199.6$ based on Theorem 10.

In Figure 1, the trajectories of the uncontrolled state of nodes are depicted which demonstrate the original behavior of the complex network (26). The simulation result with control inputs which are calculated by Corollary 3 is illustrated in Figure 2. It can be seen that the trajectories of system state are stabilized and approach to the same state orbits.

4.2. Synchronizing Network by Hybrid Controller with Adaptive Feedback Control. Here, we adopt the control scheme (6) which contains adaptive control (17) to achieve the synchronization between network (22) and a chaotic trajectory given by (26), that is, verifying the effectiveness of the control scheme by synchronizing the system and target trajectory given in Section 4.1.

The simulation result with control inputs which are calculated by Theorem 12 is presented in Figure 3, and the adaptive control gains are presented in Figure 4. It can be observed from Figure 3 that the trajectories of system state can be stabilized and converged to the given system trajectory.

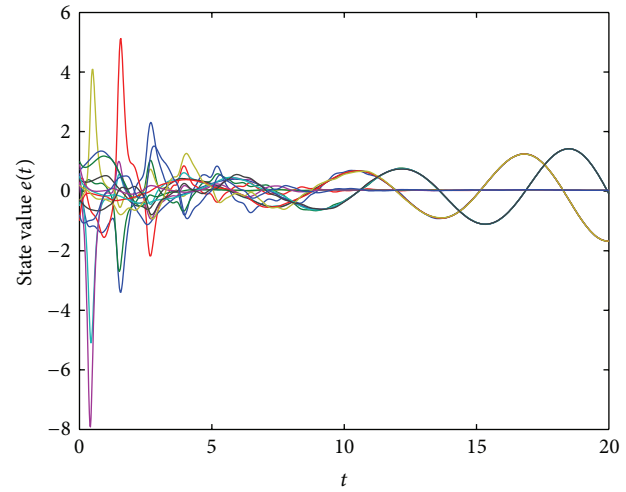


Figure 3: The trajectories of the network statewith the adaptive control.

In addition, the adaptive gains reach some certain constant values and the complex network is synchronized.

5. Conclusions

In this paper, the synchronization of time-varying complex dynamical networks with nonidentical nodes and coupling time-delay is investigated where the coupling matrix is assumed to be uncontrollable and bounded in a certain range. By combining the compensation approach for the nonidentity and error feedback effects, we presented two efficient control approaches that can successfully synchronize the complex network onto any given smooth goal dynamics. In addition, the sufficient conditions have been derived to guarantee the global asymptotic stability throughout the network synchronization process. Finally, the simulation experiments are carried out to assess the performance and the

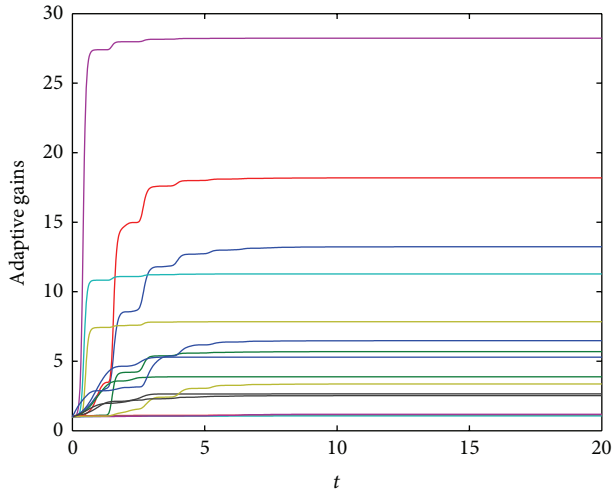


Figure 4: The adaptive gains.

numerical results confirm the effectiveness of the suggested control solutions. The conservation of the design approach for the controller needs to be further investigated and evaluated in future work, and more research effort needs to be made to improve the controller design with the minimum conservation.

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 anonymous reviewers for their invaluable comments. This work is supported in part by the National High-Tech R&D Program (863 plan) under the Grant no. 2012AA051704, the National Natural Science Foundation of China (under Grant no. 60804045 and no. 51107113), and the Science and Technology Plan of Zhejiang Province (no. 2013C31005).

References

[1] L. D. F. Costa, O. N. Oliveira, G. Travieso et al., "Analyzing and modeling real-world phenomena with complex networks: a survey of applications," *Advances in Physics*, vol. 60, no. 3, pp. 329–412, 2011.

[2] L. Lin, L. Chen, X. Fu, and S. Liu, "An improved wireless sensor network mode based on complex network theory," *Procedia Engineering*, vol. 24, pp. 360–364, 2011.

[3] Y. Zhao, Z. Duan, G. Wen, and Y. Zhang, "Distributed finite-time tracking control for multi-agent systems: an observer-based approach," *Systems & Control Letters*, vol. 62, no. 1, pp. 22–28, 2013.

[4] 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.

[5] W. Yu, G. Chen, and J. Lü, "On pinning synchronization of complex dynamical networks," *Automatica*, vol. 45, no. 2, pp. 429–435, 2009.

[6] L. Deng, Z. Wu, and Q. Wu, "Pinning synchronization of complex network with non-derivative and derivative coupling," *Nonlinear Dynamics*, vol. 73, no. 1-2, pp. 775–782, 2013.

[7] W. Yu, P. DeLellis, G. Chen, M. di Bernardo, and J. Kurths, "Distributed adaptive control of synchronization in complex networks," *IEEE Transactions on Automatic Control*, vol. 57, no. 8, pp. 2153–2158, 2012.

[8] L. Lü, M. Yu, C. Li et al., "Projective synchronization of a class of complex network based on high-order sliding mode control," *Nonlinear Dynamics*, vol. 73, no. 1-2, pp. 411–416, 2013.

[9] Y. Zhao, Z. Duan, G. Wen, and G. Chen, "Robust consensus tracking of multi-agent systems with uncertain Lur'e-type nonlinear dynamics," *IET Control Theory & Applications*, vol. 7, no. 9, pp. 1249–1260, 2013.

[10] G. Hu, "Robust consensus tracking of a class of second-order multi-agent dynamic systems," *Systems & Control Letters*, vol. 61, no. 1, pp. 134–142, 2012.

[11] J.-W. Xiao, Y. Huang, Y.-W. Wang, and J. O. Yi, "Synchronization of complex switched networks with two types of delays," *Neurocomputing*, vol. 74, no. 17, pp. 3151–3157, 2011.

[12] T. H. Lee, D. H. Ji, J. H. Park, and H. Y. Jung, "Decentralized guaranteed cost dynamic control for synchronization of a complex dynamical network with randomly switching topology," *Applied Mathematics and Computation*, vol. 219, no. 3, pp. 996–1010, 2012.

[13] T. H. Lee, Ju. H. Park, and Z. Wu, "Robust H_∞ decentralized dynamic control for synchronization of a complex dynamical network with randomly occurring uncertainties," *Nonlinear Dynamics*, vol. 70, no. 1, pp. 559–570, 2012.

[14] D. W. Lee, W. J. Yoo, D. H. Ji, and J. H. Park, "Integral control for synchronization of complex dynamical networks with unknown non-identical nodes," *Applied Mathematics and Computation*, vol. 224, pp. 140–149, 2013.

[15] Q. Song, J. Cao, and F. Liu, "Synchronization of complex dynamical networks with nonidentical nodes," *Physics Letters A*, vol. 374, no. 4, pp. 544–551, 2010.

[16] M. Shi, J. Li, and C. He, "Synchronization of complex dynamical networks with nonidentical nodes and derivative coupling via distributed adaptive control," *Mathematical Problems in Engineering*, vol. 2013, Article ID 172608, 11 pages, 2013.

[17] Y.-Q. Fan, Y.-H. Wang, Y. Zhang, and Q.-R. Wang, "The synchronization of complex dynamical networks with similar nodes and coupling time-delay," *Applied Mathematics and Computation*, vol. 219, no. 12, pp. 6719–6728, 2013.

[18] L. Wang, W. Qian, and Q. Wang, "Bounded synchronization of a time-varying dynamical network with nonidentical nodes," *International Journal of Systems Science*, vol. 44, no. 12, Article ID 815825, 2013.

[19] J. Zhao, D. J. Hill, and T. Liu, "Global bounded synchronization of general dynamical networks with nonidentical nodes," *IEEE Transactions on Automatic Control*, vol. 57, no. 10, pp. 2656–2662, 2012.

[20] J. Zhao, D. J. Hill, and T. Liu, "Stability of dynamical networks with non-identical nodes: a multiple V-Lyapunov function method," *Automatica*, vol. 47, no. 12, pp. 2615–2625, 2011.

[21] F. Liu, Q. Song, and J. Cao, "Improvements and applications of entrainment control for nonlinear dynamical systems," *Chaos*, vol. 18, no. 4, Article ID 043120, 13 pages, 2008.

[22] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, vol. 15 of *SIAM Studies in Applied Mathematics*, SIAM, Philadelphia, Pa, USA, 1994.

[23] J. Hale, *Theory of Functional Differential Equations*, Springer, New York, NY, USA, 1993.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 851403, 8 pages
http://dx.doi.org/10.1155/2014/851403

Research Article

Synchronization in a Novel Local-World Dynamical Network Model

Jianeng Tang and Peizhong Liu

College of Engineering, Huaqiao University, No. 269 Chenghuabei Road, Quanzhou, Fujian 362021, China

Correspondence should be addressed to Peizhong Liu; liupeizhong_1234@126.com

Received 13 February 2014; Accepted 22 April 2014; Published 7 May 2014

Academic Editor: Wenwu Yu

Advances in complex network research have recently stimulated increasing interests in understanding the relationship between the topology and dynamics of complex networks. In the paper, we study the synchronizability of a class of local-world dynamical networks. Then, we have proposed a local-world synchronization-optimal growth topology model. Compared with the local-world evolving network model, it exhibits a stronger synchronizability. We also investigate the robustness of the synchronizability with respect to random failures and the fragility of the synchronizability with specific removal of nodes.

1. Introduction

Complex networks are the sets of interconnected large-scale nodes, in which a node is a fundamental unit that can have different meanings in different situations [1–15]. Complex networks were conventionally researched by graph theory, for which a complex network was described by a random graph, where the radical theory was introduced by Erdös and Rényi [3]. At present, Watts and Strogatz (WS) [4] introduced the conception of small-world networks to describe a transition from a regular lattice to a random graph. Based on the research of Barabási and Albert [5], empirical results display that many large-scale complex networks are scale-free. Remarkably, a scale-free network is essentially inhomogeneous; in other words, the majority of nodes have extremely few connections, but a small number of specific nodes have many connections.

As shown in [5], the BA network model captures the basic mechanism which causes the power-law degree distribution, but the model also has several limitations: it only predicts a fixed exponent in a power-law degree distribution, while the measured real networks' exponents actually vary mostly from 1 to 3, which can have non-power-law features such as exponential cutoffs or saturation for small variables. In order to overcome these limitations and further understand various microscopic processes under the influence of the network topology and evolution, the researchers have done many valuable works about the aspect. The evolution factors may include a kind of sides such as roughly different types of preferential attachments, growths, local events, and competitions. To some extent, some researchers studied a nonlinear preferential attachment scheme with the degree probability; some researchers studied the accelerated growth in a directed network; the others also investigated the competition aspect and the distance preference. Some researchers also studied the propagation mechanism and the synchronization principle of complex network models. For details, please refer to the relevant literature [6–15].

Synchronization of complex networks has been a subject of intensive research with potential applications the Internet, the World Wide Web, food webs, electric power grids, and cellular and metabolic networks. On one hand, some researchers proposed some novel control laws to study the synchronization of complex networks. A novel impulsive control law was proposed for synchronization of stochastic discrete complex networks [16]. In [17], a simple but effective pinning algorithm for reaching synchronization on a general complex dynamical network was proposed. In recent years, researchers discussed the consensus in multiagent dynamical systems [18–20]. On the other hand, network topology structure provides a powerful metaphor for describing sophisticated collaborative dynamics of many practical systems in essence. So some researchers proposed some new network models to study the synchronization of complex networks.

Local-world evolving network model was proposed in [21]. It captured an important feature in the evolution of many real-world complex networks: preferential attachment mechanism works only within local-world instead of whole network-wide. The degree distribution of this local-world evolving network model represented a transition between that of an exponential network and of a power-law scaling network, and the synchronization robustness and fragility of the local-world network model also displayed a transition between the exponential and the scale-free ones. According to the local-world evolving model, Sun et al. [22] studied the statistical properties of networks constructed and found that local-world size M had great effect on network's connectivity: bigger M made the networks more heterogeneous in connectivity. A comprehensive multilocal-world model was proposed in [23]. Gu and Sun [24] proposed a local-world node deleting evolving network model. In this paper, the objective is to find a new local area network model in which there is stronger synchronizability. Thence, a local-world synchronization-optimal dynamical network model is proposed to describe the topology of evolving networks. Compared with the local-world evolving network model, the proposed network model exhibits a stronger synchronizability.

The rest of the paper is organized as follows: Section 2 introduces synchronization stability criterion of complex dynamical networks. Then, we study a brief summary about the local-world evolving network model and propose the novel local-world dynamical network model followed by some discussions on its synchronization robustness and fragility in Section 4. Finally, Section 5 concludes the investigation.

2. Synchronization Stability Criterion of Complex Dynamical Networks

Consider a dynamical network consisting of N linearly and diffusively identical coupled nodes, with each node being an n-dimensional dynamical system. The state equations of the network can be written as [6, 25].

$$\dot{x}_i = f(x_i) + c\sum_{j=1}^{N} a_{ij}\Gamma x_j, \quad i = 1, 2, \ldots, N, \tag{1}$$

where $x_i = (x_{i1}, x_{i2}, \ldots, x_{in}) \in R^n$ are the state variables of node i and the constant $c > 0$ represents the coupling strength. For simplicity, we take $\Gamma = \mathrm{diag}\{r_1, r_2, \ldots, r_n\} \in R^{n\times n}$ and Γ is a diagonal matrix with $r_i = 1$, for a particular i, and $r_i = 0$, for $j \neq i$. If there is a connection between node i and node j $(i \neq j)$, then $a_{ij} = a_{ji} = 1$; otherwise, $a_{ij} = a_{ji} = 0$ $(i \neq j)$. We take

$$a_{ii} = -\sum_{j=1, j\neq i}^{N} a_{ij} = -k_i, \quad i = 1, 2, \ldots, N, \tag{2}$$

where the degree k_i of node i is defined to be the number of connection incidents on node i.

The coupling matrix $A = (a_{ij})_{N\times N} \in R^{N\times N}$ represents the coupling configuration of the network. Suppose that the network is connected in the sense that there are no isolate clusters. The A is a symmetric and irreducible matrix. In this case, it can be shown that zero is an eigenvalue of A with multiplicity 1 and all the other eigenvalues of A are strictly negative [26].

Dynamical network (1) is said to be (asymptotically) synchronized, if

$$x_1(t) = x_2(t) = \cdots x_N(t) = s(t) \quad as\ t \longrightarrow \infty, \tag{3}$$

where $s(t)$ is a solution of an isolated node, which can be an equilibrium point, a periodic orbit, or a chaotic attractor, depending on the interest of study.

Let $0 = \lambda_1 > \lambda_2 \geq \lambda_3 \geq \cdots \lambda_N$ be the eigenvalues of the coupling matrix A. Consider network (1); suppose that there exist an $n \times n$ diagonal matrix $\Lambda > 0$ and two constants $\overline{d} < 0$ and $\tau > 0$, such that

$$[Df(s(t)) + d\Gamma]^T\Lambda + \Lambda[Df(s(t)) + d\Gamma] \leq -\tau I_n, \tag{4}$$

for all $d \leq \overline{d}$, where $I_n \in R^{n\times n}$ is an unit matrix and $Df(s(t))$ is the Jacobian of f at $s(t)$. It has been shown that the synchronized state (3) is exponentially stable, if [12]

$$c \geq \left|\frac{\overline{d}}{\lambda_2}\right|. \tag{5}$$

Note that the criterion (5) may not hold, if the dynamical equations of a network cannot be written in the form of (2) [27].

Given the dynamics of an isolated node, the synchronizability of network (1) with respect to a specific coupling configuration A is said to be strong, if the network can synchronize with a small coupling strength c. Inequality (5) implies that the synchronizability of network (1) can be characterized by the second-largest eigenvalue of its coupling matrix; that is, the smaller the second-largest eigenvalue, the stronger the synchronizability of a network.

Duan et al. analyzed complex network synchronizability using the theory of subgraphs and complementary graphs [28, 29]. As shown in the literature [30], the local stability of the synchronized solution can be determined by analyzing the so-called master stability equation. If the synchronized region is an unbounded sector $(-\infty, \alpha)$, the eigenvalue λ_2 of A determines the synchronizability [25]; if the synchronized region is a bounded sector $[\alpha_1, \alpha_2]$, the ratio $r(A) = \lambda_2/\lambda_N$ determines the synchronizability [27]. And Duan et al. proposed a design method for the inner linking matrix of rank 1 such that the resultant network had an unbounded synchronized region because an unbounded synchronized region was always easier to analyze than a bounded synchronized region. Obviously, the synchronized region is unbounded in our paper. In the paper [31], the authors proposed novel criteria of synchronization state for both delay-independent and delay-dependent stabilities of linear time-delay systems.

3. Local-World Dynamical Network Models

3.1. The Local-World Evolving Network Model. In many real-life networks, owing to the existence of the local-world connectivity discussed above, each node in a network has only local connections and, therefore, only owns local information about the entire network. To model such a local-world effect, a local-world evolving network model (LW) is proposed and to be generated by the following algorithm [21].

(i) Start with a small number m_0 of nodes and small number e_0 of edges.

(ii) Select M nodes randomly from the existing network, referred to as the "local world" of the new coming node.

(iii) Add a new node with m edges, linked to m nodes in its local world determined in (ii), using a preferential attachment with probability $\Pi(k_i)$ defined at every time step t by

$$\prod(k_i) = \frac{M}{m_0 + t} \frac{k_i}{\sum_{j \in \text{local}} k_j}. \quad (6)$$

After t time steps, this procedure results in a network with $N = t + m_0$ nodes and $E = e_0 + mt$ edges.

3.2. The Local-World Node Deleting Evolving Network Model. In the local-world node deleting evolving network (LWD network), an undirected and unweighted network is initialized with a small number m_0 of isolated nodes. The network is evolved with the following scheme [24].

At each time step t, either we act (i) with probability p_a or we act (ii) with probability $1 - p_a$.

(i) Node adding: the addition is achieved as follows:

(1) growth: add a new node with m ($m < m_0$) edges connected to the network;

(2) local-world establishment: randomly select M nodes from the whole network as the local world;

(3) preferential attachment: add m edges between the new coming node and m existing nodes in the local-world; the probability for node i selected in the local world is

$$\prod(k_i) = \frac{M}{N(t)} \frac{k_i}{\sum_{j \in \text{local}} k_j}, \quad (7)$$

where $N(t)$ is the total number of nodes after t time steps.

(ii) Node deleting: delete a node from the network randomly and remove all the edges once attached to the deleting node.

3.3. The Local-World Synchronization-Optimal Dynamical Network Model. In this paper, a novel local-world dynamical network model called the local-world synchronization-optimal dynamical network model (LWSO) is proposed. Then, the new network generation algorithm of the model is as follows.

(i) Start with a small number m_0 of nodes and small number e_0 of edges.

(ii) Select M nodes randomly from the existing network, referred to as the "local world" of the new coming node.

(iii) Add a new node with m edges, linked to m nodes in its local world determined in (ii). The criterion for choosing the m vertices to which the new vertex connects is to optimize the synchronizability of the obtained network or, equivalently, to minimize the second-largest eigenvalue of the corresponding coupling matrix.

There are q ($q = C_M^m$) types about choosing m nodes from M nodes. After t time steps this procedure results in a network with $N = t + m_0$ nodes and $E = e_0 + mt$ edges.

The network topology has significant effects on its traffic protocols, searching algorithms, and even virus propagation; therefore, modeling the network topology is extremely important. Currently, there are quite some models proposed and applied to describe the network topological features and properties, such as the BA [5], LW [21], and LWD [24] models. For simplicity, we take $N = 1000$, $m = 3$, $m_0 = 10$, $M = 10$, and $p_a = 0.7$. The details are as shown in Figure 1.

Donetti et al. proposed an optimization algorithm to minimize the eigenvalue ratio [32]. According to the paper, we know that the synchronizability of a network becomes stronger and stronger when the eigenvalue ratio is smaller and smaller. In the paper, the rewirings are accepted in the existing networks, such as random graph, small-world, linear chain, and scale-free. In our paper, we have studied a novel growth topology model. In the model, the preferential attachment is as follows: the criterion for choosing the m vertices to which the new vertex connects is to optimize the synchronizability of the obtained network or, equivalently, to minimize the second-largest eigenvalue of the corresponding coupling matrix. Obviously, the preferential attachment is similar to the optimization algorithm. The difference between the preferential attachment and the optimization algorithm is as follows: the former is applied to the growth of networks but the latter is applied to the existing large-scale networks. But their main drawback is that the calculation of eigenvalues is slow.

We have known that the connectivity of the BA scale-free network is heterogeneous. Most vertices have few connections and a small number of vertices have many connections. We have also found that local-world networks are topological quasimulticenter networks. There is a number of the "hubs" which are almost connected with all of vertices, but most of the vertices have very few connections. However, what about the topological structure of this local-world synchronization-optimal network model? It has been pointed out that the eigenvalues spectrum of complex networks provides information about their structural properties and the quantity $R = (\lambda_1 - \lambda_2)/(\lambda_2 - \lambda_N)$ measures the distance of the first eigenvalue from the remaining part of the spectral density normalized by the extension of the remaining part [33].

As shown in Figure 2, the figure represents the values of R of the BA networks, the LW networks, the LWD networks,

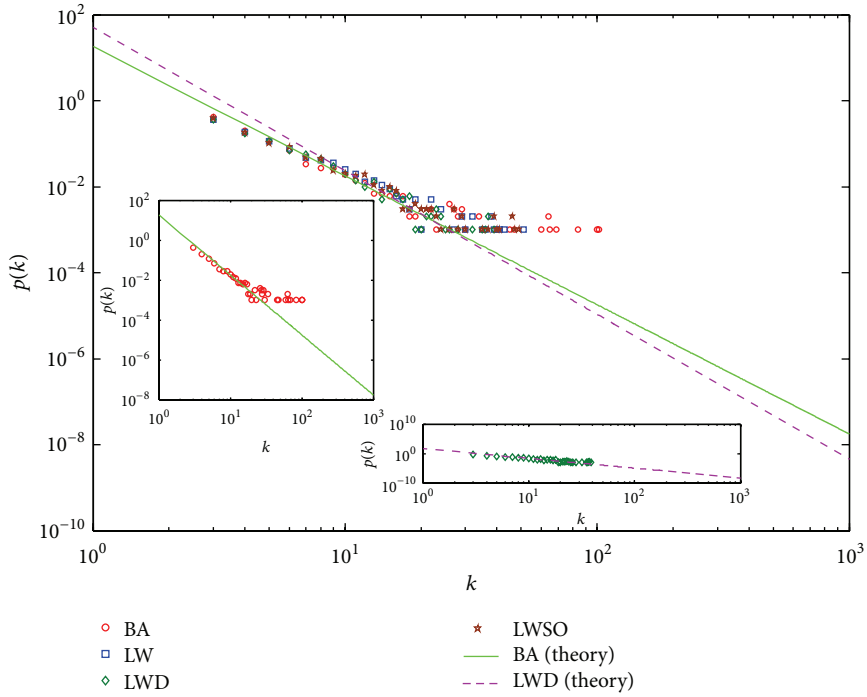


Figure 1: The degree distribution of nodes $p(k)$ of the BA networks (circles), the LW networks (squares), the LWD networks (diamonds), and the LWSO networks (pentagrams). In addition, the real line is $p(k)$ of BA model and the dash line is the theoretical prediction of $p(k)$ of LWD network. The right inset shows the degree distribution of the BA networks and the theoretical prediction; the left inset shows the LWD part.

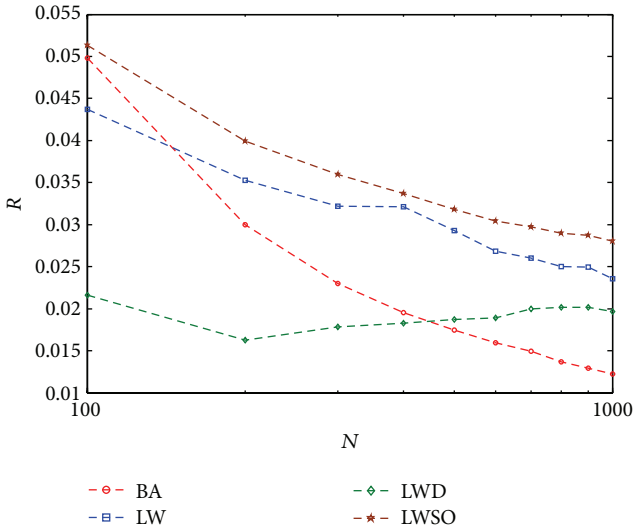


Figure 2: The values of R for the BA networks (circles), the LW networks (squares), the LWD networks (diamonds), and the LWSO networks (pentagrams). Each curve in the figure is the average result of 5 groups of networks.

and the LWSO networks, while the size N of networks ranges from 100 to 1000. It can be observed that as the network size N increases, values of R of the three networks decay to converge to a power law. But values of R of the LWD networks do not display the characteristic because the node deletion deletes a large portion of potential hubs during the network evolution. Then the value of R of the LWSO dynamical network model changes more slowly. It explains that R spans are the widest in the three models. The variation tendencies of R correlate with different values of m, which increase as m increases in extensive simulations.

3.4. Synchronization in Dynamical Networks. For clarity, we take $M = m_0$ in the construction of the four models. Then A_{BA}, A_{LW}, A_{LWD}, and A_{LWSO} represent the coupling matrices of the dynamical network (1) with the BA evolving network model, the LW evolving network model, the LWD evolving network model, and the LWSO dynamical network model, respectively, which has N nodes and $N(N - M) + e_0$ connections. Let $\lambda_{2\mathrm{BA}}$, $\lambda_{2\mathrm{LW}}$, $\lambda_{2\mathrm{LWD}}$, and $\lambda_{2\mathrm{LWSO}}$ be the second-largest eigenvalues of A_{BA}, A_{LW}, A_{LWD}, and A_{LWSO}, respectively. In numerical computations, the eigenvalues are obtained by averaging the results of 5 runs. For a fixed value of m and M, the phenomenon found out that $\lambda_{2\mathrm{BA}}$ decreases to a negative constant $\overline{\lambda}_{2\mathrm{BA}}$, as N increases. At the same time, $\lambda_{2\mathrm{LW}}$, $\lambda_{2\mathrm{LWD}}$, and $\lambda_{2\mathrm{LWSO}}$ decrease to a negative constant $\overline{\lambda}_{2\mathrm{LW}}$, $\overline{\lambda}_{2\mathrm{LWD}}$, and $\overline{\lambda}_{2\mathrm{LWSO}}$, respectively. For simplicity, we take $N = 1000$, $m = 3$, and $M = 10$. The details are as shown in "Table 1" and in Figure 3.

As shown in Figure 3, the four second-largest eigenvalues converge to four negative constants, as N increases. It has been observed that the second-largest eigenvalue of the

Table 1: A comparison of the second-largest eigenvalues of different coupling topologies.

N	m	M	$\overline{\lambda}_{2BA}$	$\overline{\lambda}_{2LW}$	$\overline{\lambda}_{2LWD}$	$\overline{\lambda}_{2LWSO}$
1000	3	10	−1.2586	−1.2205	−0.7762	−1.4052

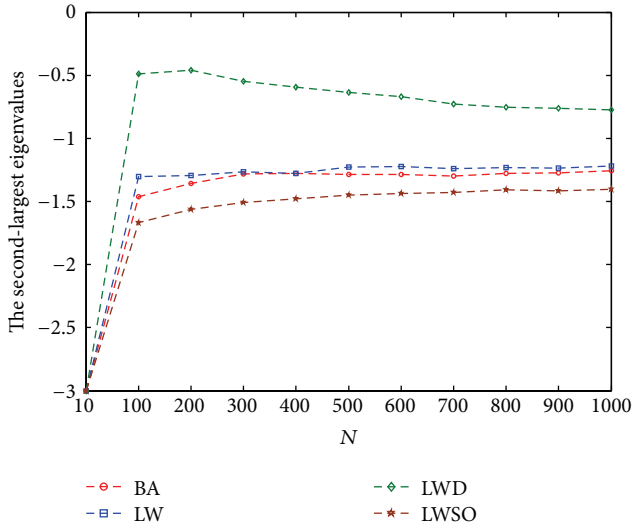


Figure 3: The second-largest eigenvalues of the coupling matrixes for the BA networks (circles), the LW networks (squares), the LWD networks (diamonds), and the LWSO networks (pentagrams). Each curve in the figure is the average result of 5 groups of networks.

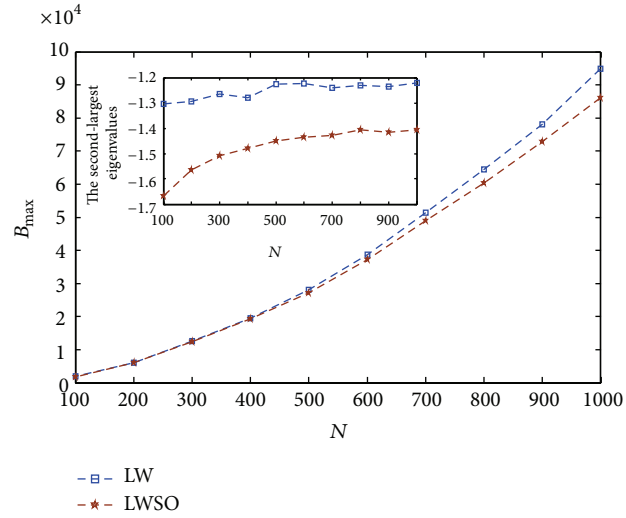


Figure 4: The values of B_{max} for the LW networks (squares) and the LWSO networks (pentagrams). The inset shows the corresponding second-largest eigenvalues. Each curve in the figure is the average result of 5 groups of networks.

LWSO network is smallest in the four networks, which indicates that the synchronizability of the LWSO network model is the strongest. With the exclusion of the LWD network, the eigenvalues of the other networks gradually become small. The eigenvalues of the LWD network are firstly larger and then they get smaller because the node deletion deletes a large portion of potential hubs during the network evolution.

Recently, more and more attention has been paid to the relation between the complex dynamic network topology characteristic and the network synchronizability by study scholars. The recent research work has discovered that many factors have had different influence on the network synchronizability, such as the maximum degree, the average way length, and the degree distribution. In [31], the effects of the maximum betweenness centrality B_{max} on the network synchronizability appear to be as follow: synchronizability is always improved, as B_{max} is reduced. Therefore, the betweenness centrality is proposed as a suitable indicator for predicting synchronizability on complex networks. On one hand, we can see that the second-largest eigenvalues of the LWSO network are smaller than those of the LW network as shown in Figure 3. Inequality (5) implies that the synchronizability of network (1) can be characterized by the second-largest eigenvalue of its coupling matrix; that is, the smaller the second-largest eigenvalue, the stronger the synchronizability of a network. Then, it is clear that the synchronizability of the LWSO network is stronger than that of the LW network. On the other hand, we also notice, in Figure 4, that the maximum betweenness centrality of the LWSO network is smaller than that of the LW network. Accordingly, the view in [34] has been validated through the foregoing two simulations.

Here, it is worthwhile to emphasize that we have found some evidence indicating that there may exist some common features between synchronization and network traffic on a dynamical level [35–40]. Many previous works focus on the relationship between the distribution of the betweenness centrality and the capability of communication networks, with a latent assumption that the information packets go along the shortest paths from source to destination. Hence, the betweenness centrality is always considered as a static topological measure of networks. Here, we discover that this quantity is determined both by the routing algorithm and network topology; thus, one should pay more attention to the design of network topology. We believe this work may be helpful for understanding the intrinsic mechanism and the capability of network traffic.

4. Robustness and Fragility

Now, we consider the robustness of synchronization in dynamical network (1) against either random or specific removal of a small fraction f $(0 < f < 1)$ of the nodes in the network. Clearly, the removal of some nodes in a network (1) will change its coupling matrix. However, if the second-largest eigenvalue of the coupling matrix remains unchanged, then the synchronizability of the network will also remain unchanged after the removal of some of its nodes.

Let $A \in R^{N\times N}$ and $B \in R^{(N-[fN])\times(N-[fN])}$ be the coupling matrices of the original network with N nodes and the new network after removal of $[fN]$ nodes, respectively. Denote $\lambda_{A3}$and λ_{B3} as the second-largest eigenvalues of A and B,

respectively. Suppose that nodes i_1th, i_2th, ..., $i_{([fN])}$th have been removed from the network. One can construct the new coupling matrix B from the original coupling matrix A as follows [6]:

(i) from the minor matrix $B \in R^{(N-[fN])\times(N-[fN])}$ of A by removing the i_1th,i_2th, ..., $i_{[fN]}$ row-column pairs of A;

(ii) obtain $B = (b_{ij})_{(N-[fN])\times(N-[fN])}$ by recomputing the diagonal elements of the minor matrix $G = (g_{ij})$ as follows:

$$g_{ij} = b_{ij}, \quad i \neq j;$$
$$g_{ii} = -\sum_{j=1, j\neq i}^{N-[fN]} b_{ij}, \quad i = 1, 2, \ldots, N - [fN]. \tag{8}$$

Here, we mainly compare the robustness and fragility between the LW network and the LWSO network. We know that the BA network is a global network. In addition, the node deletion will delete a large portion of potential hubs during the network evolution. It is harmful to the network evolution. In the further study, how to delete the nodes instead of random deletion is an important problem. In the simulation, for simplicity, we take $N = 1000$, $m = 3$, and $M = m_0 = 4, 10$, respectively. Then we notice that "LW43" represents the LW model of $M = 4$ and $m = 3$. We also notice that "LWSO43" represents the LWSO dynamical network model of $M = 4$ and $m = 3$. At the same time, "LW103" and "LWSO103" represent the similar meaning. Then, those symbols stand for the same meaning in the following figures. The original network with the local-world or local-world synchronization-optimal topology contains 1000 nodes and about 3000 connections. We have known from [25] that the synchronizability of the original coupled network remains almost unchanged when a scale-free dynamical network has a random failure; for example, a very small fraction f of nodes are randomly removed; however, the synchronizability of the original coupled network is also significantly decreased or even destroyed when the network is attacked intentionally; for example, a small fraction f of "big" nodes are removed and then the original network changes significantly and even breaks into parts. These tell us that a scale-free network regarding its dynamical synchronization has the meanings of "robustness and yet fragility."

It has been shown, as in Figure 5, that the second-largest eigenvalues of the "LW103" have decreased from −1.2205 to −0.9046 when as many as 5% of the randomly chosen nodes are removed. Then, the decreased magnitude of the second-largest eigenvalues of the "LWSO103" is almost 32.83 percent. When more vertices are randomly removed, it is not significant that the reduction in the second-largest eigenvalue of the coupling matrix. This means that the "LW103" is more robust than the "LWSO103" against random failures. Then, we consider the fragility of the synchronizability with respect to deliberate attacks. A few isolated vertices or clusters may take place during the process of deliberate attacks. We remove the vertices with the highest degree and find that the second-largest eigenvalues of the "LWSO103" reduce from −1.4052

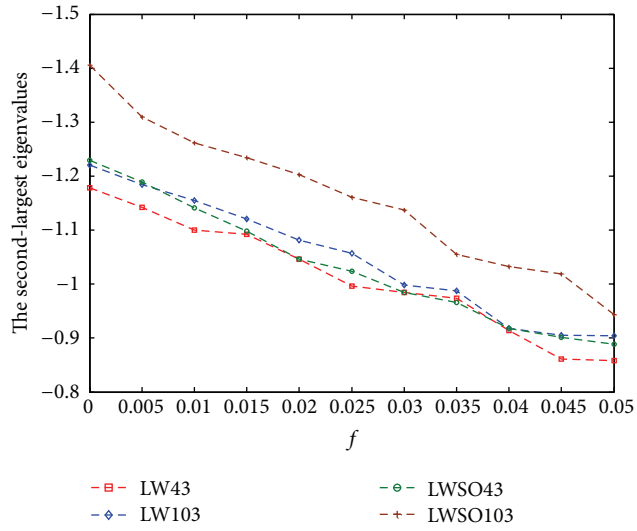


Figure 5: Synchronization robustness against random failures: changes of the second-largest eigenvalues of the local-world dynamical networks ($M = 4$, squares), the local-world synchronization-optimal dynamical networks ($M = 4$, circles), the local-world dynamical networks ($M = 10$, diamonds), and the local-world synchronization-optimal dynamical networks ($M = 10$, pluses). Each curve in the figure is the average result of 5 groups of networks.

Table 2: A comparison of the decreased magnitudes of the second-largest eigenvalues when 5% of the nodes are respect to random and specific removed.

	Robustness	Fragility
LW43	27.66%	71.91%
LWSO43	27.73%	67.01%
LW103	25.88%	92.79%
LWSO103	32.83%	86.5%

to −0.1895 when about 5% of the most connected vertices are removed. As shown in Figure 6, the decreased magnitude of the second-largest eigenvalues of "LW103" is about 92.79 percent. This implies that the "LWSO103" is less vulnerable to specific removal of those most connected vertices than the "LW103." At the same time, the comparison situation between the "LW43" and the "LWSO43" is similar, but the analysis result is less obvious than the above analysis result. The details are as shown in "Table 2." To sum up, the local-world network is particularly well suited to tolerate random errors compared with the local-world synchronization-optimal dynamical network. However, the latter is particularly well suited to tolerate intentional attacks compared with the former.

5. Conclusion

In the past decade, researchers have gained the significant progress about the effect of network topology on network synchronization dynamical behavior. In the paper, we study the synchronizability of a class of local-world dynamical networks. Then, we have proposed a local-world

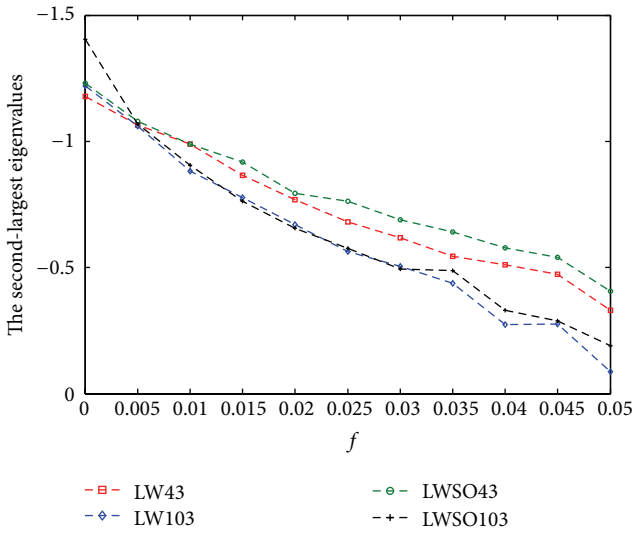


FIGURE 6: Synchronization fragility against specific attacks: changes of the second-largest eigenvalues of the local-world dynamical networks ($M = 4$, squares), the local-world synchronization-optimal dynamical networks ($M = 4$, circles), the local-world dynamical networks ($M = 10$, diamonds), and the local-world synchronization-optimal dynamical networks ($M = 10$, pluses). Each curve in the figure is the average result of 5 groups of networks.

synchronization-optimal growth topology model. Compared with the local-world evolving network model, it exhibits stronger synchronizability. We have found some evidence indicating that there may be some common features between synchronization and network traffic on a dynamical level. So this work may be helpful for understanding the intrinsic mechanism and the capability of network traffic. Then, we also investigate the robustness of the synchronizability with respect to random failures and the fragility of the synchronizability with specific removal of nodes. We know that the local-world network is particularly well suited to tolerate random errors compared to the local-world synchronization-optimal dynamical network according to the simulation. However, the latter is particularly well suited to tolerate intentional attacks compared with the former.

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 Natural Science Foundation of China under Grant no. 61231002 and no. 51075068, by Cloud Computing Platform for Internet of Things, Fujian Scientific Research Platform for Innovation under Grant no. 2013H2002, by the Fundamental Research Funds for the Central Universities under Grant no. JB-ZR1202, by the Foundation of Quanzhou under Grant no. 24201305, and by the Foundation of Huaqiao University under Grants no. 12BS228, no. 12Y0316, and no. 13BS103. The authors would like to thank the reviewers for their valuable suggestions and comments.

References

[1] S. H. Strogatz, "Exploring complex networks," *Nature*, vol. 410, pp. 268–276, 2001.

[2] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, pp. 47–97, 2002.

[3] P. Erdös and A. Rényi, "On the evolution of random graphs," *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, vol. 5, pp. 17–61, 1959.

[4] D. J. Watts and S. H. Strogatz, "Collective dynamics of "small-world" networks," *Nature*, vol. 393, pp. 440–442, 1998.

[5] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *American Association for the Advancement of Science. Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[6] J. Fan and X. F. Wang, "On synchronization in scale-free dynamical networks," *Physica A*, vol. 349, no. 3-4, pp. 443–451, 2005.

[7] A. Arenas, D.-G. Albert, J. Kurths, Y. Moreno, and C. S. Zhou, "Synchronization in complex networks," *Physics Reports*, vol. 469, no. 3, pp. 93–153, 2008.

[8] J. N. Tang, C. R. Zou, L. Zhao, X. Z. Xu, and X. M. Du, "Impulsive stabilization for control and synchronization of complex networks with coupling delays," *Journal of the Physical Society of Japan*, vol. 81, Article ID 014003, 7 pages, 2012.

[9] M. J. Van den Hof Paul, D. Arne, S. C. H. Peter, and B. Xavier, "Identification of dynamic models in complex networks with prediction error methods-Basic methods for consistent module estimates," *Automatica*, vol. 49, no. 10, pp. 2994–3006, 2013.

[10] C.-H. Li, C.-C. Tsai, and S.-Y. Yang, "Analysis of epidemic spreading of an SIRS model in complex heterogeneous networks," *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 4, pp. 1042–1054, 2014.

[11] J. N. Tang, C. R. Zou, S. P. Wang, L. Zhao, and X. P. Liu, "Chaos synchronization of chen systems with time-varying delays," *International Journal of Bifurcation and Chaos*, vol. 22, Article ID 1250183, 8 pages, 2012.

[12] G. Bianconi and A. L. Barabási, "Competition and multiscaling in evolving networks," *Europhysics Letters*, vol. 54, no. 4, p. 436, 2001.

[13] L. K. Tang, J. Lu, and G. Chen, "Synchronizability of small-world networks generated from ring networks with equal-distance edge additions ," *Chaos*, vol. 22, Article ID 023121, 2012.

[14] G. H. Zhu, X. C. Fu, and G. Chen, "Spreading dynamics and global stability of a generalized epidemic model on complex heterogeneous networks ," *Applied Mathematical Modelling*, vol. 36, no. 12, pp. 5808–5817, 2012.

[15] L. K. Tang, J. Lu, J. Lü, and X. Q. Wu, "Impact of node dynamics parameters on topology identification of complex dynamical networks," *Nonlinear Dynamics*, vol. 73, no. 1-2, pp. 1081–1097, 2013.

[16] C. J. Li, W. W. Yu, and T. W. Huang, "Impulsive synchronization schemes of stochastic complex networks with switching topology: average time approach ," *Neural Networks*, vol. 54, pp. 85–94, 2014.

[17] W. W. Yu, G. R. Chen, J. H. Lü, and J. Kurths, "Synchronization via pinning control on general complex networks," *SIAM*

Journal on Control and Optimization, vol. 51, no. 2, pp. 1395–1416, 2013.

[18] Y. Zhao, Z. S. Duan, G. H. Wen, and Y. J. Zhang, "Distributed finite-time tracking control for multi-agent systems: an observer-based approach," *Systems & Control Letters*, vol. 62, no. 1, pp. 22–28, 2013.

[19] W. W. Yu, G. R. Chen, and M. Cao, "Some necessary and sufficient conditions for second-order consensus in multi-agent dynamical systems," *Automatica*, vol. 46, no. 6, pp. 1089–1095, 2010.

[20] Y. Zhao, Z. Li, and Z. Duan, "Distributed consensus tracking of multi-agent systems with nonlinear dynamics under a reference leader," *International Journal of Control*, vol. 86, no. 10, pp. 1859–1869, 2013.

[21] X. Li and G. Chen, "A local-world evolving network model," *Physica A*, vol. 328, no. 1-2, pp. 274–286, 2003.

[22] S. W. Sun, Z. X. Liu, Z. Q. Chen, and Z. Z. Yuan, "Error and attack tolerance of evolving networks with local preferential attachment," *Physica A*, vol. 373, pp. 851–860, 2007.

[23] Z. P. Fan, G. Chen, and Y. N. Zhang, "A comprehensive multi-local-world model for complex networks," *Physics Letters A*, vol. 373, no. 18-19, pp. 1601–1605, 2009.

[24] Y. Y. Gu and J. T. Sun, "A local-world node deleting evolving network model," *Physics Letters A*, vol. 372, no. 25, pp. 4564–4568, 2008.

[25] X. F. Wang and G. R. Chen, "Synchronization in scale-free dynamical networks: robustness and fragility," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 49, no. 1, pp. 54–62, 2002.

[26] C. W. Wu and L. O. Chua, "Synchronization in an array of linearly coupled dynamical systems," *IEEE Transactions on Circuits and Systems I Fundamental Theory and Applications*, vol. 42, no. 8, pp. 430–447, 1995.

[27] M. Barahona and L. M. Pecora, "Synchronization in small-world systems," *Physical Review Letters*, vol. 89, Article ID 054101, 2002.

[28] Z. S. Duan, G. R. Chen, and L. Huang, "Complex network synchronizability: analysis and control," *Physical Review E*, vol. 76, Article ID 056103, 2007.

[29] Z. S. Duan, C. Liu, and G. R. Chen, "Network synchronizability analysis: the theory of subgraphs and complementary graphs," *Physica D*, vol. 237, no. 7, pp. 1006–1012, 2008.

[30] L. M. Pecora and T. L. Carroll, "Master stability functions for synchronized coupled systems," *Physical Review Letters*, vol. 80, p. 2109, 1998.

[31] Q. Y. Wang, G. R. Chen, Q. S. Lu, and F. Hao, "Novel criteria of synchronization stability in complex networks with coupling delays," *Physica A*, vol. 378, no. 2, pp. 527–536, 2007.

[32] L. Donetti, P. I. Hurtado, and M. A. Muñoz, "Entangled networks, synchronization, and optimal network topology," *Physical Review Letters*, vol. 95, Article ID 188701, 2005.

[33] I. J. Farkas, I. Derenyi, A. L. Barabasi, and T. Vicsek, "Spectra of "real-world" graphs: beyond the semicircle law," *Physical Review E*, vol. 64, Article ID 026704, 2001.

[34] T. Nishikawa, A. E. Motter, Y.-C. Lai, and F. C. Hoppensteadt, "Heterogeneity in oscillator networks: are smaller worlds easier to synchronize?" *Physical Review Letters*, vol. 91, Article ID 014101, 2003.

[35] A. E. Motter, C. Zhou, and J. Kurths, "Network synchronization, diffusion, and the paradox of heterogeneity," *Physical Review E*, vol. 71, Article ID 016116, 2005.

[36] M. Chavez, D.-U. Hwang, A. Amann, H. G. E. Hentschel, and S. Boccaletti, "Synchronization is enhanced in weighted complex networks," *Physical Review Letters*, vol. 94, Article ID 218701, 2005.

[37] H.-T. Zhang, T. Yu, J.-P. Sang, and X.-W. Zou, "Dynamic fluctuation model of complex networks with weight scaling behavior and its application to airport networks ," *Physica A*, vol. 393, pp. 590–599, 2014.

[38] M. Zhao, T. Zhou, B.-H. Wang, and W.-X. Wang, "Enhanced synchronizability by structural perturbations," *Physical Review E*, vol. 72, Article ID 057102, 2005.

[39] T. Zhou, M. Zhao, and B.-H. Wang, "Better synchronizability predicted by crossed double cycle," *Physical Review E*, vol. 73, Article ID 037101, 2006.

[40] C.-Y. Yin, B.-H. Wang, W.-X. Wang, T. Zhou, and H.-J. Yang, "Efficient routing on scale-free networks based on local information," *Physics Letters A*, vol. 351, no. 4-5, pp. 220–224, 2006.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 216283, 10 pages
http://dx.doi.org/10.1155/2014/216283

Research Article

Unified Finite Horizon H_∞ Fusion Filtering for Networked Dynamical System

Chenglin Wen,[1,2] Xiaoliang Feng,[1] and Jingjing Yan[1]

[1] *College of Electrical Engineering, Henan University of Technology, Zhengzhou 450001, China*
[2] *College of Automatic, Hangzhou Dianzi University, Hangzhou 310018, China*

Correspondence should be addressed to Xiaoliang Feng; fengxl2002@163.com

Received 5 February 2014; Accepted 11 March 2014; Published 27 April 2014

Academic Editor: Guanghui Wen

This paper addresses the H_∞ fusion filtering problem for networked dynamical systems, where measurements may arrive at fusion center in four different scenes and the fusion center could receive none, one, or multiple measurements in a fusion period. A unified H_∞ performance criterion function, which is suitable for different measurement arrival scenes, is designed for the filtering process of networked dynamical systems. Then, the H_∞ performance criterion function is described as an indefinite quadratic inequality and solved by a novel noise projection method in Krein space. On this basis, a unified finite horizon H_∞ filtering method is proposed for networked dynamical systems. Simulation results are provided to illustrate the correctness and the effectiveness of the theoretical analysis.

1. Introduction

The filtering methods are widely utilized in the fields of signal processing and automatic control for dynamical systems. With the development of computer and information technology, researchers begin to pay more and more attention on networked dynamical systems, such as the open channel networks and networked control systems [1–3]. However, it is inevitable that the measurement data is transmitted in the networked dynamical systems with different time delay.

In networked dynamical systems, the targets of interest are (remotely) observed by various sensors. The sampled measurements may arrive at information processing centers (especially refer to fusion filters, as in this paper) in different scenes through the transport network. Scene 1: the measurement arrives at the fusion filter in time, which is abbreviated to "ITM" in this paper. Scene 2: the measurement which arrives at the fusion filter with some time delay, but still in the sampled sequence, is abbreviated to "ISDM." Scene 3: the delay measurement arrives at the fusion filter out of the sampled sequence, which is abbreviated to "OOSM." Scene 4: the sampled measurement is missing in the transmitting process, which is also named as "packet dropout" (abbreviated to "PD"). The traditional filters are mainly proposed for systems with all measurements in Scene 1, such as Kalman filter, H_∞ filter. For the system with measurements arriving at fusion filter in other scenes, several effective filtering methods have also been proposed, recently.

(1) For systems with measurement in Scene 2, some novel filtering methods are proposed based on Kalman filter [4, 5]. And the developed H_∞ filtering approaches are also deduced for this kind of systems with bounded energy noises [6, 7].

(2) For systems with measurement in Scene 3, several OOSM filtering problems are investigated with the help of such technologies as nonstandard smoothing [8], Kalman filter with measurement weighted summation [9, 10], and reorganized innovation [11, 12].

(3) For systems with measurement in Scene 4, several filtering approaches are developed based on the traditional Kalman filter [13–15] or H_∞ filter [16], based on different descriptions of the packet dropout phenomenon, such as the Markovian jump approach [13] and the binary Bernoulli distribution approach [14–16].

Although some recent approaches have considered the systems in multiple measurement arrival scenes [17, 18], most results of them are deduced for the system with single sensor. Few papers address the filtering problems for the networked dynamical system with multiple sensors. In [19, 20], the fusion filtering methods for networked multisensor systems are deduced based on Kalman filter, which requires the system noise to satisfy zero mean Gaussian distribution with known variance, which, however, is usually not available. To the best of the author's knowledge, the filtering problem for the networked multisensor system with unknown statistic noises has not been fully investigated and still remains challenging.

Motivated by the above discussion, a unified finite horizon H_∞ filtering method is proposed for the networked dynamical system in this paper in which four different kinds of measurement arrival scenes are dealt with in a unified manner. Because of the complex arrival scenes of networked measurements, the fusion filter for the networked dynamical system could receive none, one, or multiple measurements in a fusion period. The H_∞ filtering algorithm should be deduced to achieve a H_∞ performance criterion function. In the traditional H_∞ performance criterion function, an ideal assumption condition is that the measurement sampling time is the same as the measurement arrival time and the filtering time. However, in networked dynamical systems, the arrival time of a networked measurement mostly is not equal to its sampling time. And the fusion filter deals with the sampled measurement at its arrival time, rather than its sampling time. It means that the traditional H_∞ performance criterion function cannot be applied to networked dynamical systems. In this paper, a novel unified H_∞ performance criterion function is built for the different measurement arrival scenes in networked dynamical systems, firstly. Secondly, the H_∞ performance criterion function is described as an indefinite quadratic inequality. The stationary point of indefinite quadratic form in Hilbert space corresponds to a projection in Krein space. In this paper, the stationary point of the indefinite quadratic inequality is obtained by solving a projection in Krein space. However, because of the random delay of networked measurements, the process of solving the projection with the delay measurement becomes more complex. Thirdly, a noise projection approach in Krein space is proposed to solve the projection corresponding to the stationary point. Then, a unified H_∞ filtering method is proposed for the networked dynamical system. Finally, the validity and effectiveness of the proposed method are verified in the final simulation.

The remainder of this paper is organized as follows. The problem of the fusion filtering for the networked dynamical system is formulated in Section 2. In Section 3, a unified finite horizon H_∞ filter is deduced based on a novel performance criterion function for the networked dynamical system in various measurement arrival scenes. An example for illustration is given in Section 4, and we conclude this paper in Section 5.

Notation. The elements in Hilbert space are denoted by bold face letters, such as "$\mathbf{x}, \widehat{\mathbf{x}}$," and the elements in Krein space are denoted by the bold face letters with bar, such as "$\overline{\mathbf{x}}, \widehat{\overline{\mathbf{x}}}$." The superscripts "^{T}" and "$^{-1}$" mean the transposed matrix and inverse matrix, respectively. $\theta(k) \in l_2[1, N)$ is the Euclidean norm; that is, $\sum_{k=1}^{N} \theta^T(k)\theta(k) < \infty$.

2. Problem Formulation

Consider the following discrete networked dynamical system, which is observed by N sensors

$$\begin{aligned} \mathbf{x}(k) &= \mathbf{F}(k, k-1)\,\mathbf{x}(k-1) + \mathbf{w}(k, k-1), \\ \mathbf{y}_l(k) &= \mathbf{H}_l(k)\,\mathbf{x}(k) + \mathbf{v}_l(k), \quad l = 1, 2, \ldots, N, \\ \mathbf{z}(k) &= \mathbf{L}(k)\,\mathbf{x}(k), \end{aligned} \tag{1}$$

where $\mathbf{x}(k) \in \mathbf{R}^n$ is the state vector. $\mathbf{y}_l(k) \in \mathbf{R}^q$ is the measurement output of sensor l. $\mathbf{z}(k)$ is the signal to be estimated. $\mathbf{F}(k, k-1)$, $\mathbf{H}_l(k)$, and $\mathbf{L}(k)$ are the system matrices with compatible dimensions. $\mathbf{w}(k, k-1) \in l_2[1, N)$ is the process noise and $\mathbf{v}_l(k) \in l_2[1, N)$ is the corresponding measurement noise of sensor l.

According to the traditional finite horizon H_∞ filter for single-sensor system, for a given scalar $\gamma > 0$, $\widehat{\mathbf{z}}(k \mid k)$ can be obtained as an approximation of $\mathbf{z}(k)$ based on the received measurements $\{\mathbf{y}(i) \mid i = 1, 2, \ldots k\}$ to guarantee the following H_∞ performance criterion function:

$$\begin{aligned} \sup_{\mathbf{w},\mathbf{v}\in l_2[1,N)} \Bigg(& \left(\sum_{i=1}^{k} \mathbf{e}^T(i)\,\mathbf{e}(i) \right) \\ & \times \left(\sum_{i=1}^{k} \mathbf{v}^T(i)\,\mathbf{v}(i) + \sum_{i=1}^{k} \mathbf{w}^T(i, i-1)\,\mathbf{w}(i, i-1) \right. \\ & \left. + \widetilde{\mathbf{x}}_0^T \mathbf{P}_0^{-1} \widetilde{\mathbf{x}}_0 \right)^{-1} \Bigg) < \gamma^2, \end{aligned} \tag{2}$$

where $\mathbf{e}_z(k) = \mathbf{z}(k) - \widehat{\mathbf{z}}(k \mid k)$, $\widetilde{\mathbf{x}}_0 = \mathbf{x}(0) - \widehat{\mathbf{x}}_0$, and $\widehat{\mathbf{x}}_0$ is an initial estimate of $\mathbf{x}(0)$. $\mathbf{P}_0$ is a given positive definite matrix with compatible dimension.

Remark 1. In the above performance criterion function, the filtering time of the fusion filter is the same as the sampling time of measurements, which, however, is not this case in networked dynamical systems.

In a fusion period, the networked measurement could arrive at the fusion filter in four different scenes considered in this paper; namely, the measurement could be ITM, ISDM, or OOSM. The unified filter for networked dynamical systems would deal with various measurement arrival scenes. The filtering time of networked measurement is its arrival time, rather than its sampling time. This means that the performance criterion function shown in (2) cannot be directly extended to the unified filtering process of networked dynamical systems. In the next section, a novel performance criterion function is built for various measurement arrival scenes in the networked dynamical system, firstly. On this

basis, a unified finite horizon H_∞ fusion filtering method is deduced.

3. Unified Finite Horizon H_∞ Fusion Filtering for Networked Dynamical System

3.1. Performance Criterion Function. Let $\kappa(i)$ be a counter, which counts for the number of the measurements received by the fusion filter in the fusion period $[i, i+1)$. At the start of the period, $\kappa(i) = 0$. Whenever a measurement arrives at the fusion filter, $\kappa(i) = \kappa(i) + 1$. Denote the jth measurement received by the fusion filter in the period $[i, i+1)$ as $\mathbf{y}_{\alpha_j^i}(\beta_j^i)$, in which the notations j, α_j^i, β_j^i mean that $\mathbf{y}_{\alpha_j^i}(\beta_j^i)$ is sampled by sensor α_j^i at the sampled time β_j^i. Here $\beta_j^i \le i$, $\alpha_j^i \le N$, and $j \le iN$ are all positive integers.

According to the measurement arrival scenes, for a given $\gamma > 0$, a novel finite horizon H_∞ fusion filtering performance criterion function could be given as follows in the fusion period $[k, k+1)$ to obtain $\widehat{\mathbf{z}}(k \mid k)$ based on the received measurement space $\{\mathbf{y}_{\alpha_j^i}(\beta_j^i) \mid 1 \le i \le k \text{ and } \kappa(i) \ne 0, j = 1, \ldots, \kappa(i)\}$. Consider

$$\sup_{\mathbf{w},\mathbf{v}\in l_2[1,N)} \left(\left(\sum_{\substack{i=1 \\ \kappa(i)\ne 0}}^{k} \sum_{j=1}^{\kappa(k)} \mathbf{e}_{z,j}^T(i)\, \mathbf{e}_{z,j}(i) \right) \times \left(\sum_{\substack{i=1 \\ \kappa(i)\ne 0}}^{k} \sum_{j=1}^{\kappa(k)} \mathbf{v}_{\alpha_j^i}^T\left(\beta_j^i\right) \mathbf{v}_{\alpha_j^i}\left(\beta_j^i\right) + \sum_{i=1}^{k} \mathbf{w}^T(i, i-1)\, \mathbf{w}(i, i-1) + \widetilde{\mathbf{x}}_0^T \mathbf{P}_0^{-1} \widetilde{\mathbf{x}}_0 \right)^{-1} \right) < \gamma^2, \tag{3}$$

where $\mathbf{e}_{z,j}(i) = \widehat{\mathbf{z}}_j(i \mid i) - \mathbf{z}(i)$. $\widehat{\mathbf{z}}_j(i \mid i)$ is the jth estimate of $\mathbf{z}(i)$, updated with the measurement $\mathbf{y}_{\alpha_j^i}(\beta_j^i)$.

Remark 2. The performance criterion function shown in (3) can be utilized for the finite horizon H_∞ fusion filtering processes of the dynamical system with measurements which could arrive at fusion filter in the aforementioned four kinds of arrival scenes.

(1) When none measurement arrives at the fusion filter in the period $[k, k+1)$, $\kappa(k) = 0$.

(2) When a measurement firstly arrives at the fusion filter in $[k, k+1)$, $\kappa(k) = 1$. If $\beta_1^k = k$, the measurement is an ITM $\mathbf{y}_{\alpha_1^k}(k)$. If $\beta_1^k < k$, this measurement is a delay measurement (an ISDM or a OOSM) sampled at β_1^k by sensor α_1^k.

(3) When a second (third,..., etc.) measurement arrives in $[k, k+1)$, then let $\kappa(k) = \kappa(k) + 1$. The new measurement would also be an ITM or a delay measurement.

3.2. Unified Finite Horizon H_∞ Filter Design. The performance criterion function shown in (3) can also be described as the following indefinite quadratic inequality:

$$J_{\kappa(k)} = \sum_{\substack{i=1 \\ \kappa(i)\ne 0}}^{k} \sum_{j=1}^{\kappa(k)} \left(\mathbf{v}_{\alpha_j^i}^T\left(\beta_j^i\right) \mathbf{v}_{\alpha_j^i}\left(\beta_j^i\right) - \gamma^{-2} \mathbf{e}_{z,j}^T(i)\, \mathbf{e}_{z,j}(i) \right) + \sum_{i=1}^{k} \mathbf{w}^T(i, i-1)\, \mathbf{w}(i, i-1) + \widetilde{\mathbf{x}}_0^T \mathbf{P}_0^{-1} \widetilde{\mathbf{x}}_0 > 0. \tag{4}$$

Remark 3. The quadratic form $J_{\kappa(k)}$ satisfies the indefinite quadratic inequality above if and only if (1) $J_{\kappa(k)}$ has a stationary point, (2) the value of $J_{\kappa(k)}$ at the stationary point is a minimum, and (3) the minimum is positive.

The stationary point of an indefinite quadratic form in Hilbert space corresponds to a projection in Krein space which is solved to obtain the stationary point of $J_{\kappa(k)}$ in this paper. A Krein space state-space model associated with the system shown in (1) is introduced as

$$\begin{aligned} \overline{\mathbf{x}}(i) &= \mathbf{F}(i, i-1)\, \overline{\mathbf{x}}(i-1) + \overline{\mathbf{w}}(i, i-1), \quad i = 1, \ldots, k, \\ \overline{\mathbf{y}}_{\alpha_j^i}\left(\beta_j^i\right) &= \mathbf{H}_{\alpha_j^i}\left(\beta_j^i\right) \overline{\mathbf{x}}\left(\beta_j^i\right) + \overline{\mathbf{v}}_{y,\alpha_j^i}\left(\beta_j^i\right), \\ & 1 \le i \le k, \quad \kappa(i) \ne 0, \quad 0 < j \le \kappa(i), \\ \breve{\overline{\mathbf{z}}}(i) &= \mathbf{L}(i)\, \overline{\mathbf{x}}(i) + \overline{\mathbf{v}}_{z,j}(i), \\ & 1 \le i \le k, \quad \kappa(i) \ne 0, \quad 0 < j \le \kappa(i), \end{aligned} \tag{5}$$

with

$$\left\langle \begin{bmatrix} \overline{\mathbf{x}}(0) - \widehat{\overline{\mathbf{x}}}_0 \\ \overline{\mathbf{w}}(j_1, j_1 - 1) \\ \overline{\mathbf{v}}_{y,j_3}\left(\beta_{j_3}^{l_1}\right) \\ \overline{\mathbf{v}}_{z,j_3}(l_1) \end{bmatrix}, \begin{bmatrix} \overline{\mathbf{x}}(0) - \widehat{\overline{\mathbf{x}}}_0 \\ \overline{\mathbf{w}}(j_2, j_2 - 1) \\ \overline{\mathbf{v}}_{y,j_4}\left(\beta_{j_4}^{l_2}\right) \\ \overline{\mathbf{v}}_{z,j_4}(l_2) \end{bmatrix} \right\rangle = \begin{bmatrix} \mathbf{P}_0 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}\delta_{j_1,j_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}\delta_{j_3,j_4}\delta_{l_1,l_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\gamma^2\mathbf{I}\delta_{j_3,j_4}\delta_{l_1,l_2} \end{bmatrix}, \tag{6}$$

$$1 \le j_1, \quad j_2 \le k, \qquad 1 \le j_3, \quad j_4 \le k, \quad \kappa(j_3) > 1, \qquad 0 < l_1 \le \kappa(j_3), \quad 0 < l_2 \le \kappa(j_4).$$

Denote $\overline{\mathbf{W}}(k) := [\overline{\mathbf{w}}^T(1, 0), \ldots, \overline{\mathbf{w}}^T(k, k-1)]^T$, $\overline{\xi}(k) := [\overline{\mathbf{x}}^T(0), \overline{\mathbf{W}}^T(k)]^T$. The stationary point of the indefinite quadratic form shown in (4) corresponds to the projection of $\overline{\xi}(k)$ into the Krein subspace $\Gamma_{\kappa(k)}(k)$ spanned by $\{\overline{\mathbf{y}}_{\alpha_j^i}(\beta_j^i), \breve{\overline{\mathbf{z}}}_j(i) \mid 1 \le i \le k \text{ and } \kappa(i) \ne 0, j = 1, \ldots, \kappa(i)\}$.

According to the Krein space state equation shown in (5), we have $\overline{\mathbf{x}}(i-1) = \mathbf{F}(i-1,i)(\overline{\mathbf{x}}(i) - \overline{\mathbf{w}}(i,i-1))$, and thus

$$\begin{aligned}\overline{\mathbf{y}}_{\alpha_j^i}\left(\beta_j^i\right) &= \mathbf{H}_{\alpha_j^i}\left(\beta_j^i\right)\overline{\mathbf{x}}\left(\beta_j^i\right) + \overline{\mathbf{v}}_{y,\alpha_j^i}\left(\beta_j^i\right)\\ &= \mathbf{H}_{\alpha_j^i}\left(\beta_j^i\right)\mathbf{F}\left(\beta_j^i,i\right)\overline{\mathbf{x}}(i) + \overline{\mathbf{v}}_{y,\alpha_j^i}\left(\beta_j^i\right)\\ &\quad - \mathbf{H}_{\alpha_j^i}\left(\beta_j^i\right)\mathbf{F}\left(\beta_j^i,i\right)\overline{\mathbf{w}}\left(i,\beta_j^i\right).\end{aligned} \tag{7}$$

Let $\overline{\mathbf{y}}_j^*(i) = \overline{\mathbf{y}}_{\alpha_j^i}(\beta_j^i)$; $\mathbf{H}_j^*(i) = \mathbf{H}_{\alpha_j^i}(\beta_j^i)\mathbf{F}(\beta_j^i,i)$; $\overline{\mathbf{v}}_{y,j}^*(i) = \overline{\mathbf{v}}_{y,\alpha_j^i}(\beta_j^i) - \mathbf{H}_{\alpha_j^i}(\beta_j^i)\mathbf{F}(\beta_j^i,i)\overline{\mathbf{w}}(i,\beta_j^i)$; we have

$$\overline{\mathbf{y}}_j^*(i) = \mathbf{H}_j^*(i)\,\overline{\mathbf{x}}(i) + \overline{\mathbf{v}}_{y,j}^*(i), \tag{8}$$

with the Gramian matrixes

$$\begin{aligned}&\left\langle \overline{\mathbf{w}}(l,l-1), \overline{\mathbf{v}}_{y,j}^*(i)\right\rangle\\ &\quad = \mathbf{Q}(l,l-1)\,\mathbf{F}^T(i,l+1)\left(\mathbf{H}_j^*(i)\right)^T,\\ &\qquad\qquad \beta_j^i \le l \le i-1,\\ &\mathbf{P}_{vyy,j}^*(i) = \left\langle \overline{\mathbf{v}}_{y,j}^*(i), \overline{\mathbf{v}}_{y,j}^*(i)\right\rangle\\ &\quad = \mathbf{I} + \mathbf{H}_j^*(i)\left(\sum_{l=\beta_j^i+1}^{i}\mathbf{F}(i,l)\,\mathbf{F}^T(i,l)\right)\left(\mathbf{H}_j^*(i)\right)^T,\\ &\qquad\qquad \beta_j^i < i,\end{aligned} \tag{9}$$

where $\mathbf{F}(i,i) = \mathbf{I}$.

Denote the Krein subspace spanned by $\{\overline{\mathbf{y}}_l(\beta_l^\tau), \breve{\mathbf{z}}_l(\tau) \mid 1 \le \tau \le i-1 \text{ and } \kappa(\tau) \ne 0, l = 1,\ldots,\kappa(\tau)\}$ and $\{\overline{\mathbf{y}}_l(\beta_l^i), \breve{\mathbf{z}}_l(i) \mid l = 1,\ldots,j-1\}$ by $\Gamma_{j-1}(i)$. The projection of $\overline{\mathbf{y}}_j^*(i)$ into $\Gamma_{j-1}(i)$ is given by

$$\widehat{\overline{\mathbf{y}}}_j^*(i \mid i-1) = \mathbf{H}_j^*(i)\,\widehat{\overline{\mathbf{x}}}_j(i \mid i-1) + \widehat{\overline{\mathbf{v}}}_{y,j}^*(i \mid i-1), \tag{10}$$

in which

$$\begin{aligned}\widehat{\overline{\mathbf{x}}}_j(i \mid i-1) &= \widehat{\overline{\mathbf{x}}}_{j-1}(i \mid i), \quad j > 1,\\ \widehat{\overline{\mathbf{x}}}_1(i \mid i-1) &= \mathbf{F}(i,i-1)\,\widehat{\overline{\mathbf{x}}}_{\kappa(i-1)}(i-1 \mid i-1),\end{aligned} \tag{11}$$

$$\widehat{\overline{\mathbf{v}}}_{y,j}^*(i \mid i-1) = -\mathbf{H}_j^*(i)\,\widehat{\overline{\mathbf{w}}}\left(i,\beta_j^i \mid i-1\right), \tag{12}$$

$$\widehat{\breve{\mathbf{z}}}_j(i \mid i-1) = \mathbf{L}(i)\,\widehat{\overline{\mathbf{x}}}_j(i \mid i-1). \tag{13}$$

Let $\overline{\mathbf{e}}_{yz,j}^*(i) := \begin{bmatrix}\overline{\mathbf{y}}_j^*(i) - \widehat{\overline{\mathbf{y}}}_j^*(i|i-1)\\ \breve{\mathbf{z}}_j(i) - \widehat{\breve{\mathbf{z}}}_j(i|i-1)\end{bmatrix} = \begin{bmatrix}\mathbf{H}_j^*(i)\\ \mathbf{L}(i)\end{bmatrix}\widetilde{\overline{\mathbf{x}}}_j(i \mid i-1) + \begin{bmatrix}\overline{\mathbf{v}}_{y,j}^*(i|i-1)\\ \overline{\mathbf{v}}_{z,j}(i)\end{bmatrix}$, $\mathbf{R}_{wyz,l}(i,\beta_j^i,k) := \langle\overline{\mathbf{w}}(i,\beta_j^i), \overline{\mathbf{e}}_{yz,l}^*(k)\rangle$, $\mathbf{R}_{eyz,j}^*(i) := \langle\overline{\mathbf{e}}_{yz,j}^*(i), \overline{\mathbf{e}}_{yz,j}^*(i)\rangle$. It is obvious that $\overline{\mathbf{e}}_{yz,j-1}^*(i \mid i-1) \perp \Gamma_{j-1}(i)$, and $\{\overline{\mathbf{e}}_{yz,j}^*(i \mid i-1) \mid 1 \le i \le k \text{ and } \kappa(i) \ne 0, j = 1,\ldots,\kappa(i)\}$ is an orthogonal basis of $\Gamma_{\kappa(k)}(k)$. The projection of $\overline{\xi}(k)$ into $\Gamma_{\kappa(k)}(k)$ is given by

$$\widehat{\overline{\xi}}_{\kappa(k)}(k \mid k) = \sum_{\substack{i=1,\\ \kappa(i)\ne 0}}^{k}\sum_{j=1}^{\kappa(i)}\left\langle\overline{\xi}(k), \overline{\mathbf{e}}_{yz,j}^*(i)\right\rangle\left(\mathbf{R}_{eyz,j}^*(i)\right)^{-1}\overline{\mathbf{e}}_{yz,j}^*(i). \tag{14}$$

The projection of $\overline{\mathbf{x}}(k)$ into $\Gamma_{\kappa(k)}(k)$ is

$$\begin{aligned}\widehat{\overline{\mathbf{x}}}_{\kappa(k)}(k \mid k) &= \sum_{\substack{i=1,\\ \kappa(i)\ne 0}}^{k}\sum_{j=1}^{\kappa(i)}\left\langle\overline{\mathbf{x}}(k), \overline{\mathbf{e}}_{yz,j}^*(i)\right\rangle\left(\mathbf{R}_{eyz,j}^*(i)\right)^{-1}\overline{\mathbf{e}}_{yz,j}^*(i)\\ &= \sum_{\substack{i=1,\\ \kappa(i)\ne 0}}^{k}\sum_{j=1}^{\kappa(i)}\left\langle\overline{\mathbf{x}}(k), \overline{\mathbf{e}}_{yz,j}^*(i)\right\rangle\left(\mathbf{R}_{eyz,j}^*(i)\right)^{-1}\overline{\mathbf{e}}_{yz,j}^*(i)\\ &= \widehat{\overline{\mathbf{x}}}_{\kappa(k)-1}(k \mid k) + \left\langle\overline{\mathbf{x}}(k), \overline{\mathbf{e}}_{yz,\kappa(k)}^*(k)\right\rangle\\ &\quad\times\left(\mathbf{R}_{eyz,\kappa(k)}^*(k)\right)^{-1}\overline{\mathbf{e}}_{yz,\kappa(k)}^*(k)\\ &= \widehat{\overline{\mathbf{x}}}_{\kappa(k)}(k \mid k-1) + \left\langle\overline{\mathbf{x}}(k), \overline{\mathbf{e}}_{yz,\kappa(k)}^*(k)\right\rangle\\ &\quad\times\left(\mathbf{R}_{eyz,\kappa(k)}^*(k)\right)^{-1}\overline{\mathbf{e}}_{yz,\kappa(k)}^*(k)\\ \widehat{\overline{\mathbf{x}}}_0(k \mid k) &= \mathbf{F}(k,k-1)\,\widehat{\overline{\mathbf{x}}}_{\kappa(k-1)}(k-1 \mid k-1),\end{aligned} \tag{15}$$

and the noise projection $\widehat{\overline{\mathbf{w}}}(i,\beta_j^i \mid i-1)$ in (12) is given by

$$\begin{aligned}\widehat{\overline{\mathbf{w}}}_j\left(i,\beta_j^i \mid i-1\right) &= \widehat{\overline{\mathbf{w}}}_{j-1}\left(i,\beta_j^i, i-1\right)\\ &\quad + \mathbf{R}_{wyz,j-1}\left(i,\beta_j^i,i\right)\left(\mathbf{R}_{eyz,j-1}^*(i)\right)^{-1}\\ &\quad\times\overline{\mathbf{e}}_{yz,j-1}^*(i),\\ \widehat{\overline{\mathbf{w}}}_1\left(i,\beta_j^i \mid i-1\right) &= \widehat{\overline{\mathbf{w}}}_{\kappa(i-1)}\left(i,\beta_j^i \mid i-2\right)\\ &\quad + \mathbf{R}_{wyz,\kappa(i-1)}\left(i,\beta_j^i,i-1\right)\\ &\quad\times\left(\mathbf{R}_{eyz,\kappa(i-1)}^*(i-1)\right)^{-1}\\ &\quad\times\overline{\mathbf{e}}_{yz,\kappa(i-1)}^*(i-1),\\ \widehat{\overline{\mathbf{w}}}_{\kappa(i-1)}\left(i,\beta_j^i \mid i-2\right) &= \mathbf{F}(i,i-1)\,\widehat{\overline{\mathbf{w}}}_{\kappa(i-1)}\left(i-1,\beta_j^i \mid i-2\right),\end{aligned} \tag{16}$$

where

$$\begin{aligned}\mathbf{K}_{\kappa(k)}(k) &:= \left\langle \overline{\mathbf{x}}(k), \overline{\mathbf{e}}^{*}_{yz,\kappa(k)}(k)\right\rangle \left(\mathbf{R}^{*}_{eyz,\kappa(k)}(k)\right)^{-1} \\ &:= \mathbf{R}_{xyz,\kappa(k)}(k)\left(\mathbf{R}^{*}_{eyz,\kappa(k)}(k)\right)^{-1},\end{aligned} \tag{17}$$

$$\begin{aligned}&\mathbf{R}_{xyz,\kappa(k)}(k) \\ &= \left\langle \overline{\mathbf{x}}(k), \overline{\mathbf{e}}^{*}_{yz,\kappa(k)}(k)\right\rangle \\ &= \left\langle \overline{\mathbf{x}}(k), \widetilde{\overline{\mathbf{x}}}_{\kappa(k)}(k \mid k-1)\right\rangle \left[\left(\mathbf{H}^{*}_{\kappa(k)}(k)\right)^{T} \ \mathbf{L}^{T}(k)\right] \\ &\quad - \left[\left\langle \overline{\mathbf{x}}(k), \widetilde{\overline{\mathbf{w}}}_{\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\right\rangle \left(\mathbf{H}^{*}_{\kappa(k)}(k)\right)^{T} \ \mathbf{0}\right] \\ &:= \mathbf{P}_{\kappa(k)}(k)\left[\left(\mathbf{H}^{*}_{\kappa(k)}(k)\right)^{T} \ \mathbf{L}^{T}(k)\right] \\ &\quad - \left[\mathbf{R}_{xw,\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\left(\mathbf{H}^{*}_{\kappa(k)}(k)\right)^{T} \ \mathbf{0}\right] \\ &:= \left[\mathbf{R}_{xy,\kappa(k)}(k) \ \ \mathbf{R}_{xz,\kappa(k)}(k)\right],\end{aligned}$$

$$\begin{aligned}&\mathbf{R}^{*}_{eyz,\kappa(k)}(k) \\ &:= \begin{bmatrix}\mathbf{H}^{*}_{\kappa(k)}(k) \\ \mathbf{L}(k)\end{bmatrix}\mathbf{P}_{\kappa(k)}(k)\left[\left(\mathbf{H}^{*}_{\kappa(k)}(k)\right)^{T} \ \mathbf{L}^{T}(k)\right] + \mathbf{R}^{*}_{vyz,\kappa(k)}(k) \\ &\quad - \begin{bmatrix}\mathbf{H}^{*}_{\kappa(k)}(k)\mathbf{R}^{T}_{xw,\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\left(\mathbf{H}^{*}_{\kappa(k)}(k)\right)^{T} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix} \\ &\quad - \begin{bmatrix}\mathbf{H}^{*}_{\kappa(k)}(k)\mathbf{R}^{T}_{xw,\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\left(\mathbf{H}^{*}_{\kappa(k)}(k)\right)^{T} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}^{T},\end{aligned}$$

$$\mathbf{R}^{*}_{vyz,\kappa(k)}(k) = \begin{bmatrix}\mathbf{R}^{*}_{vyy,\kappa(k)}(k) & \mathbf{0} \\ \mathbf{0} & -\gamma^{2}\mathbf{I}\end{bmatrix}$$

$$\begin{aligned}&\mathbf{R}_{xw,\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right) \\ &= \left\langle \overline{\mathbf{x}}(k), \widetilde{\overline{\mathbf{w}}}_{\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\right\rangle \\ &= \left\langle \overline{\mathbf{x}}(k), \overline{\mathbf{w}}\left(k, \beta^{k}_{\kappa(k)}\right)\right\rangle \\ &\quad - \left\langle \overline{\mathbf{x}}(k), \widehat{\overline{\mathbf{w}}}_{\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\right\rangle \\ &:= \mathbf{Q}\left(k, \beta^{k}_{\kappa(k)}\right) \\ &\quad - \mathbf{P}_{xw,\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right),\end{aligned}$$

$$\begin{aligned}&\mathbf{R}^{*}_{vyy,\kappa(k)}(k) \\ &= \left\langle \widetilde{\overline{\mathbf{v}}}^{*}_{y,\kappa(k)-1}(k \mid k-1), \widetilde{\overline{\mathbf{v}}}^{*}_{y,\kappa(k)-1}(k \mid k-1)\right\rangle \\ &= \left\langle \overline{\mathbf{v}}^{*}_{y,\kappa(k)-1}(k \mid k-1), \overline{\mathbf{v}}^{*}_{y,\kappa(k)-1}(k \mid k-1)\right\rangle \\ &\quad - \left\langle \widetilde{\overline{\mathbf{v}}}^{*}_{y,\kappa(k)-1}(k \mid k-1), \widehat{\overline{\mathbf{v}}}^{*}_{y,\kappa(k)-1}(k \mid k-1)\right\rangle \\ &= \mathbf{P}^{*}_{vyy,\kappa(k)}(k) - \mathbf{H}^{*}_{\kappa(k)}(k)\mathbf{R}^{T}_{ww,\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right) \\ &\quad \times \left(\mathbf{H}^{*}_{\kappa(k)}(k)\right)^{T},\end{aligned}$$

$$\begin{aligned}&\mathbf{P}_{xw,\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right) \\ &= \left\langle \overline{\mathbf{x}}(k), \widehat{\overline{\mathbf{w}}}_{\kappa(k)}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\right\rangle \\ &= \left\langle \overline{\mathbf{x}}(k), \widehat{\overline{\mathbf{w}}}_{\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\right\rangle \\ &\quad + \left\langle \overline{\mathbf{x}}(k), \overline{\mathbf{e}}^{*}_{yz,\kappa(k)-1}(k)\right\rangle \left(\mathbf{R}^{*}_{eyz,\kappa(k)-1}(k)\right)^{-1} \\ &\quad \times \mathbf{R}^{T}_{wyz,\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)}, k\right) \\ &= \mathbf{P}_{xw,\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right) + \mathbf{R}_{xyz,\kappa(k)-1}(k) \\ &\quad \times \left(\mathbf{R}^{*}_{eyz,\kappa(k)-1}(k)\right)^{-1}\mathbf{R}^{T}_{wyz,\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)}, k\right), \quad \kappa(k) > 1,\end{aligned}$$

$$\begin{aligned}&\mathbf{P}_{xw,1}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right) \\ &= \left\langle \overline{\mathbf{x}}(k), \widehat{\overline{\mathbf{w}}}_{1}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\right\rangle \\ &= \mathbf{F}(k, k-1)\left\langle \overline{\mathbf{x}}(k-1), \widehat{\overline{\mathbf{w}}}_{\kappa(k-1)}\left(k-1, \beta^{k}_{\kappa(k)} \mid k-2\right)\right\rangle \\ &\quad \times \mathbf{F}^{T}(k, k-1) + \left\langle \overline{\mathbf{x}}(k), \overline{\mathbf{e}}^{*}_{yz,\kappa(k-1)}(k-1)\right\rangle \\ &\quad \times \left(\mathbf{R}^{*}_{eyz,\kappa(k-1)}(k-1)\right)^{-1}\mathbf{R}^{T}_{wyz,\kappa(k-1)}\left(k, \beta^{k}_{\kappa(k)}, k-1\right) \\ &= \mathbf{F}(k, k-1)\mathbf{P}_{xw,\kappa(k-1)}\left(k-1, \beta^{k}_{\kappa(k)} \mid k-2\right)\mathbf{F}^{T}(k, k-1) \\ &\quad + \mathbf{R}_{xyz,\kappa(k-1)}(k-1)\left(\mathbf{R}^{*}_{eyz,\kappa(k-1)}(k-1)\right)^{-1} \\ &\quad \times \mathbf{R}^{T}_{wyz,\kappa(k-1)}\left(k, \beta^{k}_{\kappa(k)}, k-1\right),\end{aligned}$$

$$\begin{aligned}&\mathbf{R}_{wyz,\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)}, k\right) \\ &= \left\langle \overline{\mathbf{w}}\left(k, \beta^{k}_{\kappa(k)}\right), \overline{\mathbf{e}}^{*}_{yz,\kappa(k)-1}(k)\right\rangle \\ &= \left\langle \overline{\mathbf{w}}\left(k, \beta^{k}_{\kappa(k)}\right), \begin{bmatrix}\mathbf{H}^{*}_{\kappa(k)-1}(k) \\ \mathbf{L}(k)\end{bmatrix}\widetilde{\overline{\mathbf{x}}}_{\kappa(k)-1}(k \mid k-1) \right. \\ &\qquad \left. + \begin{bmatrix}\widetilde{\overline{\mathbf{v}}}^{*}_{y,\kappa(k)-1}(k \mid k-1) \\ \overline{\mathbf{v}}_{z,\kappa(k)-1}(k)\end{bmatrix}\right\rangle \\ &= \mathbf{R}^{T}_{xw,\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\begin{bmatrix}\mathbf{H}^{*}_{\kappa(k)-1}(k) \\ \mathbf{L}(k)\end{bmatrix}^{T} \\ &\quad - \begin{bmatrix}\mathbf{R}^{T}_{ww,\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)}k-1\right)\left(\mathbf{H}^{*}_{\kappa(k)-1}(k)\right)^{T} \\ \mathbf{0}\end{bmatrix},\end{aligned}$$

$$\begin{aligned}&\mathbf{R}_{ww,\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right) \\ &:= \left\langle \overline{\mathbf{w}}\left(k, \beta^{k}_{\kappa(k)}\right), \widetilde{\overline{\mathbf{w}}}_{\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\right\rangle \\ &= \left\langle \overline{\mathbf{w}}\left(k, \beta^{k}_{\kappa(k)}\right), \overline{\mathbf{w}}\left(k, \beta^{k}_{\kappa(k)}\right)\right\rangle \\ &\quad - \left\langle \overline{\mathbf{w}}\left(k, \beta^{k}_{\kappa(k)}\right), \widehat{\overline{\mathbf{w}}}_{\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right)\right\rangle \\ &= \mathbf{Q}\left(k, \beta^{k}_{\kappa(k)}\right) - \mathbf{P}_{ww,\kappa(k)-1}\left(k, \beta^{k}_{\kappa(k)} \mid k-1\right),\end{aligned}$$

$$
\begin{aligned}
&\mathbf{P}_{ww,\kappa(k)}\left(k,\beta^k_{\kappa(k)}\mid k-1\right)\\
&:=\left\langle\overline{\mathbf{w}}\left(k,\beta^k_{\kappa(k)}\right),\widehat{\overline{\mathbf{w}}}_{\kappa(k)}\left(k,\beta^k_{\kappa(k)}\mid k-1\right)\right\rangle\\
&=\left\langle\overline{\mathbf{w}}\left(k,\beta^k_{\kappa(k)}\right),\widehat{\overline{\mathbf{w}}}_{\kappa(k)-1}\left(k,\beta^k_{\kappa(k)}\mid k-1\right)\right\rangle\\
&\quad+\left\langle\overline{\mathbf{w}}\left(k,\beta^k_{\kappa(k)}\right),\overline{\mathbf{e}}^*_{yz,\kappa(k)-1}(k)\right\rangle\\
&\quad\times\left(\mathbf{R}^*_{eyz,\kappa(k)-1}(k)\right)^{-1}\mathbf{R}^T_{wyz,\kappa(k)-1}\left(k,\beta^k_{\kappa(k)},k\right)\\
&=\mathbf{P}_{ww,\kappa(k)-1}\left(k,\beta^k_{\kappa(k)}\mid k-1\right)+\mathbf{R}_{wyz,\kappa(k)-1}(k)\\
&\quad\times\left(\mathbf{R}^*_{eyz,\kappa(k)-1}(k)\right)^{-1}\mathbf{R}^T_{wyz,\kappa(k)-1}\left(k,\beta^k_{\kappa(k)},k\right),\quad \kappa(k)>1,\\
&\mathbf{P}_{ww,1}\left(k,\beta^k_{\kappa(k)}\mid k-1\right)\\
&=\left\langle\overline{\mathbf{w}}\left(k,\beta^k_{\kappa(k)}\right),\widehat{\overline{\mathbf{w}}}_1\left(k,\beta^k_{\kappa(k)}\mid k-1\right)\right\rangle\\
&=\left\langle\overline{\mathbf{w}}\left(k,\beta^k_{\kappa(k)}\right),\widehat{\overline{\mathbf{w}}}_{\kappa(k-1)}\left(k,\beta^k_{\kappa(k)}\mid k-2\right)\right\rangle\\
&\quad+\left\langle\overline{\mathbf{w}}\left(k,\beta^k_{\kappa(k)}\right),\overline{\mathbf{e}}^*_{yz,\kappa(k-1)}(k-1)\right\rangle\\
&\quad\times\left(\mathbf{R}^*_{eyz,\kappa(k-1)}(k-1)\right)^{-1}\mathbf{R}^T_{wyz,\kappa(k-1)}\left(k,\beta^k_{\kappa(k)},k-1\right)\\
&=\mathbf{F}(k,k-1)\,\mathbf{P}_{ww,\kappa(k-1)}\left(k-1,\beta^k_{\kappa(k)}\mid k-2\right)\mathbf{F}^T(k,k-1)\\
&\quad+\mathbf{R}_{wyz,\kappa(k-1)}\left(k,\beta^k_{\kappa(k)},k-1\right)\left(\mathbf{R}^*_{eyz,\kappa(k-1)}(k-1)\right)^{-1}\\
&\quad\times\mathbf{R}^T_{wyz,\kappa(k-1)}\left(k,\beta^k_{\kappa(k)},k-1\right),
\end{aligned}
\tag{18}
$$

and

$$
\mathbf{Q}\left(k,\beta^k_{\kappa(k)}\right)=\begin{cases}\mathbf{0}, & \beta^k_{\kappa(k)}=k;\\ \displaystyle\sum_{i=\beta^k_{\kappa(k)}+1}^{k}\mathbf{F}(k,i)\,\mathbf{F}^T(k,i), & \beta^k_{\kappa(k)}<k.\end{cases}
\tag{19}
$$

The projection of $\overline{\xi}(k)$ in (14) corresponds to a stationary point of the indefinite quadratic form $J_{\kappa(k)}$ in (4), and the value of $J_{\kappa(k)}$ at this stationary point is

$$
\begin{aligned}
\mathbf{J}^*_{\kappa(k)}(k)&=\sum_{\substack{i=1,\\ \kappa(i)\neq 0}}^{k}\sum_{j=1}^{\kappa(i)}\left(\mathbf{e}^*_{yz,j}(i)\right)^T\left(\mathbf{R}^*_{eyz,j}(i)\right)^{-1}\mathbf{e}^*_{yz,j}(i)\\
&=\mathbf{J}^*_{\kappa(k)-1}(k)+\left(\mathbf{e}^*_{yz,\kappa(k)}(k)\right)^T\left(\mathbf{R}^*_{eyz,\kappa(k)}(k)\right)^{-1}\\
&\quad\times\mathbf{e}^*_{yz,\kappa(k)}(k),
\end{aligned}
\tag{20}
$$

where $\mathbf{J}^*_0(k)=\mathbf{J}^*_{\kappa(k-1)}(k-1)$,

$$
\begin{aligned}
\mathbf{e}^*_{yz,\kappa(k)}(k)&=\begin{bmatrix}\mathbf{e}^*_{y,\kappa(k)}(k\mid k-1)\\ \mathbf{e}_{z,\kappa(k)}(k\mid k-1)\end{bmatrix}\\
&=\begin{bmatrix}\mathbf{y}_{\alpha^k_{\kappa(k)}}\left(\beta^k_{\kappa(k)}\right)-\widehat{\mathbf{y}}^*_{\kappa(k)}(k\mid k-1)\\ \widehat{\mathbf{z}}_{\kappa(k)}(k\mid k)-\widehat{\mathbf{z}}_{\kappa(k)}(k\mid k-1)\end{bmatrix},\\
\widehat{\mathbf{y}}^*_{\kappa(k)}(k\mid k-1)&=\mathbf{H}^*_{\kappa(k)}(k)\,\widehat{\mathbf{x}}_{\kappa(k)}(k\mid k-1)\\
&\quad-\mathbf{H}^*_{\kappa(k)}(k)\,\widehat{\mathbf{w}}_{\kappa(k)}\left(k,\beta^k_{\kappa(k)}\mid k-1\right),\\
\widehat{\mathbf{z}}_{\kappa(k)}(k\mid k-1)&=\mathbf{L}(k)\,\widehat{\mathbf{x}}_{\kappa(k)}(k\mid k-1),\\
\widehat{\mathbf{w}}_j\left(i,\beta^i_j\mid i-1\right)&=\widehat{\mathbf{w}}_{j-1}\left(i,\beta^i_j\mid i-1\right)+\mathbf{R}_{wyz,j-1}\left(i,\beta^k_j,i\right)\\
&\quad\times\left(\mathbf{R}^*_{eyz,j-1}(i)\right)^{-1}\mathbf{e}^*_{yz,j-1}(i),\quad j>1,\\
\widehat{\mathbf{w}}_1\left(i,\beta^i_j\mid i-1\right)&=\widehat{\mathbf{w}}_{\kappa(i-1)}\left(i,\beta^i_j\mid i-2\right)\\
&\quad+\mathbf{R}_{wyz,\kappa(i-1)}\left(i,\beta^k_j,i-1\right)\\
&\quad\times\left(\mathbf{R}^*_{eyz,\kappa(i-1)}(i-1)\right)^{-1}\\
&\quad\times\mathbf{e}^*_{yz,\kappa(i-1)}(i-1),\\
\widehat{\mathbf{w}}_{\kappa(i-1)}\left(i,\beta^i_j\mid i-2\right)&=\mathbf{F}(i,i-1)\,\widehat{\mathbf{w}}_{\kappa(i-1)}\left(i-1,\beta^i_j\mid i-2\right),\\
\widehat{\mathbf{x}}_{\kappa(k)}(k\mid k-1)&=\widehat{\mathbf{x}}_{\kappa(k)-1}(k\mid k),\quad \kappa(k)>0,\\
\widehat{\mathbf{x}}_0(k\mid k-1)&=\mathbf{F}(k,k-1)\,\widehat{\mathbf{x}}_{\kappa(k-1)}(k-1\mid k-1).
\end{aligned}
\tag{21}
$$

In (20), $\mathbf{R}^*_{eyz,\kappa(k)}(k)$ can also be expressed as follows:

$$
\mathbf{R}^*_{eyz,\kappa(k)}(k):=\begin{bmatrix}\mathbf{R}_{yy,\kappa(k)}(k) & \mathbf{R}_{yz,\kappa(k)}(k)\\ \mathbf{R}_{zy,\kappa(k)}(k) & \mathbf{R}_{zz,\kappa(k)}(k)\end{bmatrix};
\tag{22}
$$

then,

$$
\begin{aligned}
&\left(\mathbf{R}^*_{eyz,\kappa(k)}(k)\right)^{-1}\\
&=\begin{bmatrix}\mathbf{R}_{yy,\kappa(k)}(k) & \mathbf{R}_{yz,\kappa(k)}(k)\\ \mathbf{R}_{zy,\kappa(k)}(k) & \mathbf{R}_{zz,\kappa(k)}(k)\end{bmatrix}^{-1}\\
&=\begin{bmatrix}\mathbf{I} & -\mathbf{R}^{-1}_{yy,\kappa(k)}(k)\,\mathbf{R}_{yz,\kappa(k)}(k)\\ \mathbf{0} & \mathbf{I}\end{bmatrix}\\
&\quad\times\begin{bmatrix}\mathbf{R}_{yy,\kappa(k)}(k) & \mathbf{0}\\ \mathbf{0} & \mathbf{R}_{zz,\kappa(k)}(k)-\mathbf{R}_{zy,\kappa(k)}(k)\mathbf{R}^{-1}_{yy,\kappa(k)}(k)\mathbf{R}_{yz,\kappa(k)}(k)\end{bmatrix}^{-1}\\
&\quad\times\begin{bmatrix}\mathbf{I} & \mathbf{0}\\ -\mathbf{R}_{zy,\kappa(k)}(k)\,\mathbf{R}^{-1}_{yy,\kappa(k)}(k) & \mathbf{I}\end{bmatrix}.
\end{aligned}
\tag{23}
$$

Letting $\hat{\mathbf{z}}^*_{\kappa(k)}(k \mid k) = \hat{\mathbf{z}}_{\kappa(k)}(k \mid k-1) - \mathbf{R}_{zy,\kappa(k)}(k)\mathbf{R}^{-1}_{yy,\kappa(k)}(k)\mathbf{e}^*_{y,\kappa(k)}(k \mid k-1)$, the last term in (20) is

$$\begin{aligned}
&\mathbf{e}^*_{yz,\kappa(k)}(k)\left(\mathbf{R}^*_{eyz,\kappa(k)}(k)\right)^{-1}\mathbf{e}^*_{yz,\kappa(k)}(k)\\
&= \left(\mathbf{e}^*_{y,\kappa(k)}(k \mid k-1)\right)^T \mathbf{R}^{-1}_{yy,\kappa(k)}(k)\,\mathbf{e}^*_{y,\kappa(k)}(k \mid k-1)\\
&\quad + \left(\hat{\mathbf{z}}_{\kappa(k)}(k \mid k) - \hat{\mathbf{z}}^*_{\kappa(k)}(k \mid k)\right)^T\\
&\quad \times \left(\mathbf{R}_{zz,\kappa(k)}(k) - \mathbf{R}_{zy,\kappa(k)}(k)\,\mathbf{R}^{-1}_{yy,\kappa(k)}(k)\,\mathbf{R}_{yz,\kappa(k)}(k)\right)^{-1}\\
&\quad \times \left(\hat{\mathbf{z}}_{\kappa(k)}(k \mid k) - \hat{\mathbf{z}}^*_{\kappa(k)}(k \mid k)\right).
\end{aligned} \tag{24}$$

According to Lemma 12 in [21], $\mathbf{J}^*_{\kappa(k)}(k)$ is the minimum of $\mathbf{J}_{\kappa(k)}(k)$ if and only if $\mathbf{R}^*_{eyz,\kappa(k)}(k)$ and $\mathbf{R}^*_{vyz,\kappa(k)}(k)$ have the same inertia. Considering the block triangular factorization of $\mathbf{R}^*_{eyz,\kappa(k)}(k)$ as shown in (23), the minimum condition can also be given by

$$\mathbf{R}_{zz,\kappa(k)}(k) - \mathbf{R}_{zy,\kappa(k)}(k)\,\mathbf{R}^{-1}_{yy,\kappa(k)}(k)\,\mathbf{R}_{yz,\kappa(k)}(k) < \mathbf{0}. \tag{25}$$

Therefore, a choice of $\hat{\mathbf{z}}_{\kappa(k)}(k \mid k)$ to guarantee $\mathbf{J}^*_{\kappa(k)}(k) > 0$ is $\hat{\mathbf{z}}_{\kappa(k)}(k \mid k) = \hat{\mathbf{z}}^*_{\kappa(k)}(k \mid k)$, and the minimum of $\mathbf{J}_{\kappa(k)}(k)$ is

$$\begin{aligned}
\mathbf{J}^*_{\kappa(k)}(k) &= \mathbf{J}^*_{\kappa(k)-1}(k) + \mathbf{e}^*_{y,\kappa(k)}(k \mid k-1)\\
&\quad \times \left(\mathbf{R}_{yy,\kappa(k)}(k)\right)^{-1}\mathbf{e}^*_{y,\kappa(k)}(k \mid k-1).
\end{aligned} \tag{26}$$

Then, the estimation of the signal to be estimated is

$$\hat{\mathbf{z}}(k \mid k) = \mathbf{L}(k)\,\hat{\mathbf{x}}_{\kappa(k)}(k \mid k) \tag{27}$$

in which

$$\hat{\mathbf{x}}_{\kappa(k)}(k \mid k) = \hat{\mathbf{x}}_{\kappa(k)}(k \mid k-1) + \mathbf{K}_{y,\kappa(k)}(k)\,\mathbf{e}^*_{y,\kappa(k)}(k \mid k-1), \tag{28}$$

$$\mathbf{K}_{y,\kappa(k)}(k) = \mathbf{R}_{xy,\kappa(k)}(k)\,\mathbf{R}^{-1}_{yy,\kappa(k)}(k). \tag{29}$$

The parameters $\mathbf{R}_{xy,\kappa(k)}(k)$ and $\mathbf{R}^{-1}_{yy,\kappa(k)}(k)$ in (29) can be obtained by iterating the equations in (18).

In summary, the unified finite horizon H_∞ fusion filtering algorithm is given by

$$\begin{aligned}
\hat{\mathbf{z}}(k \mid k) &= \mathbf{L}(k)\,\hat{\mathbf{x}}_{\kappa(k)}(k \mid k)\\
\hat{\mathbf{x}}_{\kappa(k)}(k \mid k) &= \hat{\mathbf{x}}_{\kappa(k)-1}(k \mid k) + \mathbf{K}_{y,\kappa(k)}(k)\,\mathbf{e}^*_{y,\kappa(k)}(k \mid k-1)\\
\hat{\mathbf{x}}_0(k \mid k) &= \mathbf{F}(k,k-1)\,\hat{\mathbf{x}}_{\kappa(k-1)}(k-1 \mid k-1),
\end{aligned} \tag{30}$$

where

$$\begin{aligned}
&\mathbf{K}_{y,\kappa(k)}(k) = \mathbf{R}_{xy,\kappa(k)}(k)\,\mathbf{R}^{-1}_{yy,\kappa(k)}(k)\\
&\mathbf{e}^*_{y,\kappa(k)}(k \mid k-1)\\
&\quad = \mathbf{y}_{\alpha^k_{\kappa(k)}}\left(\beta^k_{\kappa(k)}\right) - \hat{\mathbf{y}}^*_{\kappa(k)}(k \mid k-1)\\
&\quad = \mathbf{y}_{\alpha^k_{\kappa(k)}}\left(\beta^k_{\kappa(k)}\right) - \mathbf{H}^*_{\kappa(k)}(k)\,\hat{\mathbf{x}}_{\kappa(k)}(k \mid k-1)\\
&\qquad - \mathbf{H}^*_{\kappa(k)}(k)\,\widehat{\mathbf{w}}_{\kappa(k)}\left(k, \beta^k_{\kappa(k)} \mid k-1\right).
\end{aligned} \tag{31}$$

The Riccati equations are given as follows.

(1) If the next measurement arrives at fusion filter in the fusion period $[k, k+1)$, the Ricatti equation is

$$\mathbf{P}_{\kappa(k)+1}(k) = \mathbf{P}_{\kappa(k)}(k) - \mathbf{K}_{\kappa(k)}(k)\,\mathbf{R}^T_{xyz,\kappa(k)}(k). \tag{32}$$

(2) Otherwise, the next measurement arrives at fusion filter in the fusion period $[k+1, k+2)$. The Ricatti equation is

$$\begin{aligned}
\mathbf{P}_1(k+1) &= \mathbf{F}(k+1,k)\left(\mathbf{P}_{\kappa(k)}(k) - \mathbf{K}_{\kappa(k)}(k)\,\mathbf{R}^T_{xy,\kappa(k)}(k)\right)\\
&\quad \times \mathbf{F}^T(k+1,k) + \mathbf{I}.
\end{aligned} \tag{33}$$

Remark 4. In ideal communication networks, all the measurements arrive at the fusion filter in time; namely, the measurements are all ITMs. In this case, the above unified finite horizon H_∞ fusion filtering algorithm will be simplified into the following sequential finite horizon H_∞ fusion filtering algorithm, which can deal with the ITMs in real time according to their sampled sequence

$$\hat{\mathbf{z}}(k \mid k) = \mathbf{L}(k)\,\hat{\mathbf{x}}_N(k \mid k), \tag{34}$$

in which

$$\begin{aligned}
\hat{\mathbf{x}}_l(k \mid k) &= \hat{\mathbf{x}}_{l-1}(k \mid k)\\
&\quad + \mathbf{K}_{y,l}(k)\,\mathbf{e}_{y,l}(k \mid k-1), \quad l = 1, 2, \ldots, N,\\
\hat{\mathbf{x}}_0(k \mid k) &= \mathbf{F}(k,k-1)\,\hat{\mathbf{x}}_N(k-1 \mid k-1),\\
\mathbf{K}_{y,l}(k) &= \mathbf{H}_l(k)\,\mathbf{P}_l(k)\left[\mathbf{H}_l(k)\,\mathbf{P}_l(k)\,\mathbf{H}^T_l(k) + \mathbf{I}\right]^{-1},\\
\mathbf{e}_{y,l}(k \mid k-1) &= \mathbf{y}_l(k) - \mathbf{H}_l(k)\,\hat{\mathbf{x}}_{l-1}(k \mid k-1).
\end{aligned} \tag{35}$$

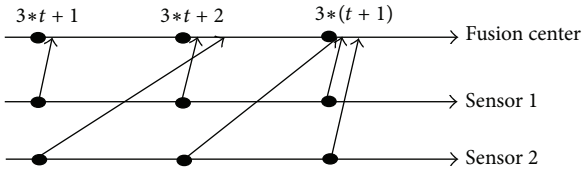


Figure 1: The measurement arrival scenes.

The corresponding Riccati equations are

$$\begin{aligned}\mathbf{P}_1(k) &= \mathbf{F}(k,k-1)\\ &\quad\times\left(\mathbf{P}_N(k-1)-\mathbf{R}_{xyz,l}(k)\,\mathbf{R}_{eyz,l}^{-1}(k)\right.\\ &\quad\left.\times\,\mathbf{R}_{xyz,N}^{T}(k-1)\right)\\ &\quad\times\,\mathbf{F}^{T}(k,k-1)+\mathbf{I}\\ \mathbf{P}_{l+1}(k) &= \mathbf{P}_l(k)-\mathbf{R}_{xyz,l}(k)\,\mathbf{R}_{eyz,l}^{-1}(k)\,\mathbf{R}_{xyz,l}^{T}(k),\qquad(36)\\ \mathbf{R}_{xyz,l}(k) &= \mathbf{P}_l(k)\left[(\mathbf{H}_l(k))^{T}\quad\mathbf{L}^{T}(k)\right],\\ \mathbf{R}_{eyz,l}(k) &:= \begin{bmatrix}\mathbf{H}_l(k)\\ \mathbf{L}(k)\end{bmatrix}\mathbf{P}_l(k)\\ &\quad\times\left[(\mathbf{H}_l(k))^{T}\quad\mathbf{L}^{T}(k)\right]+\begin{bmatrix}I & \mathbf{0}\\ \mathbf{0} & -\gamma^{2}\mathbf{I}\end{bmatrix}.\end{aligned}$$

4. Simulation

In order to illustrate the viability and the effectiveness of the proposed method, the discrete dynamical system as shown in (1) is considered in this section, in which $\mathbf{F}(k,k-1)=\left[\begin{smallmatrix}1&1\\0&1\end{smallmatrix}\right]$, $\mathbf{H}_i=[1,1],(i=1,2)$, $L=[0.1,0]$, $\gamma=1.5$. Moreover, the initial value is selected as $\mathbf{x}_0=[1,5]$, and $\mathbf{P}_0=\left[\begin{smallmatrix}1&1/2\\1/2&2\end{smallmatrix}\right]$.

The measurement arrival scenes are designed as follows. All the measurements are sampled in time. For Sensor 1, all the measurements arrive at the fusion filter in time. For Sensor 2, the measurements sampled at the moments with indexes modulo 3 equal to 1 or 2 arrive at the fusion filter with one-step delay, and other measurements arrive in time, as shown in Figure 1.

In this simulation, two simulation results are compared. The first one is the result of the proposed method in the above measurement arrival scene, which is for short marked as "Algorithm 1" in this section. The other one is the simulation result of the sequential fusion filtering method in the scene that all the measurements arrive in time, as shown in Remark 4, which is marked as "Algorithm 2."

According to the simulation results shown in Table 1, Figures 1 and 2, the following performances of the proposed algorithm are illustrated.

(1) The proposed algorithm could deal with different kinds of arrived measurements: ITMs (the measurements sampled by Sensor 1 and the ones sampled by Sensor 2 at the sampled moments with indexes divided by 3 exactly), ISDMs (the measurements sampled by Sensor 2 at the sampled moments with indexes modulo 3 equal to 1), and OOSMs (the measurements sampled by Sensor 2 at the sampled moments with indexes modulo 3 equal to 2).

(2) For Algorithm 1, at the fusion time with the indexes divided by 3 exactly, the delay measurements can all arrive at the fusion center. At this fusion time, the fusion filtering results of Algorithm 1are better than the ones of Algorithm 2. The mean absolute estimation errors at these fusion times of Algorithm 1 are 0.3787, while the one of Algorithm 2 is 0.3832. It is because more amount of information is applied to update the estimate at this fusion time in Algorithm 1 (Figure 3).

(3) It is implied that the proposed algorithm could deal with the delay measurements effectively.

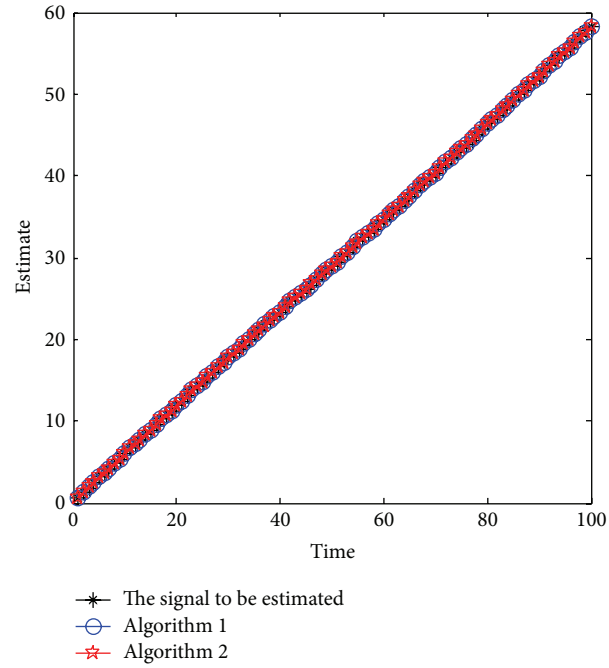


Figure 2: The estimate curves.

Table 1: The mean absolute estimation errors at filtering time with the indexes are divided by 5 exactly.

	Algorithm 1	Algorithm 2
The mean absolute estimation errors	0.3787	0.3832

5. Conclusion

In this paper, a unified finite horizon H_∞ filtering method is proposed for general networked dynamical systems, the fusion filter of which could receive none, one, or multiple measurements in a fusion period. According to the complex arrival scenes of networked measurements, a novel H_∞ performance criterion function is built to restrain the H_∞ filtering process. Based on the projection method in

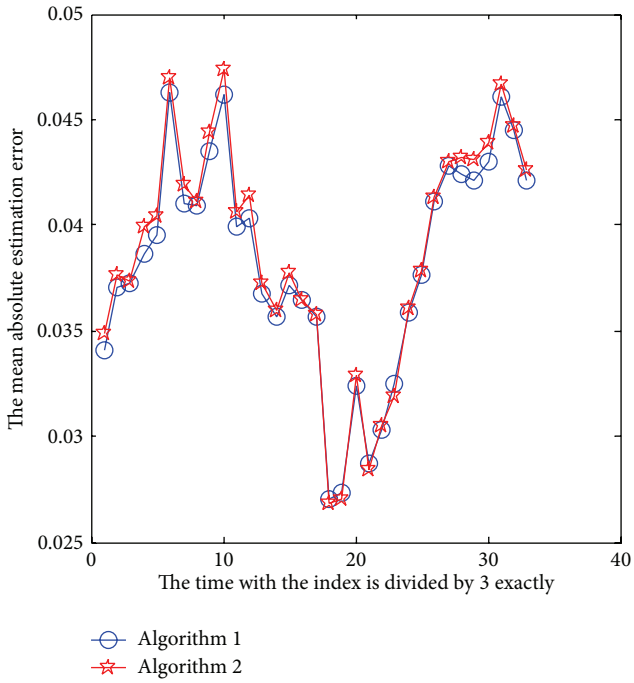


FIGURE 3: The absolute estimation error curves at the fusion time with the indexes are divided by 3 exactly.

Krein space, a novel H_∞ filtering method is proposed to uniformly deal with various delay measurements and ITMs in the centralized fusion frame. Otherwise, there are several interesting future directions along the line of this work:

(1) how to deal with the networked measurements in the distributed fusion frame in a uniform manner,

(2) how to deal with various quantified delay measurements for networked multisensor systems.

Conflict of Interests

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

Acknowledgment

The paper was supported by the National Natural Science Foundation of China under Grants 61333005, 61172133, 61174112, 61304258, and 61273075.

References

[1] Q. Wu, X. Litrico, and M. B. Alexandre, "Data reconciliation of an open channel flow network using modal decomposition," in *Proceedings of the 47th IEEE Conferenece on Decision and Control (CDC '08)*, pp. 3903–3910, Cancún, Mexico, December 2008.

[2] Z. He, *Research on comprehensive energy saving and coordinative optimization control of distributed pumping stations in urban drainage systems [Ph.D. thesis]*, Zhejiang University, Hangzhou, China, 2009.

[3] Y. Zhang, C. Liu, and X. Mu, "On stochastic finite-time control of discrete-time fuzzy systems with packet dropout," *Discrete Dynamics in Nature and Society*, vol. 2012, Article ID 752950, 18 pages, 2012.

[4] C. Wen, R. Liu, and T. Chen, "Linear unbiased state estimation with random one-step sensor delay," *Circuits, Systems, & Signal Processing*, vol. 26, no. 4, pp. 573–590, 2007.

[5] C. Su and C. Lu, "Interconnected network state estimation using randomly delayed measurements," *IEEE Transactions on Power Systems*, vol. 16, no. 4, pp. 870–878, 2001.

[6] H. Song, L. Yu, and W. Zhang, "H_∞ filtering of network-based systems with random delay," *Signal Processing*, vol. 89, no. 4, pp. 615–622, 2009.

[7] B. Shen, Z. Wang, H. Shu, and G. Wei, "H_∞ filtering for nonlinear discrete-time stochastic systems with randomly varying sensor delays," *Automatica*, vol. 45, no. 4, pp. 1032–1037, 2009.

[8] K. Zhang, X. R. Li, and Y. Zhu, "Optimal update with out-of-sequence measurements," *IEEE Transactions on Signal Processing*, vol. 53, no. 6, pp. 1992–2004, 2005.

[9] Q. Ge, T. Xu, X. Feng, and C. Wen, "Universal delayed Kalman filter with measurement weighted summation for the linear time invariant system," *Chinese Journal of Electronics*, vol. 20, no. 1, pp. 67–72, 2011.

[10] C. Wen, Q. Ge, and X. Feng, "Hybrid filter with predict-estimator and compensator for the linear time invariant delayed system," *Journal of Electronics*, vol. 26, no. 5, pp. 666–672, 2009.

[11] H. Zhang, G. Feng, and C. Han, "Linear estimation for random delay systems," in *Proceedings of the 49th IEEE Conferenece on Decision and Control (CDC '10)*, pp. 449–454, Atlanta, Ga, USA, 2010.

[12] H. Zhang, L. Xie, H. S. Zhang, and Y. C. Soh, "A reorganized innovation approach to linear estimation," *IEEE Transaction on Automatic Control*, vol. 49, no. 10, pp. 1810–1814, 2004.

[13] B. Chen, L. Yu, and W. Zhang, "Robust Kalman filtering for uncertain state delay systems with random observation delays and missing measurements," *IET Control Theory & Applications*, vol. 5, no. 17, pp. 1945–1954, 2011.

[14] S. Sun, "Linear minimum variance estimators for systems with bounded random measurement delays and packet dropouts," *Signal Processing*, vol. 89, no. 7, pp. 1457–1466, 2009.

[15] S. Sun, L. Xie, W. Xiao, and Y. C. Soh, "Optimal linear estimation for systems with multiple packet dropouts," *Automatica*, vol. 44, no. 5, pp. 1333–1342, 2008.

[16] M. Sahebsara, T. Chen, and S. L. Shah, "Optimal H_∞ filtering in networked control systems with multiple packet dropouts," *Systems & Control Letters*, vol. 57, no. 9, pp. 696–702, 2008.

[17] M. Moayedi, Y. K. Foo, and Y. C. Soh, "Optimal and suboptimal minimum-variance filtering in networked systems with mixed uncertainties of random sensor delays, packet dropouts and missing measurements," *International Journal of Control, Automation and Systems*, vol. 8, no. 6, pp. 1179–1188, 2010.

[18] H. Gao and T. Chen, "H_∞ estimation for uncertain systems with limited communication capacity," *IEEE Transactions on Automatic Control*, vol. 52, no. 11, pp. 2070–2084, 2007.

[19] X. Shen, E. Song, Y. Zhu, and Y. Luo, "Globally optimal distributed Kalman fusion with local out-of-sequence-measurement updates," *IEEE Transactions on Automatic Control*, vol. 54, no. 8, pp. 1928–1934, 2009.

[20] X. Shen, Y. Zhu, E. Song, and Y. Luo, "Optimal centralized update with multiple local out-of-sequence measurements," *IEEE Transactions on Signal Processing*, vol. 57, no. 4, pp. 1551–1562, 2009.

[21] B. Hassibi, A. H. Sayed, and T. Kailath, "Linear estimation in Krein spaces. II. Applications," *IEEE Transactions on Automatic Control*, vol. 41, no. 1, pp. 34–49, 1996.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 513765, 14 pages
http://dx.doi.org/10.1155/2014/513765

Research Article

Finite-Time Fault Detection for Large-Scale Networked Systems with Randomly Occurring Nonlinearity and Fault

Yong Zhang,[1] Huajing Fang,[2] and Zhenxing Liu[1]

[1] *School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan 430081, China*
[2] *School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China*

Correspondence should be addressed to Yong Zhang; zhangyong77@wust.edu.cn

Received 14 January 2014; Accepted 15 March 2014; Published 27 April 2014

Academic Editor: Housheng Su

The finite-time fault detection problem is investigated for a class of nonlinear quantized large-scale networked systems with randomly occurring nonlinearities and faults. A nonlinear Markovian jump system model with partially unknown transition probabilities is employed to describe this Makov data assignment pattern. Based on obtained model, in finite-time stable framework, the desired mode-dependent fault detection filters are constructed such that the augmented error systems are finite-time stochastically stable with H_∞ attenuation level. Especially, the sufficient conclusions provide quantitative relationship between network characteristic, quantization level, and finite-time system parameter with finite-time fault detection performance. The effectiveness of the proposed methods is demonstrated by simulation examples.

1. Introduction

The past decade has witnessed an ever increasing research interest in networked systems (NSs) due to their advantages in many aspects such as low cost, simple installation and maintenance, increased system agility, reduced system wiring, and high reliability. For large-scale NSs, multiple sensors and actuators are connected to the central control and fault detection station through communication medium. Actually, the introduction of communication network inevitably brings communication constraints to the systems analysis and synthesis. Especially, network-induced delay, packet dropout, and signal quantization have significant effect on the performance and stability, even fault of NSs [1–9].

Fault detection has been an active research field over the past decades because of the ever increasing demand for higher performance, higher safety, and reliability standard [10–12]. Recently, there are increasing interests on fault detection (FD) of networked systems. However, compared with the rich results in control and stability analysis of networked systems, only a limited number of contributions about FD have been found [13]. To deal with the FD of nonlinear networked systems (NCSs), [14] study T-S fuzzy model based fault detection for NCSs with stochastic mixed time delays and successive packet dropouts. In [15], an FD framework for a class of nonlinear NCSs via a shared communication medium has been proposed. In addition, the robust fault detection problem is studied in [16] for a class of NSs with unknown input, multiple state delays, and data missing. [17] study robust fault detection of NSs with delay distribution characterization.

For networked systems, stability analysis may be one of the most important research attention. Even almost all existing stability results are Lyapunov stability, actually, finite-time stability (FTS) [18] is a more practical concept which is utilized to study the behavior of the system within a finite time interval. Markovian jump systems (MJSs) are said to be stochastically finite-time stable if once we fix a finite-time interval, its state remains within prescribed bounds during this time interval. Obviously, MJSs may be not Lyapunov stochastically stable but finite-time stable. For large-scale networked systems, the nonlinear may be random due to stochastic change from network-induced phenomenon, which give rise to the so-called randomly occurring nonlinearities (RONs) [19]. Actually, compared

to deterministic fault of networked systems [13–17], faults may also occur in a probabilistic way and they arerandomly changeable in terms of their types and/or intensity. To the best of the authors' knowledge, up to now, almost no attention has been paid to the study of finite-time fault detection for nonlinear multiple channels data transmission networked systems with RONs and randomly occurring faults (ROFs); the main purpose of this paper is to shorten such a gap.

The main contributions of this paper are summarized as follows. (1) A Markovian jump system model with partially unknown transition probabilities is proposed to describe the multiple channels data transmission networked systems with channel-dependent measurement quantization; (2) based on the obtained model, by utilizing observer-based fault detection filter as residual generator, finite-time fault detection of large-scale networked systems is formulated as nonlinear finite-time H_∞ attenuation problem; and (3) by means of linear matrix inequalities (LMIs) method, sufficient conditions of finite-time stochastic stability are obtained and H_∞ attenuation level is guaranteed, and the explicit expression of the desired mode-dependent fault detection filters is also derived, which establish the quantitative relationship between quantization level and finite-time system parameters with fault detection performance. Especially, fault detection of traditional Markovian jump systems (known transition probability) and switched systems (unknown transition probability) with Lyapunov asymptotic stability (assuming finite-time system parameter $\sigma = 1$) can be contained as its special case. Numerical simulations are utilized to demonstrate the effectiveness of the presented methods.

Notation. Throughout the paper, the superscripts "–1" and "T" stand for the inverse and transpose of a matrix, respectively. $\mathbb{R}^n$ denotes the n-dimensional Euclidean space and $\|\cdot\|$ refers to Euclidean norm for vectors. $P > 0$ (≥ 0) means that P is a real symmetric positive definite (semidefinite) matrix. $\mathbf{E}\{x\}$ is the expectation of the stochastic variable x. **Prob**$\{\cdot\}$ means the occurrence probability of event "$\cdot$". I and 0 represent identity matrix and zero matrix; we utilize asterisk ($*$) to represent a term that is induced by symmetry and diag$\{\cdots\}$ stands for a block diagonal matrix.

2. Problem Formulation

Consider the discrete-time NSs with the following nonlinear system model:

$$\begin{aligned} x(k+1) &= Ax(k) + \alpha(k) N_1 g(x(k)) \\ &\quad + B_1 d(k) + \beta(k) B_2 f(k) \\ y(k) &= Cx(k), \end{aligned} \tag{1}$$

where $x(k) \in \mathbb{R}^{n_x}$ is the state vector, $y(k) \in \mathbb{R}^{n_y}$ is the measurement output, $d(k) \in \mathbb{R}^{n_d}$ is the disturbance input vector, and $f(k) \in \mathbb{R}^{n_f}$ is the fault vector to be detected. A, B_1, B_2, N_1, and C are known real matrices with appropriate dimension. The nonlinear function $g(x(k))$ satisfies $g(0) = 0$ and the following condition:

$$g^T(x(k))\, g(x(k)) \leq \delta x^T(k) G^T G x(k), \tag{2}$$

where $\delta > 0$ is the bounding parameters on the nonlinear function $g(x(k))$; G is known real constant matrix.

Random variables $\alpha(k) \in \mathbb{R}$ and $\beta(k) \in \mathbb{R}$ are utilized to account for the phenomena of randomly occurred nonlinearities and faults, which are assumed to be independent from each other and take values of 0 and 1 with

$$\begin{aligned} \text{Prob}(\alpha(k) = 1) &= \mathbf{E}\{\alpha(k)\} = \overline{\alpha}; \\ \text{Prob}(\beta(k) = 1) &= \mathbf{E}\{\beta(k)\} = \overline{\beta}, \end{aligned} \tag{3}$$

where $\overline{\alpha} \in [0, 1]$ and $\overline{\beta} \in [0, 1]$ are known constants. For large-scale complex networked systems, the nonlinearities and faults may be subject to random changes in environmental circumstances, for instance, network-induced random failures and repairs of components, sudden environmental disturbances, and so forth. Therefore, both the nonlinearities and faults may occur in a probabilistic way with certain types and intensity, which is particularly true in a networked environment.

In this paper, before accessing the observer, output signal $y(k)$ will be quantized by quantizer $q(\cdot)$ and quantized signal $\widetilde{y}(k)$ can be expressed as

$$\widetilde{y}(k) = q(y(k)) = \left[q_1(y_1(k)), \ldots, q_{n_y}\left(y_{n_y}(k)\right)\right]^T. \tag{4}$$

If quantizer $q(\cdot)$ is assumed to be logarithmic type, the set of quantization levels $q_j(\cdot)$ ($1 \leq j \leq n_y$) is described by $\mathfrak{F} = \{\pm u_i^{(j)}, u_i^{(j)} = \chi_j^i u_0^{(j)}, i = 0, \pm 1, \pm 2, \ldots\} \bigcup \{0\}$, $0 < \chi_j < 1$, $u_0^{(j)} > 0$. Each quantization level corresponds to a segment such that the quantizer maps the whole segment to this quantization level. In addition, these segments form a partition of $\mathbb{R}$; that is, they are disjoint and their union equals $\mathbb{R}$. The logarithmic quantizer $q_j(\cdot)$ is defined as

$$\begin{aligned} &q_j\left(y_j(k)\right) \\ &= \begin{cases} u_i^{(j)}, & \dfrac{1}{1+\overline{\delta}_j} u_i^{(j)} < y_j(k) \leq \dfrac{1}{1-\overline{\delta}_j} u_i^{(j)} \\ 0, & y_j(k) = 0 \\ -q_j\left(-y_j(k)\right), & y_j(k) < 0, \end{cases} \end{aligned} \tag{5}$$

where $\overline{\delta}_j = (1 - \chi_j)/(1 + \chi_j)$. Similar to [20], we have the expression of $q_j(y_j(k)) = (1 + \Delta_j(k)) y_j(k)$ with $|\Delta_j(k)| \leq \overline{\delta}_j$. Defining $\Delta(k) = \text{diag}\{\Delta_1(k), \ldots, \Delta_{n_y}(k)\}$, the measurements after quantization have the following form:

$$\begin{aligned} \widetilde{y}(k) &= (I + \Delta(k)) y(k) \\ &= (I + \Delta(k)) Cx(k) + (I + \Delta(k)) D_1 d(k) \\ &\quad + (I + \Delta(k)) D_2 f(k). \end{aligned} \tag{6}$$

Actually, the quantized effect can be transformed into sector bound uncertainties. By defining $\overline{\Delta} = \operatorname{diag}\{\delta_1, \ldots, \delta_{n_y}\}$ and $F(k) = \Delta(k)\overline{\Delta}^{-1}$, we obtain an unknown real-valued time-varying matrix $F(k)$ satisfying $F(k)F^T(k) \leq I$.

In this paper, two-valued function $\theta_l(k) : Z \to \{0, 1\}$ ($l = 1, \ldots, n_y$) is used to describe the lth channels transmission status in sampling time k, where 1 means successful data transmission and 0 means data loss. Specifically, only corresponding data packet access the communication medium, that is, $\theta_l(k) = 1$, the quantized output $q_l(y_l(k))$ to observer of lth channels is available. Otherwise, when $\theta_l(k) = 0$, the output of lth channels will be zero by the observer and $q_l(y_l(k))$ will be ignored due to its being unavailable. If we regard $\overline{y}_l(k)$ as the lth channels signal received by the observer, we describe the transmission dynamics of lth channels as:

$$\overline{y}_l(k) = \theta_l(k)\, q_l\left(y_l(k)\right) \quad \left(l = 1, \ldots, n_y\right). \tag{7}$$

In the need of investigation, we define transmission matrix as $M_{n_y} \triangleq \{M_{n_y}^{0,1}, M_{n_y}^{1,1}, \ldots, M_{n_y}^{1,n_y}, \ldots, M_{n_y}^{n_y-1,1}, \ldots, M_{n_y}^{n_y-1,n_y}, M_{n_y}^{n_y,1}\}$, and matrix $M_{n_y}^{s,t} = \operatorname{diag}\{\theta_1(k), \ldots, \theta_{n_y}(k)\}$ can be expressed in the following form:

$$\begin{aligned}
M_{n_y}^{0,1} &\triangleq \operatorname{diag}\left\{\underbrace{0, \ldots, 0}_{n_y}\right\},\\
M_{n_y}^{1,1} &\triangleq \operatorname{diag}\left\{1, \underbrace{0, \ldots, 0}_{n_y-1}\right\}, \ldots,\\
M_{n_y}^{1,n_y} &\triangleq \operatorname{diag}\left\{\underbrace{0, \ldots, 0}_{n_y-1}, 1\right\}, \ldots,\\
M_{n_y}^{n_y-1,1} &\triangleq \operatorname{diag}\left\{\underbrace{1, \ldots, 1}_{n_y-1}, 0\right\}, \ldots,\\
M_{n_y}^{n_y-1,n_y} &\triangleq \operatorname{diag}\left\{0, \underbrace{1, \ldots, 1}_{n_y-1}\right\},\\
M_{n_y}^{n_y,1} &\triangleq \operatorname{diag}\left\{\underbrace{1, \ldots, 1}_{n_y}\right\}.
\end{aligned} \tag{8}$$

According to above discussion, we achieve the quantized output dynamics as

$$\overline{y}(k) = M_{n_y}^{\mu(k)}\widetilde{y}(k) = M_{n_y}^{\mu(k)} q\left(\widetilde{y}(k)\right), \tag{9}$$

where $M_{n_y}^{\mu(k)} \in M_{n_y}$. Let $\mu(k)$ be a Markov chain taking values in a finite state space $\mathfrak{R} = \{1, 2, \ldots, 2^{n_y}\}$ with transition probability matrix $\Lambda = (\rho_{ij})$ given by

$$\rho_{ij} = \operatorname{Prob}\left(\mu(k+1) = j \mid \mu(k) = i\right), \tag{10}$$

where $\rho_{ij} \geq 0$, $\forall i, j \in \mathfrak{R}$, and $\sum_{j=1}^{2^{n_y}} \rho_{ij} = 1$. It is suitable that we assume that Markov chain $\mu(k)$ is independent of the stochastic variables $\alpha(k)$ and $\beta(k)$.

In this paper, the following mode-dependent observer-based fault detection filter is constructed as a residual generator:

$$\begin{aligned}
\widehat{x}(k+1) &= A\widehat{x}(k) + L_{\mu(k)}\left(\overline{y}(k) - \widehat{y}(k)\right),\\
\widehat{y}(k) &= C\widehat{x}(k),\\
r(k) &= V_{\mu(k)}\left(\overline{y}(k) - \widehat{y}(k)\right),
\end{aligned} \tag{11}$$

where $\widehat{x}(k) \in \mathbb{R}^{n_{\widehat{x}}}$ and $\widehat{y}(k) \in \mathbb{R}^{n_{\widehat{y}}}$ represent the state and output estimation vectors, respectively. $r(k)$ is the residual signal. FDF parameters are the observer gain matrices $L_{\mu(k)}$ and residual weighting matrices $V_{\mu(k)}(\forall \mu(k) \in \mathfrak{R})$. Observer with above structure is assumed to jump synchronously with the modes in (9), which is hereby mode-dependent.

Let the estimation error be $e_1(k) = x(k) - \widehat{x}(k)$, then error systems can be obtained by combining (1), (9), and (11):

$$\begin{aligned}
e_1(k+1) &= \left(A - L_{\mu(k)}C\right)e_1(k) - L_{\mu(k)}\\
&\quad \times \left[M_p^{\mu(k)}\left(I + \Delta(k)\right)C - C\right]x(k)\\
&\quad + \alpha(k)\, N_1 g(x(k)) + B_1 d(k) + \beta(k)\, B_2 f(k).
\end{aligned} \tag{12}$$

By setting $r_e(k) = r(k) - f(k)$, $e(k) = [x^T(k), e_1^T(k)]^T$, $w(k) = [d^T(k), f^T(k)]^T$, $g(e(k)) = [g^T(x(k)), g^T(e_1(k))]^T$ and integrating (1) and (12), we obtain the following augmented error systems:

$$\begin{aligned}
e(k+1) &= \overline{A}_{\mu(k)}e(k) + \alpha(k)\,\overline{N}_1 H g_1(e(k))\\
&\quad + \left[\overline{B}_1 + \beta(k)\,\overline{B}_2\right]w(k)\\
r_e(k) &= \overline{C}_{\mu(k)}e(k) + \beta(k)\,\overline{D}w(k),
\end{aligned} \tag{13}$$

where $\overline{A}_{\mu(k)} = \begin{bmatrix} A & 0 \\ L_{\mu(k)}C - L_{\mu(k)}M_p^{\mu(k)}(I+\Delta(k))C & A - L_{\mu(k)}C \end{bmatrix}$, $\overline{B}_1 = \begin{bmatrix} B_1 & 0 \\ B_1 & 0 \end{bmatrix}$, $\overline{B}_2 = \begin{bmatrix} 0 & B_2 \\ 0 & B_2 \end{bmatrix}$ $\overline{N}_1 = \begin{bmatrix} N_1 \\ N_1 \end{bmatrix}$, $\overline{C}_{\mu(k)} = [V_{\mu(k)}M_p^{\mu(k)}(I + \Delta(k))C - V_{\mu(k)}C, V_{\mu(k)}C]$, $D = \begin{bmatrix} 0 & -I \end{bmatrix}$, and $H = \begin{bmatrix} I & 0 \end{bmatrix}$; $r_e(k)$ is the residual error which contains the stochastic fault information of occurrence time and location. In addition, the transition probabilities of jumping process $\{\mu(k), k \geq 0\}$ are assumed to be partly accessed; that is, some elements are unknown. For notation clarity, we denote $\mathfrak{R} = \mathfrak{R}_K^i + \mathfrak{R}_{UK}^i$ ($\forall i \in \mathfrak{R}$) with $\mathfrak{R}_K^i \triangleq \{j : \rho_{ij}$ is known$\}$ and $\mathfrak{R}_{UK}^i \triangleq \{j : \rho_{ij}$ is unknown$\}$.

Now, to present the main objective of this paper more precisely, we need the following finite-time stochastic stability definition for augmented error systems (13), which are essential for the later development.

Definition 1. Augmented error systems (13) are said to be finite-time stochastically stable with respect to (c_1, c_2, R, N) for $w(k) = 0$ and every initial condition $e(0)$, where $R > 0$ are positive define matrix, $0 < c_1 < c_2$ and $N \in \mathbb{Z}$, if

$$e^T(0) Re(0) \le c_1 \Longrightarrow \mathbf{E}\left[e^T(k) Re(k)\right] < c_2. \tag{14}$$

The purpose of this paper is to design mode-dependent observer-based fault detection parameters $L_{\mu(k)}$ and $V_{\mu(k)}$ such that augmented error systems (13) are finite-time stochastically stable; under zero-initial condition, for any nonzero $w(k)$, we have

$$\sum_{k=0}^{\infty} \mathbf{E}\left\{\|r_e(k)\|^2\right\} < \gamma^2 \sum_{k=0}^{\infty} \|w(k)\|^2. \tag{15}$$

In order to detect the faults, the widely adopted approach is to choose an appropriate threshold J_{th} and residual evolution function $J(r_e)$ as

$$J(r_e) = \mathbf{E}\left\{\sum_{k=k_0}^{k=k_0+\kappa} r_e^T(k) r_e(k)\right\}^{1/2}, \qquad J_{\text{th}} = \sup_{d(k)\in\ell_2, f(k)=0} J(r_e), \tag{16}$$

where k_0 denotes the initial evaluation time instant and κ denotes the evaluation time steps.

Based on the threshold, the occurrence of fault can be detected by comparing J_{th} and $J(r_e)$ according to the following test:

$$J(r_e) \ge J_{\text{th}} \Longrightarrow \text{alarm for fault}, \qquad J(r_e) < J_{\text{th}} \Longrightarrow \text{no fault}. \tag{17}$$

3. Main Results

The following theorem provides a sufficient condition under which the augmented error systems (13) are finite-time stochastically stable and the residual estimation error $r_e(k)$ satisfies the H_∞ criterion (15) under zero-initial condition.

Theorem 2. *Augmented error systems* (13) *are finite-time stochastically stable with respect to* (c_1, c_2, R_i, N) *and satisfy* H_∞ *performance level* (15)*, if there exist positive define matrices* $P_i > 0$ $(i \in \mathfrak{R})$*, scalars* $\varepsilon_s > 0$*,* $\overline{\sigma} \ge 1$*, and* $\gamma > 0$ *such that the following matrix inequalities hold:*

$$\begin{bmatrix} \Xi_{11} & \Xi_{12}^T & \Xi_{13}^T \Xi_{22} \\ * & \text{diag}\{I, I\} & 0 \\ * & * & \Xi_{22} \end{bmatrix} < 0, \tag{18}$$

$$\frac{c_1^2}{\underline{\lambda}} < \frac{c_2^2}{\overline{\sigma}^N \overline{\lambda}}, \tag{19}$$

where

$$\Xi_{12}^T = \begin{bmatrix} \sqrt{\overline{\beta}}\overline{C}_i^T & \sqrt{1-\overline{\beta}}\overline{C}_i^T \\ 0 & 0 \\ \sqrt{\overline{\beta}}\overline{D}^T & 0 \end{bmatrix},$$

$$\Xi_{13}^T = \begin{bmatrix} 0 & \sqrt{1-\overline{\alpha}}\overline{A}_i^T & \sqrt{\overline{\alpha}}\overline{A}_i^T \\ 0 & 0 & \sqrt{\overline{\alpha}}\overline{N}_1^T \\ \sqrt{\overline{\beta}\left(1-\overline{\beta}\right)}\overline{B}_2^T & \sqrt{1-\overline{\alpha}}\left(\overline{B}_2^T + \overline{\beta}\overline{B}_2^T\right) & \sqrt{\overline{\alpha}}\left(\overline{B}_2^T + \overline{\beta}\overline{B}_2^T\right) \end{bmatrix},$$

$$\Xi_{11} = \text{diag}\left\{-\lambda P_i - \varepsilon_1 \delta H^T G^T G H, -\varepsilon_1 I, -\gamma^2 I\right\},$$

$$\Xi_{22} = \text{diag}\left\{-\Upsilon_j, -\Upsilon_j, -\Upsilon_j\right\}, \qquad \Upsilon_j \triangleq \begin{cases} \dfrac{1}{\rho_K^i} P_K^i, & \forall j \in \mathfrak{R}_K^i \\ P_j, & \forall j \in \mathfrak{R}_{UK}^i, \end{cases}$$

$$P_K^i = \sum_{j\in\mathfrak{R}_K^i} \rho_{ij} P_j, \qquad \rho_K^i = \sum_{j\in\mathfrak{R}_K^i} \rho_{ij}, \qquad \widetilde{P}_i = R^{-1/2} P_i R^{-1/2},$$

$$\overline{\lambda} = \max_{i\in\mathfrak{R}} \lambda_{\max}\left(\widetilde{P}_i\right), \qquad \underline{\lambda} = \min_{i\in\mathfrak{R}} \lambda_{\min}\left(\widetilde{P}_i\right). \tag{20}$$

Proof. For augmented error systems (13) with $w(k) = 0$, we construct stochastic Lyapunov function as

$$V\left(e\left(k\right),\mu\left(k\right)\right) = e^{T}\left(k\right)P_{\mu(k)}e\left(k\right), \tag{21}$$

where $P_{\mu(k)} > 0$ ($\mu(k) \in \mathfrak{R}$). If we denote $\mu(k) = i$, then $P_{\mu(k)}$ is P_i. So, along the solution of (13), we obtain

$$\begin{aligned}
&\mathbf{E}\left[\Delta V\left(e\left(k\right)\right)\right]\\
&= \mathbf{E}\left[V\left(e\left(k+1\right),k+1 \mid e\left(k\right),\mu\left(k\right)\right)\right] - \overline{\sigma}V\left(e\left(k\right)\right)\\
&= e^{T}\left(k+1\right)\sum_{j\in\mathfrak{R}}\rho_{ij}P_{j}e\left(k+1\right) - \overline{\sigma}e^{T}\left(k\right)P_{i}e\left(k\right)\\
&= e^{T}\left(k+1\right)\left[\sum_{j\in\mathfrak{R}_{K}^{i}}\rho_{ij}P_{j} + \sum_{j\in\mathfrak{R}_{UK}^{i}}\rho_{ij}P_{j}\right]e\left(k+1\right)\\
&\quad - \overline{\sigma}e^{T}\left(k\right)\left[\sum_{j\in\mathfrak{R}_{K}^{i}}\rho_{ij} + \sum_{j\in\mathfrak{R}_{UK}^{i}}\rho_{ij}\right]P_{i}e\left(k\right)\\
&= e^{T}\left(k+1\right)P_{K}^{i}e\left(k+1\right) - \overline{\sigma}e^{T}\left(k\right)\rho_{K}^{i}e\left(k\right)\\
&\quad + \sum_{j\in\mathfrak{R}_{UK}^{i}}\rho_{ij}\left[e^{T}\left(k+1\right)P_{j}e\left(k+1\right) - \overline{\sigma}e^{T}\left(k\right)P_{i}e\left(k\right)\right].
\end{aligned} \tag{22}$$

For any scalar $\varepsilon_1 > 0$, it follows readily from (2) that

$$\varepsilon_{1}\left(\delta e^{T}\left(k\right)H^{T}G^{T}GHe\left(k\right) - g^{T}\left(e\left(k\right)\right)H^{T}Hg\left(e\left(k\right)\right)\right) \geq 0. \tag{23}$$

Combining (13) and (22)-(23) and denoting $\xi(k) = [e^T(k), g^T(e(k))H^T]^T$ lead to

$$\mathbf{E}\left[\Delta V\left(\mathfrak{F}\left(k\right),k\right)\right] = \xi^{T}\left(k\right)\left\{\Xi_{1}^{i} + \sum_{j\in\mathfrak{R}_{UK}^{i}}\rho_{ij}\Xi_{2}^{i}\right\}\xi\left(k\right), \tag{24}$$

where

$$\begin{aligned}
\Xi_{1}^{i} &= \begin{bmatrix}\overline{A}_{i}^{T}P_{K}^{i}\overline{A}_{i} - \overline{\sigma}\rho_{K}^{i}P_{i} - \varepsilon_{1}\delta\rho_{K}^{i}H^{T}G^{T}GH & \overline{\alpha}\overline{A}_{i}^{T}P_{K}^{i}\overline{N}_{1}\\ * & \overline{\alpha}\overline{N}_{1}^{T}P_{K}^{i}\overline{N}_{1} - \varepsilon_{1}\rho_{K}^{i}I\end{bmatrix},\\
\Xi_{2}^{i} &= \begin{bmatrix}\overline{A}_{i}^{T}P_{j}\overline{A}_{i} - \lambda P_{i} - \varepsilon_{1}\delta H^{T}G^{T}GH & \overline{\alpha}\overline{A}_{i}^{T}P_{j}\overline{N}\\ * & \overline{\alpha}\overline{N}^{T}P_{j}\overline{N} - \varepsilon_{1}I\end{bmatrix}.
\end{aligned} \tag{25}$$

By Schur complement, (18) implies $\Xi_1^i < 0$ and $\Xi_2^i < 0$, which means that

$$\mathbf{E}\left[V\left(\xi\left(k+1\right)\right)\right] < \overline{\sigma}V\left(\xi\left(k\right)\right). \tag{26}$$

Applying above inequality interactively, we obtain

$$\mathbf{E}\left[V\left(\xi\left(k\right)\right)\right] < \overline{\sigma}^{k}V\left(\xi\left(0\right)\right), \quad k = 1,\ldots,N. \tag{27}$$

Now, letting $\widetilde{P}_i = R^{-1/2}P_iR^{-1/2}$ and using the fact that $\overline{\sigma} \geq 1$, we have

$$\begin{aligned}
\overline{\sigma}^{k}V\left(\xi\left(0\right),\mu\left(0\right)\right) &= \overline{\sigma}^{k}\xi^{T}\left(0\right)P_{i}\xi\left(0\right)\\
&\leq \overline{\sigma}^{k}\lambda_{\max}\left(\widetilde{P}_{i}\right)\xi^{T}\left(0\right)R\xi\left(0\right)\\
&\leq \overline{\sigma}^{N}\overline{\lambda}\left(\widetilde{P}_{i}\right)c_{1}^{2}.
\end{aligned} \tag{28}$$

Furthermore, according to the following fact,

$$\begin{aligned}
\mathbf{E}\left[V\left(\xi\left(k\right)\right)\right] &= \mathbf{E}\left[\xi^{T}\left(k\right)P_{i}\xi\left(k\right)\right]\\
&= \mathbf{E}\left[\xi^{T}\left(k\right)R^{1/2}\widetilde{P}_{i}R^{1/2}\xi\left(k\right)\right]\\
&\geq \lambda_{\min}\left(\widetilde{P}_{i}\right)\xi^{T}\left(k\right)R\xi\left(k\right) \geq \underline{\lambda}\xi^{T}\left(k\right)R\xi\left(k\right).
\end{aligned} \tag{29}$$

Putting (27) and (29) together may yield

$$\mathbf{E}\left[\xi^{T}\left(k\right)R\xi\left(k\right)\right] \leq \frac{\overline{\sigma}^{N}}{\underline{\lambda}}\overline{\lambda}c_{1}^{2}. \tag{30}$$

Thus, augmented error systems (13) are finite-time stochastically stable from Definition 1.

On the other hand, under zero initial condition, for $w(k) \neq 0$ in (13), we consider the following performance index:

$$J_{N} = \mathbf{E}\sum_{k=0}^{N}\left[r_{e}^{T}\left(k\right)r_{e}\left(k\right) - \gamma w^{T}\left(k\right)w\left(k\right)\right]. \tag{31}$$

Let $\zeta(k) = [\xi^T(k), w^T(k)]^T$, we have

$$\begin{aligned}
J_{N} &= \mathbf{E}\sum_{k=0}^{N}\left[r_{e}^{T}\left(k\right)r_{e}\left(k\right) - \gamma^{2}w^{T}\left(k\right)w\left(k\right) + \Delta V\left(\mathfrak{F}\left(k\right)\right)\right]\\
&\quad - \mathbf{E}V\left(N+1\right)\\
&\leq \mathbf{E}\sum_{k=0}^{N}\left[r_{e}^{T}\left(k\right)r_{e}\left(k\right) - \gamma^{2}w^{T}\left(k\right)w\left(k\right) + \Delta V\left(\mathfrak{F}\left(k\right)\right)\right]\\
&= \sum_{k=0}^{N}\zeta^{T}\left(k\right)\left[\overline{\Xi}_{1}^{i} + \sum_{j\in\mathfrak{R}_{UK}^{i}}\rho_{ij}\overline{\Xi}_{2}^{i}\right]\zeta\left(k\right),
\end{aligned} \tag{32}$$

where

$$\overline{\Xi}_1^i = \begin{bmatrix} \Xi_1^i & \Omega_1^i \\ * & \Omega_2^i \end{bmatrix},$$

$$\Omega_1^i = \left[\overline{B}_1^T P_K^i \overline{A}_i + \overline{\beta}\overline{B}_2^T P_K^i \overline{A}_i + \rho_K^i \overline{\beta}\overline{C}_i^T \overline{D} \;\; \overline{\alpha}\overline{B}_1^T P_K^i \overline{N}_1 + \overline{\alpha}\overline{\beta}\overline{B}_2^T P_K^i \overline{N}_1\right]^T,$$

$$\overline{\Xi}_2^i = \begin{bmatrix} \Xi_2^i & \Omega_3^i \\ * & \Omega_4^i \end{bmatrix}, \tag{33}$$

$$\Omega_2^i = \overline{B}_1^T P_K^i \overline{B}_1 + 2\overline{\beta}\overline{B}_1^T P_K^i \overline{B}_2 + \overline{\beta}\overline{B}_2^T P_K^i \overline{B}_2 + \overline{\beta}\rho_K^i \overline{D}^T \overline{D} - \gamma^2 \rho_K^i I,$$

$$\Omega_3^i = \left[\overline{B}_1^T P_j \overline{A}_i + \overline{\beta}\overline{B}_2^T P_j \overline{A}_i + \overline{\beta}\overline{C}_i^T \overline{D} \;\; \overline{\alpha}\overline{B}_1^T P_j \overline{N}_1 + \overline{\alpha}\overline{\beta}\overline{B}_2^T P_j \overline{N}_1\right]^T,$$

$$\Omega_4^i = \overline{B}_1^T P_j \overline{B}_1 + 2\overline{\beta}\overline{B}_1^T P_j \overline{B}_2 + \overline{\beta}\overline{B}_2^T P_j \overline{B}_2 + \overline{\beta}\overline{D}^T \overline{D} - \gamma^2 I.$$

By Schur complement, (18) is equivalent to $\overline{\Xi}_1^i + \sum_{j\in\mathfrak{R}_{UK}^i} \rho_{ij}\overline{\Xi}_2^i < 0$, which implies $J_N < 0$, that is (15), so the proof is completed. □

Remark 3. As [18] pointed out, Lyapunov asymptotic stability (LAS) is independent on finite-time stability (FTS); that is, a system which is FTS may not be LAS; conversely a LAS system could not be FTS. In this paper, we investigate more practical fault detection of networked systems in finite-time stochastic stability (FTSS) framework rather than in Lyapunov stochastic stability (LSS) framework [1–9, 13–17]. Especially, if we let $\overline{\sigma} = 1$, Theorem 2 will degenerate to fault detection in FTSS framework, in view of this, our conclusion is more general.

Next, sufficient conditions on the existence of mode-dependent observer-based finite-time fault detection filters would be given, the slack matrix will be constructed with a special structure to eliminate the cross coupling between system matrices and Lyapunov matrices among different operation modes, which allows us to obtain a solution within strict linear matrix inequalities framework for the proposed systems.

Theorem 4. *Augmented error system* (13) *is finite-time stochastically stable with respect to* (c_1, c_2, R, N) *and satisfies* H_∞ *performance level* (15) *if there exist matrices* $\overline{L}_i$, $\overline{V}_i$, X_i, Y_i, T_i, $P_{i1} > 0$, P_{i2}, $P_{i3} > 0$ $(i \in \mathfrak{R})$, *scalars* $\varepsilon_s > 0$ $(s = 1, 2, 3)$, $\beta \geq 1$, *and* $\gamma > 0$ *such that* (19) *and the following linear matrix inequalities hold:*

$$\begin{bmatrix} \Xi_{11}^i & 0 & \Xi_{12}^i & 0 & \sqrt{1-\overline{\alpha}}\Xi_{13}^i & \sqrt{\overline{\alpha}}\Xi_{13}^i & \Xi_{14}^i \\ * & \Xi_{22} & 0 & \Xi_{23}^i & \Xi_{24}^i & \Xi_{24}^i + \overline{\Xi}_{24}^i & 0 \\ * & * & \Xi_{33} & 0 & 0 & 0 & \Xi_{34}^i \\ * & * & * & \Xi_{44}^i & 0 & 0 & 0 \\ * & * & * & * & \Xi_{44}^i & 0 & \Xi_{45}^i \\ * & * & * & * & * & \Xi_{44}^i & \Xi_{45}^i \\ * & * & * & * & * & * & \Xi_{55} \end{bmatrix} < 0, \tag{34}$$

where

$$\Xi_{11}^i = \begin{bmatrix} -\lambda P_{i1} - \varepsilon_1 \delta G^T G & -\lambda P_{i2} \\ * & -\lambda P_{i3} \end{bmatrix},$$

$$\Xi_{12}^i = \begin{bmatrix} \sqrt{\overline{\beta}}C^T M_p^i \overline{V}_i^T - \sqrt{\overline{\beta}}C^T \overline{V}_i^T & \sqrt{1-\overline{\beta}}C^T M_p^i \overline{V}_i^T - \sqrt{1-\overline{\beta}}C^T \overline{V}_i^T \\ \sqrt{\overline{\beta}}C^T \overline{V}_i^T & \sqrt{1-\overline{\beta}}C^T \overline{V}_i^T \end{bmatrix},$$

$$\Xi_{13}^{i} = \begin{bmatrix} A^T X_i^T + C^T \overline{L}_i^T - C^T M_p^i \overline{L}_i^T & A^T U_i^T + C^T \overline{L}_i^T - C^T M_p^i \overline{L}_i^T \\ A^T Y_i^T - C^T \overline{L}_i^T & A^T Y_i^T - C^T \overline{L}_i^T \end{bmatrix},$$

$$\Xi_{14}^{i} = \begin{bmatrix} -\varepsilon_2 C^T \overline{\Delta}^T & 0 & \varepsilon_3 C^T \overline{\Delta}^T & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\Xi_{23}^{i} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ B_2^T X_i^T + B_2^T Y_i^T & B_2^T U_i^T + B_2^T Y_i^T \end{bmatrix},$$

$$\Xi_{24}^{i} = \begin{bmatrix} 0 & 0 \\ B_1^T X_i^T + B_1^T Y_i^T & B_1^T U_i^T + B_1^T Y_i^T \\ \overline{\beta} B_2^T X_i^T + \overline{\beta} B_2^T Y_i^T & \overline{\beta} B_2^T U_i^T + \overline{\beta} B_2^T Y_i^T \end{bmatrix},$$

$$\overline{\Xi}_{24}^{i} = \begin{bmatrix} N_1^T X_i^T + N_1^T Y_i^T & N_1^T U_i^T + N_1^T Y_i^T \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$\Xi_{34}^{i} = \begin{bmatrix} 0 & 0 & 0 & \overline{V}_i M_p^i \\ 0 & 0 & 0 & \overline{V}_i M_p^i \end{bmatrix}, \qquad \Xi_{45}^{i} = \begin{bmatrix} 0 & \overline{L}_i M_p^i & 0 & 0 \\ 0 & \overline{L}_i M_p^i & 0 & 0 \end{bmatrix},$$

$$\Xi_{44}^{i} = \begin{bmatrix} \Upsilon_{j1} - X_i - X_i^T & \Upsilon_{j2} - Y_i - U_i^T \\ * & \Upsilon_{j3} - Y_i - Y_i^T \end{bmatrix},$$

$$\Xi_{22}^{i} = \operatorname{diag}\left\{-\varepsilon_1 I, -\gamma^2 I, -\gamma^2 I\right\},$$

$$\Xi_{55}^{i} = \operatorname{diag}\left\{-\varepsilon_2 I, -\varepsilon_2 I, -\varepsilon_3 I, -\varepsilon_3 I\right\}, \qquad \Xi_{33}^{i} = \operatorname{diag}\left\{-I, -I\right\}. \tag{35}$$

Then, there exists mode-dependent observer-based finite-time fault detection filters such that the augmented error systems (13) *with partly unknown transition probabilities are finite-time stochastically stable with H_∞ performance level* (15). *Moreover, if LMIs* (34) *have feasible solution, the desired finite-time fault detection filters can be given by*

$$L_i = Y_i^{-1} \overline{L}_i, \qquad V_i = \overline{V}_i \quad (i \in \mathfrak{R}). \tag{36}$$

Proof. For an arbitrary matrix $R_i = \begin{bmatrix} X_i & Y_i \\ U_i & Y_i \end{bmatrix}$, $\forall i \in \mathfrak{R}$, assuming that Y_i is inverse, we have the following fact:

$$\begin{gathered} \left(\frac{1}{\rho_K^i} P_K^i - R_i\right)\left(\frac{1}{\rho_K^i} P_K^i\right)^{-1}\left(\frac{1}{\rho_K^i} P_K^i - R_i\right)^T \geq 0; \\ \left(P_j - R_i\right) P_j^{-1} \left(P_j - R_i\right)^T \geq 0. \end{gathered} \tag{37}$$

Then, we have $\Upsilon_j - R_i^T - R_i \geq -R_i \Upsilon_j^{-1} R_i^T$; performing a congruence transformation to (18) by $J_1 = \operatorname{diag}\{I, I, I, I, I, \Upsilon_j^{-1} R_i^T, \Upsilon_j^{-1} R_i^T, \Upsilon_j^{-1} R_i^T\}$, we obtain

$$\begin{bmatrix} \Xi_{11} & \Xi_{12}^T & \Xi_{13}^T \overline{\Xi}_{14} \\ * & \operatorname{diag}\{I, I\} & 0 \\ * & * & \overline{\Xi}_{22} \end{bmatrix} < 0, \tag{38}$$

where $\overline{\Xi}_{14} = \operatorname{diag}\{R_i^T, R_i^T, R_i^T\}$ and $\overline{\Xi}_{22} = \operatorname{diag}\{\Upsilon_j - R_i^T - R_i, \Upsilon_j - R_i^T - R_i, \Upsilon_j - R_i^T - R_i\}$.

Define matrices variables $\overline{L}_i = Y_i L_i$ and $\overline{V}_i = V_i$, noting that

$$\begin{aligned} \overline{A}_i^T R_i^T &= \begin{bmatrix} A^T X_i^T + C^T \overline{L}_i^T - C^T M_p^i \overline{L}_i^T & A^T U_i^T + C^T \overline{L}_i^T - C^T M_p^i \overline{L}_i^T \\ A^T X_i^T - C^T \overline{L}_i^T & A^T X_i^T - C^T \overline{L}_i^T \end{bmatrix} \\ &\quad + \begin{bmatrix} -C^T \Delta^T(k) M_p^i \overline{L}_i^T & -C^T \Delta^T(k) M_p^i \overline{L}_i^T \\ 0 & 0 \end{bmatrix} = \underline{A}_i^T + \widehat{A}^T F^T(k) \breve{A}_i^T \end{aligned} \tag{39}$$

$$\overline{C}_i^T = \begin{bmatrix} C^T M_p^i \overline{V}_i^T - C^T \overline{V}_i^T \\ C^T \overline{V}_i^T \end{bmatrix} + \begin{bmatrix} C^T \Delta^T(k) M_p^i \overline{V}_i^T \\ 0 \end{bmatrix} = \underline{C}_i^T + \widehat{C}^T F^T(k) \breve{C}_i^T,$$

where

$$\underline{A}_i^T = \begin{bmatrix} A^T X_i^T + C^T \overline{L}_i^T - C^T M_p^i \overline{L}_i^T & A^T U_i^T + C^T \overline{L}_i^T - C^T M_p^i \overline{L}_i^T \\ A^T X_i^T - C^T \overline{L}_i^T & A^T X_i^T - C^T \overline{L}_i^T \end{bmatrix},$$

$$\widehat{A}^T = \begin{bmatrix} -C^T \overline{\Delta}^T \\ 0 \end{bmatrix}, \qquad \breve{A}_i^T = \begin{bmatrix} M_p^i \overline{L}_i^T & M_p^i \overline{L}_i^T \end{bmatrix}, \qquad \underline{C}_i^T = \begin{bmatrix} C^T M_p^i \overline{V}_i^T - C^T \overline{V}_i^T \\ C^T \overline{V}_i^T \end{bmatrix}, \tag{40}$$

$$\widehat{C}^T = C^T \overline{\Delta}^T, \qquad \breve{C}_i^T = \begin{bmatrix} M_p^i \overline{V}_i^T \\ 0 \end{bmatrix}.$$

Then (38) can be rewritten as

$$\Sigma + \Sigma_1 F(k) \Sigma_2 + \Sigma_2^T F^T(k) \Sigma_1^T + \Sigma_3 F(k) \Sigma_4 + \Sigma_4^T F^T(k) \Sigma_3^T < 0, \tag{41}$$

where

$$\Sigma = \begin{bmatrix} \Xi_{11} & \overline{\Xi}_{12}^T & \overline{\Xi}_{13}^T \overline{\Xi}_{14} \\ * & \operatorname{diag}\{I, I\} & 0 \\ * & * & \overline{\Xi}_{22} \end{bmatrix},$$

$$\overline{\Xi}_{13}^T = \begin{bmatrix} 0 & \sqrt{1-\overline{\alpha}}\,\underline{A}_i^T & \sqrt{\overline{\alpha}}\,\underline{A}_i^T \\ 0 & 0 & \sqrt{\overline{\alpha}}\,\overline{N}_1^T \\ \sqrt{\overline{\beta}\left(1-\overline{\beta}\right)}\,\overline{B}_2^T & \sqrt{1-\overline{\alpha}}\left(\overline{B}_2^T + \overline{\beta}\overline{B}_2^T\right) & \sqrt{\overline{\alpha}}\left(\overline{B}_2^T + \overline{\beta}\overline{B}_2^T\right) \end{bmatrix},$$

$$\overline{\Xi}_{12}^T = \begin{bmatrix} \sqrt{\overline{\beta}}\,\underline{C}_i^T & \sqrt{1-\overline{\beta}}\,\underline{C}_i^T \\ 0 & 0 \\ \sqrt{\overline{\beta}}\,\overline{D}^T & 0 \end{bmatrix}, \qquad \Sigma_1^T = \left[\widehat{A}, \underbrace{0, \ldots, 0}_{7}\right]^T, \qquad \Sigma_2^T = \left[\underbrace{0, \ldots, 0}_{6}, \breve{A}_i^T, \breve{A}_i^T\right],$$

$$\Sigma_3^T = \left[\widehat{C}, \underbrace{0, \ldots, 0}_{7}\right]^T, \qquad \Sigma_4^T = \left[0, 0, 0, \breve{C}_i^T, \breve{C}_i^T, 0, 0, 0\right]. \tag{42}$$

From Schur Complement, for any scalars $\varepsilon_i > 0$ $(i = 2, 3)$, we get

$$\begin{bmatrix} \Sigma & \varepsilon_2 \Sigma_1^T & \Sigma_2 & \varepsilon_3 \Sigma_3^T & \Sigma_4 \\ * & -\varepsilon_2 I & 0 & 0 & 0 \\ * & * & -\varepsilon_2 I & 0 & 0 \\ * & * & * & -\varepsilon_3 I & 0 \\ * & * & * & * & -\varepsilon_3 I \end{bmatrix} < 0. \tag{43}$$

Now, partition P_i as $P_i = \begin{bmatrix} P_{i1} & P_{i2} \\ * & P_{i3} \end{bmatrix}$; then define matrics $\Upsilon_j \triangleq \sum_{j \in \mathfrak{R}_K^i} \sigma_{ij} P_j = \sum_{j \in \mathfrak{R}_K^i} \sigma_{ij} \begin{bmatrix} P_{j1} & P_{j2} \\ * & P_{j3} \end{bmatrix} \triangleq \begin{bmatrix} \Upsilon_{j1} & \Upsilon_{j2} \\ * & \Upsilon_{j3} \end{bmatrix}$. If we replace $\overline{L}_i = Y_i L_i$ and $\overline{V}_i = V_i$ into (43), we readily obtain (34). From Theorem 2, augmented error systems (13) will be finite-time stochastically stable with H_∞ performance level (15). Meanwhile, if the solution of (34) exists, the admissible finite-time fault detection filters are given by (36). The proof is completed. □

From a viewpoint of computation, it should be noted that the conditions in Theorem 4 are still not standard linear matrix inequalities (LMIs) conditions due to (18). Actually, conditions (18) can also be guaranteed by LMIs conditions once the values of $\overline{\sigma}$ is set. For given positive scalar λ, it is easy to check that condition (18) is guaranteed by imposing condition $\vartheta \le \underline{\lambda} = \min_{i \in \mathfrak{R}} \lambda_{\min}(\widetilde{P}_i) < 1$ and

$$\vartheta R \le P_i \le R, \tag{44}$$

$$\frac{c_1^2}{\vartheta} < \frac{c_2^2}{\overline{\sigma}^N}. \tag{45}$$

Then, inequality (18) can be converted into the following LMI by using Schur Complement:

$$\begin{bmatrix} -\frac{c_2^2}{\overline{\sigma}^N} & c_1 \\ * & -\vartheta \end{bmatrix} < 0. \tag{46}$$

Thus, once $\overline{\sigma}$ is fixed, the feasibility of (18) in Theorem 4 can be translated into LMI-based conditions (44) and (46). Theorem 4 can be solved by Matlab's LMI toolbox [21].

Remark 5. As the special cases of partly unknown transition probabilities, when all the transition probabilities are completely accessible ($\mathfrak{R}_K^i = \mathfrak{R}, \mathfrak{R}_{UK}^i = \Phi$) and completely inaccessible ($\mathfrak{R}_{UK}^i = \mathfrak{R}, \mathfrak{R}_K^i = \Phi$), the underlying systems are the traditional Markovian jump systems and the switched systems under arbitrary switching, respectively. Correspondingly, the fault detection results can be found in some existing references, [10] investigated linear discrete-time Markovian jump systems (completely accessible), [11] studied linear discrete-time switched systems (completely inaccessible), [22] considered transition probabilities with polytopic uncertainties which require the knowledge of uncertainties structure and it can still be viewed as accessible. Therefore, finite-time fault detection with partly unknown transition probabilities is a more natural assumption to the Markovian jump systems and hence covers the existing ones. Furthermore, when $\overline{\alpha} = 1$ and $\overline{\beta} = 1$, then augmented error systems (13) with RON and ROF are the usually nonlinear system [19] and fault detection system [10–17]; from this view, (13) is also a more comprehensive networked systems model.

4. Numerical Example and Simulation

Consider the nonlinear networked systems (1) with the following parameters:

$$\begin{aligned} A &= \begin{bmatrix} -0.2 & 0.1 \\ -0.2 & 0.3 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0.7 \\ 0.2 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ -0.3 \end{bmatrix}, \\ C &= \begin{bmatrix} 0.2 & 0.5 \\ 0.1 & -0.3 \end{bmatrix}, \quad N_1 = \begin{bmatrix} 0 & 0.3 \\ 0.1 & -0.5 \end{bmatrix}, \\ G &= \begin{bmatrix} -0.1 & 0.3 \\ 0 & 0.2 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad N = 20. \end{aligned} \tag{47}$$

Attention is focused on the design of mode-dependent observer-based finite-time fault detection filters, which make the augmented error systems (13) finite-time stochastically stable with H_∞ performance level (15). If we consider two channels data transmission networked systems, according to the transmission pattern presented in Section 2, the transmission matrices are constructed as

$$\begin{aligned} M_2^{2,1} &\triangleq \operatorname{diag}\{1,1\}, \quad M_2^{0,1} \triangleq \operatorname{diag}\{0,0\}, \\ M_2^{1,1} &\triangleq \operatorname{diag}\{1,0\}, \quad M_2^{1,2} \triangleq \operatorname{diag}\{0,1\}. \end{aligned} \tag{48}$$

With above data transmission pattern, from Remark 5, we consider the Case II with partly unknown transition probabilities in Table 1, where "?" means that element is unknown. As the special cases of Case II, corresponding results of traditional Markovian jump systems (Case I) and switched systems (Case III) can be included in our theorems. By Theorem 4, for the given $c_1 = 0.6$, $\overline{\alpha} = 0.5$, and $\delta = 0.4$, the suboptimal finite-time fault detection performance level γ^* is obtained in Table 2. From Table 2, it can be easily seen that finite-time fault detection performance level γ^* is dependent on ROFs probability $\overline{\beta}$ and RONs probability $\overline{\alpha}$, finite-time stability index $\overline{\sigma}$, quantization level χ_i, and the information of transition probability matrices, which show the effectiveness of our discussion.

Assuming that the parameters are given by $u_0^1 = u_0^2 = 1$, $\chi_1 = \chi_2 = 0.8$, $\overline{\beta} = 0.6$, $\overline{\sigma} = 1.3$, $c_1 = 0.6$, $\overline{\alpha} = 0.5$, $\delta = 0.4$, and $\gamma = 1.1$, by applying (34), (36), (44), and (46) of Theorem 4, mode-dependent observer-based finite-time fault detection filter matrices L_i^j and V_i^j ($i = 1,2,3,4$; $j = 1,2,3$) can be obtained as follows:

$$\begin{aligned}
L_1^1 &= \begin{bmatrix} -0.0458 & -0.0346 \\ 0.0889 & -0.1042 \end{bmatrix}, \quad V_1^1 = \begin{bmatrix} -0.0167 & 0.0385 \end{bmatrix}, \\
L_2^1 &= \begin{bmatrix} -1.0599 & -5.6530 \\ 1.7666 & 4.2394 \end{bmatrix}, \quad V_2^1 = \begin{bmatrix} -0.0431 & -0.2377 \end{bmatrix}, \\
L_3^1 &= \begin{bmatrix} -0.0714 & -0.2247 \\ 0.0941 & -0.4006 \end{bmatrix}, \quad V_3^1 = \begin{bmatrix} -0.0369 & -0.1734 \end{bmatrix}, \\
L_4^1 &= \begin{bmatrix} -0.0078 & -0.0235 \\ 0.1124 & -0.1258 \end{bmatrix}, \quad V_4^1 = \begin{bmatrix} 0.0479 & 0.0529 \end{bmatrix}, \\
L_1^2 &= \begin{bmatrix} -0.0658 & -0.0699 \\ 0.1156 & -0.1768 \end{bmatrix}, \quad V_1^2 = \begin{bmatrix} -0.0155 & 0.0426 \end{bmatrix}, \\
L_2^2 &= \begin{bmatrix} -0.0761 & -1.3612 \\ 0.9768 & 1.9094 \end{bmatrix}, \quad V_2^2 = \begin{bmatrix} -0.0260 & -0.1211 \end{bmatrix}, \\
L_3^2 &= \begin{bmatrix} -0.2241 & -0.6533 \\ 0.1698 & -0.7305 \end{bmatrix}, \quad V_3^2 = \begin{bmatrix} -0.0321 & -0.1185 \end{bmatrix}, \\
L_4^2 &= \begin{bmatrix} 0.0272 & -0.0135 \\ 0.1279 & -0.2607 \end{bmatrix}, \quad V_4^2 = \begin{bmatrix} 0.0679 & 0.0757 \end{bmatrix}, \\
L_1^3 &= \begin{bmatrix} -0.0317 & -0.0356 \\ 0.0835 & -0.0987 \end{bmatrix}, \quad V_1^3 = \begin{bmatrix} -0.0098 & 0.0267 \end{bmatrix}, \\
L_2^3 &= \begin{bmatrix} 0.2694 & 5.7487 \\ -1.1041 & -3.3997 \end{bmatrix}, \quad V_2^3 = \begin{bmatrix} -0.0443 & -0.2361 \end{bmatrix}, \\
L_3^3 &= \begin{bmatrix} -0.0561 & -0.2962 \\ 0.0792 & -0.2882 \end{bmatrix}, \quad V_3^3 = \begin{bmatrix} -0.0375 & -0.1713 \end{bmatrix}, \\
L_4^3 &= \begin{bmatrix} -0.0143 & -0.0284 \\ 0.1083 & -0.1125 \end{bmatrix}, \quad V_4^3 = \begin{bmatrix} 0.0541 & 0.0435 \end{bmatrix}.
\end{aligned} \tag{49}$$

To demonstrate the effectiveness of designed finite-time fault detection filter, for $k = 0, 1, \ldots, 20$, unknown disturbance input $d(k)$ is assumed to be band-limited white noise

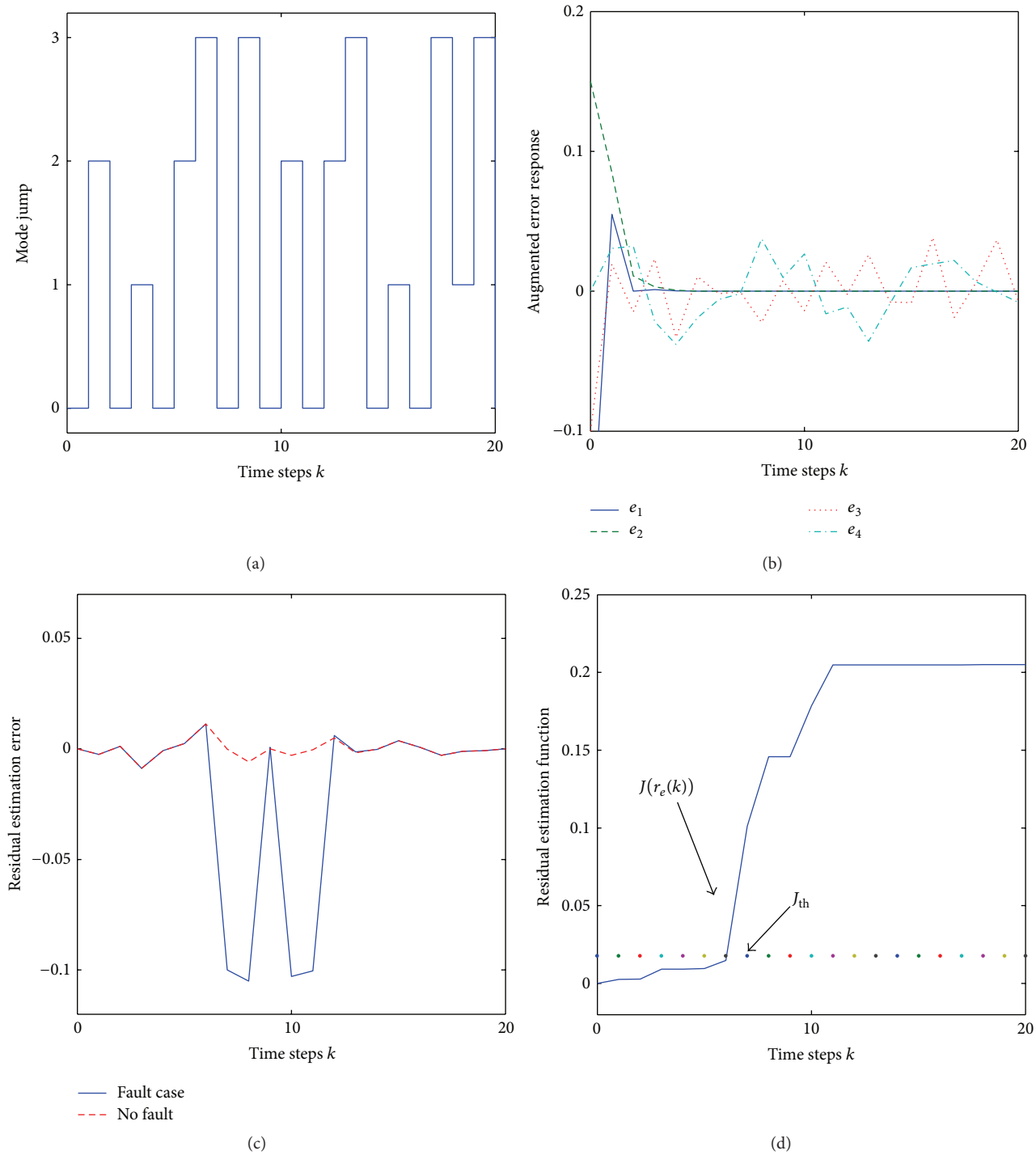


Figure 1: Corresponding simulation of Case I.

with power of 0.05, and the fault signal $f(k)$ is simulated as a square wave of 0.1 amplitude that occurred from 8 to 14 steps and the nonlinear function is given by $g(x(k)) = \sin(0.1 \times x(k))$. Under Cases I, II, and III, the initial state of augmented error systems (13) is assumed as $e(0) = [-0.2 \;\; 0.15 \;\; -0.1 \;\; 0]^T$, corresponding evolution of residual estimation error signal $r_e(k)$ and residual evaluation function $J(r_e)$ are shown in Figures 1, 2, and 3, respectively. For given $k_0 = 0$ and $\kappa = 20$, the threshold J_{th} can be determined by utilizing 300 Monte Carlo simulations in Table 3; from Table 3, we observe that, when $k = 8$, $J(r_e) \geq J_{\text{th}}$, for the first time, which means that the fault $f(k)$ can be detected

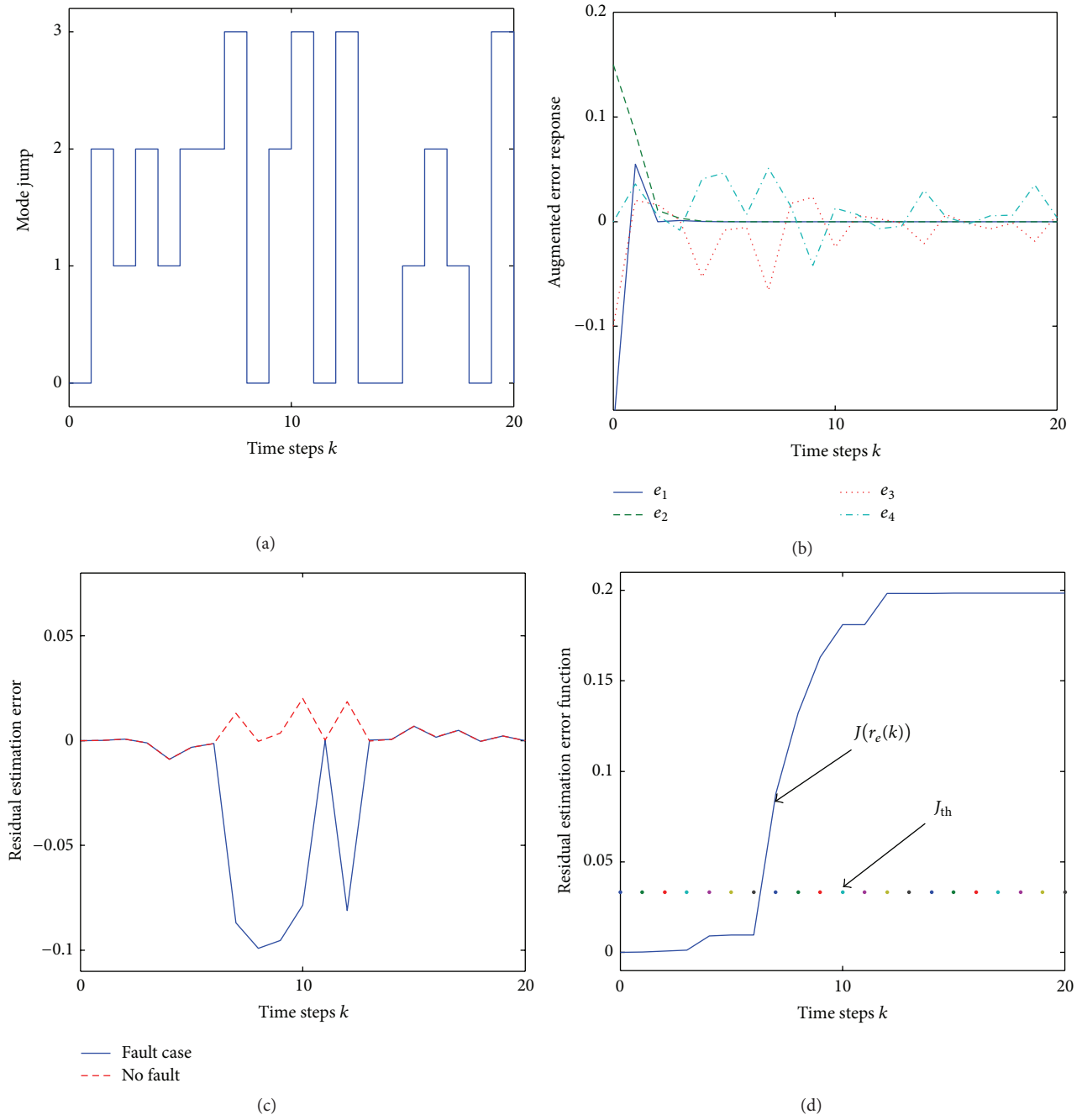


Figure 2: Corresponding simulation of Case II.

Table 1: Different transition probabilities matrices cases.

Completely known (Case I)	Partly unknown (Case II)	Completely unknown (Case III)
$\begin{bmatrix} 0.1 & 0.2 & 0.3 & 0.4 \\ 0.1 & 0.6 & 0.1 & 0.2 \\ 0.5 & 0.1 & 0.2 & 0.2 \\ 0.6 & 0.1 & 0.2 & 0.1 \end{bmatrix}$	$\begin{bmatrix} 0.1 & 0.2 & 0.3 & 0.4 \\ ? & 0.6 & ? & 0.2 \\ 0.5 & ? & ? & ? \\ 0.6 & 0.1 & 0.2 & 0.1 \end{bmatrix}$	$\begin{bmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{bmatrix}$

as soon as its occurrence, respectively, so the effectiveness of proposed finite time fault detection problem is illustrated.

5. Conclusion

This paper is concerned with the problem of finite-time fault detection for large-scale networked systems. A Markovian jump systems model with partly unknown transition

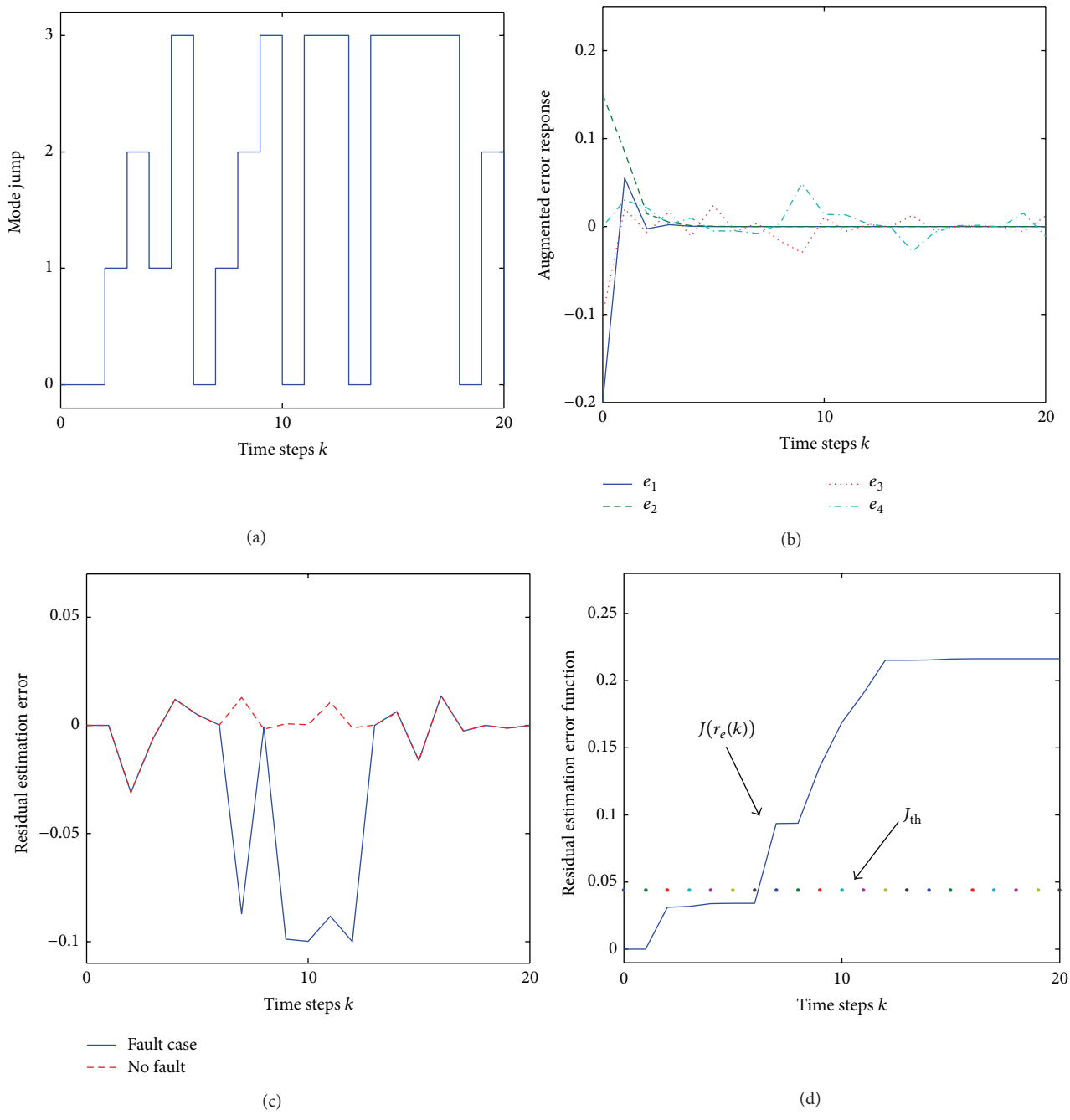


Figure 3: Corresponding simulation of Case III.

Table 2: Corresponding minimum finite-time H_∞ attenuation level γ^* for different cases.

	$\overline{\sigma} = 1.1$, $\overline{\beta} = 0.4$ $\chi_1 = \chi_2 = 0.3$	$\overline{\sigma} = 1.5$, $\overline{\beta} = 0.4$ $\chi_1 = \chi_2 = 0.3$	$\overline{\sigma} = 1.5$, $\overline{\beta} = 0.7$ $\chi_1 = \chi_2 = 0.3$	$\overline{\sigma} = 1.5$, $\overline{\beta} = 0.7$ $\chi_1 = \chi_2 = 0.8$
Completely known	1.0001	1.0022	1.0102	1.0103
Partly unknown	1.0001	1.0083	1.0146	1.0145
Completely unknown	1.0001	1.0113	1.0895	1.0895

Table 3: Corresponding threshold and residual evolution function value for different cases.

	Completely known	Partly unknown	Completely unknown
Threshold	$J_{th} = 0.0177$	$J_{th} = 0.0333$	$J_{th} = 0.0442$
Residual evolution Error function	$0.0148 = J(7) < J_{th} < J(8) = 0.1012$	$0.0096 = J(7) < J_{th} < J(8) = 0.0875$	$0.0343 = J(7) < J_{th} < J(8) = 0.0937$

probabilities is introduced to describe multiple channels data transmission pattern, while the cases with completely known or completely unknown transition probabilities have been investigated as its special cases. The randomly occurring nonlinearities and randomly occurring faults are also introduced to reflect the limited capacity of the communication network resulting from the noisy environment and probabilistic communication failures. Based on this, more natural model, finite-time fault detection of nonlinear large-scale networked systems, is formulated as nonlinear finite-time H_∞ attention problem. The main objective is to design mode-dependent observer-based finite-time fault detection filter such that the error between residual signal and fault signal is made as small as possible. Simulations are given to illustrate the effectiveness of proposed design techniques.

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 editor and the reviewers for their helpful suggestions to improve the quality of this correspondence. This work is supported by e National Natural Science Foundation of China by Grant nos. 61104027, 61174107, and 61034006.

References

[1] W. Zhang, M. S. Branicky, and S. M. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84–99, 2001.

[2] J. P. Hespanha, P. Naghshtabrizi, and Y. G. Xu, "A survey of recent results in networked control systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 138–162, 2007.

[3] G.-P. Liu, Y. Q. Xia, R. D. Rees, and W. S. Hu, "Design and stability criteria of networked predictive control systems with random network delay in the feedback channel," *IEEE Transactions on Systems, Man and Cybernetics C: Applications and Reviews*, vol. 37, no. 2, pp. 173–184, 2007.

[4] D. Yue, E. G. Tian, Z. D. Wang, and J. Lam, "Stabilization of systems with probabilistic interval input delays and its applications to networked control systems," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 39, no. 4, pp. 939–945, 2009.

[5] D. Ye, Y. Long, and G.-H. Yang, "Finite-frequency filter design for networked control systems with missing measurements," *Mathematical Problems in Engineering*, vol. 2013, Article ID 825143, 8 pages, 2013.

[6] L. Zhang and D. Hristu-Varsakelis, "Communication and control co-design for networked control systems," *Automatica*, vol. 42, no. 6, pp. 953–958, 2006.

[7] D.-S. Kim, D.-H. Choi, and P. Mohapatra, "Real-time scheduling method for networked discrete control systems," *Control Engineering Practice*, vol. 17, no. 5, pp. 564–570, 2009.

[8] J. N. Li, L. F. Wei, C. Liu, and H. Y. Yu, "H_∞ control of network-based systems with packet disordering and packet loss compensation," *Mathematical Problems in Engineering*, vol. 2013, Article ID 836314, 9 pages, 2013.

[9] S. Hu and W.-Y. Yan, "Stability of networked control systems under a multiple-packet transmission policy," *IEEE Transactions on Automatic Control*, vol. 53, no. 7, pp. 1706–1711, 2008.

[10] M. Y. Zhong, S. X. Ding, J. Lam, and H. Wang, "Fault detection for Markovian jump systems," *IEE Proceedings—Control Theory and Application*, vol. 152, no. 4, pp. 397–402, 2005.

[11] D. Wang, W. Wang, and P. Shi, "Robust fault detection for switched linear systems with state delays," *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 39, no. 3, pp. 800–805, 2009.

[12] Y. Yin, P. Shi, and F. Liu, "Gain-scheduled robust fault detection on time-delay stochastic nonlinear systems," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 10, pp. 4908–4916, 2011.

[13] H. J. Fang, H. Ye, and M. Y. Zhong, "Fault diagnosis of networked control systems," *Annual Reviews in Control*, vol. 31, no. 1, pp. 55–68, 2007.

[14] H. L. Dong, Z. D. Wang, J. Lam, and H. J. Gao, "Fuzzy-model-based robust fault detection with stochastic mixed time delays and successive packet dropouts," *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 42, no. 2, pp. 365–376, 2012.

[15] Z. Mao, B. Jiang, and P. Shi, "Protocol and fault detection design for nonlinear networked control systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 56, no. 3, pp. 255–259, 2009.

[16] X. He, Z. Wang, and D. H. Zhou, "Robust fault detection for networked systems with communication delay and data missing," *Automatica*, vol. 45, no. 11, pp. 2634–2639, 2009.

[17] Y. Zhang and H. J. Fang, "Fault detection for nonlinear networked systems with random packet dropout and probabilistic interval delay," *International Journal of Adaptive Control and Signal Processing*, vol. 25, no. 12, pp. 1074–1086, 2011.

[18] F. Amato, M. Ariola, and P. Dorato, "Finite-time control of linear systems subject to parametric uncertainties and disturbances," *Automatica*, vol. 37, no. 9, pp. 1459–1463, 2001.

[19] Z. D. Wang, Y. Wang, and Y. Liu, "Global synchronization for discrete-time stochastic complex networks with randomly occurred nonlinearities and mixed time delays," *IEEE Transactions on Neural Networks*, vol. 21, no. 1, pp. 11–25, 2010.

[20] M. Fu and L. Xie, "The sector bound approach to quantized feedback control," *IEEE Transactions on Automatic Control*, vol. 50, no. 11, pp. 1698–1711, 2005.

[21] S. Boyd, L. el Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, SIAM, Philadelphia, Pa, USA, 1994.

[22] E. Carlos, C. E. de Souza, A. Trofino, and K. A. Barbosa, "Mode-independent H_∞ filters for Markovian jump linear systems," *IEEE Transactions on Automatic Control*, vol. 51, no. 11, pp. 1837–1841, 2006.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 578713, 6 pages
http://dx.doi.org/10.1155/2014/578713

Research Article

A Comparison of Online Social Networks and Real-Life Social Networks: A Study of Sina Microblogging

Dayong Zhang and Guang Guo

Department of New Media and Arts, Harbin Institute of Technology, Harbin 150001, China

Correspondence should be addressed to Dayong Zhang; zhdy@hit.edu.cn

Received 10 November 2013; Revised 17 February 2014; Accepted 21 March 2014; Published 22 April 2014

Academic Editor: Guoqiang Hu

Online social networks appear to enrich our social life, which raises the question whether they remove cognitive constraints on human communication and improve human social capabilities. In this paper, we analyze the users' following and followed relationships based on the data of Sina Microblogging and reveal several structural properties of Sina Microblogging. Compared with real-life social networks, our results confirm some similar features. However, Sina Microblogging also shows its own specialties, such as hierarchical structure and degree disassortativity, which all mark a deviation from real-life social networks. The low cost of the online network forms a broader perspective, and the one-way link relationships make it easy to spread information, but the online social network does not make too much difference in the creation of strong interpersonal relationships. Finally, we describe the mechanisms for the formation of these characteristics and discuss the implications of these structural properties for the real-life social networks.

1. Introduction

In the past decade, the online social network has made new opportunities for communication and has revolutionized many aspects of our lives. As a new kind of online social networks, the Microblogging, such as Twitter and Sina Microblogging, has been developing rapidly in recent years. The online social network provides people with a public network platform, meets needs of computer-mediated communications, and rebuilds social connections. The structure and evolution of online social network have attracted the attention of researchers in different disciplines, including sociology, physics, and computer information technology [1, 2]. But large-scale and complex structure of online social network brings many difficulties in complete topological analysis. However, the emergence of complex network research [3–5] provides us with an effective method to study the online social network topology and information dissemination characteristics [6–11].

As for the mapping and expansion of the actual social networks on the Internet, online social networks have broken the spatiotemporal limitations and decreased communication cost. Will the online social networks change the rules of actual social networks? The following studies reach fundamentally different conclusions: Backstrom and Boldi [12] studied the largest online social network that has ever been created (721 million active Facebook users and their 69 billion friendship links). The analyses of data proved that the average distance of Facebook has shortened from 5.28 in 2008 to 4.74 in 2011. When the search scale was narrowed to a country, it showed that most of the people were in fact only four apart. The result presented the "shrinking diameter" phenomena [13]. It suggested that the online social network made the relationship among people come closer. Six degrees of separation is inapplicable to the online social network. However, the statistics of Facebook's official website showed that the average number of Facebook active users' friends was 130 (http://www.facebook.com/press/info.php?statistics), which agrees well with Dunbar's number. Similarly, Gonçalves et al. [14] analyzed a dataset of Twitter conversations collected across six months involving 1.7 million individuals and found that users kept stable relationships with 100–200 users. The data is in agreement with Dunbar's result too. Thus, the

Table 1: Summary of degree distribution.

k	Number of nodes	Proportion	*Extremum*
$1 \le k < 10$	783	93.3%	$k_{\min} = 1$
$10 \le k < 100$	50	6.00%	
$k \ge 100$	6	0.07%	$k_{\max} = 327$

"economy of attention" is limited in the online world by cognitive and biological constraints as predicted by Dunbar's theory [15].

Has online social network changed our real-life social pattern? Why do such discrepancies exist between the studies? The differences are due to the different analytical content and analysis approach. The low cost of the online network forms a broader perspective, and the one-way link relationships make it easy to spread information, but the online social networks do not make too much difference in the creation of strong interpersonal relationships. These online social relationships, which may reflect the real-life social networks, create an unprecedented field to understand the characteristics of human networks [16, 17].

In this paper, we study the topological characteristics of Sina Microblogging and try to explain its characteristics formation mechanism by comparing it with the real-life social networks. This paper is organized as follows. Section 2 describes our data collection on Sina Microblogging. Then, we conduct topological analysis of the Sina Microblogging network and analyze the mechanisms for the formation of the structure in Section 3. In Section 4, we study the mixing patterns on the Sina Microblogging network. In Section 5, we focus on the analysis of node importance by betweenness centrality. In Section 6, we conclude.

2. Sina Microblogging Data Collection

In August 2009, Sina Microblogging began its trial service and became the most popular Microblogging service in China, with more than 500 million users as of 2013. Sina blog and Sina news provide good capital bases and natural advantages for the success of Sina Microblogging. Due to these advantages, Sina Microblogging developed the characteristics of We Media that spread news faster than any other media. In order to study the structure of Sina Microblogging, we develop a spider with Python for Sina Microblogging. The snowball [18] crawl algorithm has been applied in the spider. We collected profiles of 3441 users on Sina Microblogging until November 11, 2011. Isolated points are excluded, so that filter leaves only 839 nodes. In order to protect privacy, we only keep the users' ID number in the original data information about the users. Based on the "following" and "followed," we construct an undirected network which has 2112 links and analyze its basic characteristics.

3. Analysis of Network Structure

We begin our analysis of Sina Microblogging from the following aspects. In Section 3.1, first, we calculate the degree distribution. Then, we evaluate the average path length and clustering coefficient and study the reason why Sina Microblogging possesses obviously small-world property in Section 3.2.

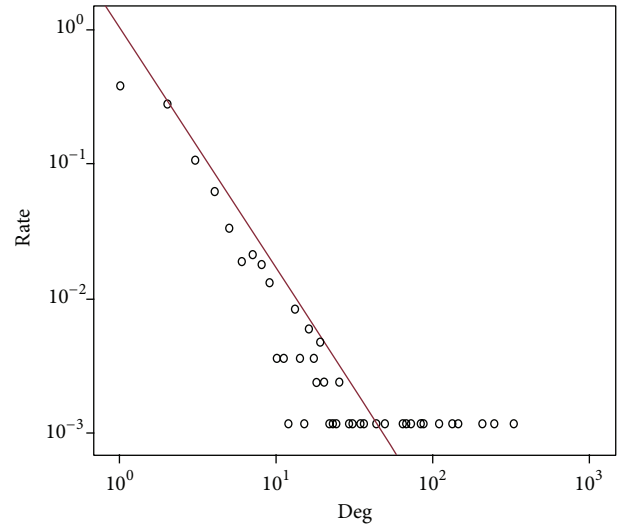


Figure 1: Degree distribution for nodes in Sina Microblogging.

3.1. Power-Law Node Degrees. In Table 1, we summarize the degree distribution of Sina Microblogging. We find that only a few nodes of the network have a high degree. The degree distribution is very unevenly distributed.

We begin the analysis of Sina Microblogging topology by looking at its degree distributions. Networks of a power-law degree distribution, $(k) \propto k^{-\gamma}$, where k is the node degree and $\gamma \le 3$, attest to the existence of a relatively small number of nodes with a very large number of links. Many social networks satisfy power-law degree distribution [19, 20, 23], and a few networks obey stretched exponential distribution [24] which is defined as $p(k) = be^{-(k/k_0)^c}$.

The degree distribution of Sina Microblogging is shown as in Figure 1. The y-axis represents the frequency of degree. It satisfies a power-law distribution with an exponent of 1.75; the goodness of fit is 0.712.

The networks which meet a power-law distribution have scale-free property, and such networks are called scale-free networks. Therefore, Sina Microblogging is a scale-free network. The real-life social networks are usually scale-free networks. The scale-free network is caused by the growing and preferential mechanisms that the new nodes trend to connect with hub nodes, and this phenomenon is called the Matthew effect.

3.2. The Small-World Property. The study of "small-world" networks has become a key to understanding the societal structure, ever since Stanley Milgram's famous "six degrees of separation" experiment. In his work, he reports that any two people could be connected on average within six hops from each other. Watts and Strogatz [3] have revealed two important characteristics of "small-world" networks:

Table 2: Small-World property compared with different social networks.

Social network	Average path length	Clustering coefficient
Sina Microblogging	2.794	0.374
Random network	4.162	0.006
Cyworld [19]	3.2	0.16
Renren [20]	3.48	0.2
Mixi [21]	5.53	0.1215
Co-author network [22]	4.6	0.066

(1) the "small-world" networks have small characteristic path lengths, like random graphs; (2) the "small-world" networks can be highly clustered, like regular lattices. The network measurement studies have shown that the real networks are mostly small-world, especially social networks [13, 21].

The concept of average path length for undirected networks is well known; the measures are related by

$$L = \frac{1}{(1/2)\,N\,(N+1)} \sum_{i \geq j} d_{ij}, \tag{1}$$

where i, j are two nodes, N is the number of nodes in the network, and d_{ij} denote the shortest distance between i and j.

The average path length is closely connected to the characteristics of the network, such as connectivity, reachability, and transferring latency. A short average path length facilitates the quick transfer of information and reduces costs.

Another property of "small-world" networks is the clustering coefficient. It is the measure of the extent to which one's friends are also friends of each other. The local clustering coefficient for a node is then given by the proportion of links between the nodes within its neighborhood divided by the number of links that could possibly exist between them. The local clustering coefficient for undirected graphs can be defined as

$$C_i = \frac{2E_i}{k_i\,(k_i - 1)}, \tag{2}$$

where k_i is the degree of node i and E_i is the number of edges between neighbors of node i.

The clustering coefficient for the whole network is given by Watts and Strogatz [3] as the average of the local clustering coefficients of all the nodes; the measures are related by

$$C = \frac{1}{N} \sum_{i}^{N} C_i. \tag{3}$$

The networks with the largest possible average clustering coefficient are found to have a modular structure.

The results concerning average path length and clustering coefficient are displayed in Table 2. Compared with the same scale random network, the average path length of Sina Microblogging is shorter than random network (the average path length of random network is defined as ($L \propto \ln N / \ln k$), and the clustering coefficient is much greater than the random network (the clustering coefficient of random network is defined as = $\langle k \rangle / N$). Thus, we confirm that Sina Microblogging has the small-world property.

Renren, Cyworld, and Mixi are the largest and oldest online social networking services in China, South Korea, and Japan, respectively. Coauthor network is a kind of real-life social network. By contrast, the average path length of Sina Microblogging has a shorter average path length and a greater clustering coefficient. It is revealed that Sina Microblogging's small-world property is the most obvious. The striking small-world phenomenon indicates that there are rich local connections in Sina Microblogging network, the nodes are linked closely, and information dissemination is more efficient.

We analyze the mechanisms for the formation of small-world property from the following two aspects. First is the user's behavior aspect. In 2011, for many users, the main reasons they use social network out there are the latest developments of friends (73.4%), keeping in touch with old friends (73.1%), and documentation of life and feeling (67.5%), and the main purposes of Microblogging users are getting information (58.1%), following celebrities (57.6%), discussing the hot topics, and personal experience (52.3%), according to the data of "the user behavior research of SNS and microblogging in China." The Microblogging users tend to use the Microblogging to record personal feelings, share the news, and find groups with similar interests, and the traditional SNS users show themselves, contact with friends in real life, and expand social circle on SNS. Consequently, the former are oriented to information exchange, but the latter stress interpersonal communication.

Second is the aspect of users' link mode. The main difference between the Renren, Cyworld, and Mixi networks and Microblogging is the directed nature of Microblogging relationship. In Renren, Cyworld, and Mixi, a link represents a mutual agreement of a relationship, while on Sina Microblogging a user is not obligated to reciprocate followers by following them. Thus, a path from a user to another may follow different hops or not exist in the reverse direction. Both the internal links (strong ties) and one-way following links (weak ties) are in Microblogging. And the Microblogging's distinction between different types of interactions allows us to get more information: personal interactions are more likely to occur on internal links (strong ties) and events transmitting new information rely more on one-way following links (weak ties) [25].

In conclusion, Microblogging is easier to form the user centric We Media than traditional SNS; therefore, it possesses small-world property that can facilitate the flow of information.

In order to gauge the correlation between clustering coefficient and the degree, we plot the clustering coefficient of nodes (y) against the degree of nodes (x) in Figure 2 and bin the clustering coefficient in log scale. If the correlation between clustering coefficient and the degree accords with $C(k) = k^{-a}$, we can consider that the network is of apparent hierarchical structure. We observe that the clustering coefficient varies inversely as the degree. It satisfies a power-law

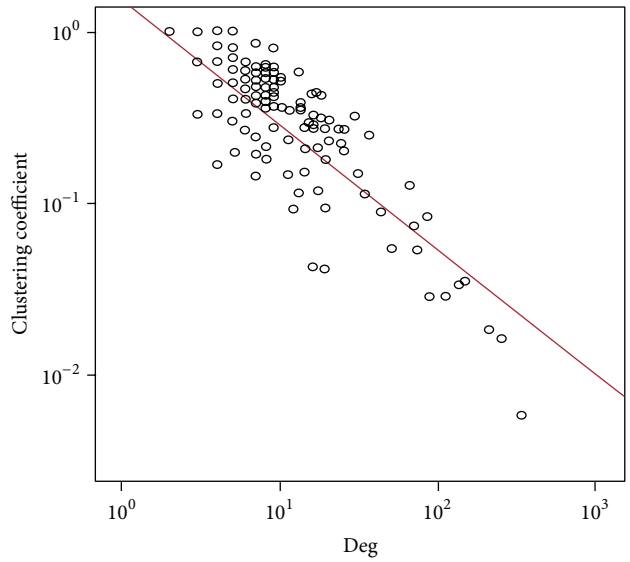


Figure 2: Correlations between clustering coefficients and degrees.

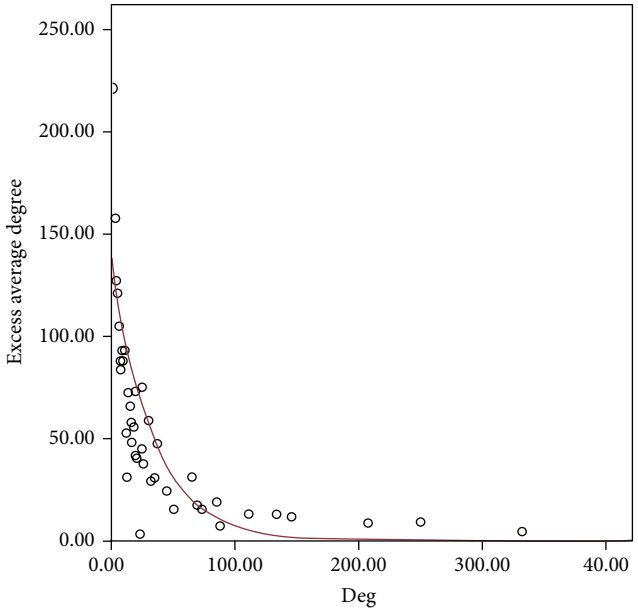


Figure 3: Correlations between excess average degrees and degrees.

distribution with an exponent of 0.733, and the goodness of fit is 0.712. Thus, the Sina Microblogging has apparent hierarchical structure, where vertices divide into groups that further subdivide into groups of groups and so forth over multiple scales.

The structure is caused by Sina Microblogging's fans mechanism. Basing on the user interest and the celebrity effect, Sina Microblogging builds many kinds of fans groups. Celebrities and professionals in a certain area who have more resources and influence will get more attention, and these users only keep internal links with peers or slightly lower level users and so forth until the hierarchical structure is formed.

4. The Mixing Patterns

Mixing patterns refer to systematic tendencies of one type of nodes in a network to connect to another type. There are three types of mixing patterns: assortative network, disassortative network, and neutral network. Similar vertices tend to connect to each other in assortativity network, and nodes of low degree are more likely to connect with nodes of high degree in disassortative network. A lot of empirical studies [26, 27] had revealed that the real-life social networks trend to assortativity, opposite of the online social networks. In real-life social networks, the ordinary people want to get along with the celebrity, while the celebrity tends to make the acquaintance of peers; therefore, the ordinary people have less opportunity to integrate into the circles of the celebrity. In contrast, the ordinary people can easily get connected with the celebrity; the celebrities are also willing to show their influence by the number of fans in the online social network. Thus, the online social networks trend to be disassortative networks.

We cannot understand the mixing pattern intuitively, so we introduce the concept of excess average degree. The excess degree is the number of edges leaving the vertex other than the one we arrived along. This number is one less than the degrees of the vertices themselves. The excess average degree is defined as

$$\langle k_{nn} \rangle (k) = \frac{1}{q_k} \sum_{k=k_{\min}}^{k_{\max}} k' e_{kk}. \quad (4)$$

The excess average degree is positive in assortative network and negative otherwise. Figure 3 plots the curve of the excess average degree (y) against the degree (x) as the red line. We see significant negative correlation. Hence, the Sina Microblogging is a disassortative network.

To better understand the meaning of disassortative network, we illustrate the network connections of the node ID 975 in Figure 4. In connected component, different colors represent different communities. We find that the node ID 975 connects with three hubs of community and shows disassortativity.

5. Analysis of Node Importance

Social networks are discrete systems with a large amount of heterogeneity among nodes. Measures of centrality direct at a quantification of nodes' importance for structure and function. The most direct measure of centrality is the degree centrality; that is, the node with greater degree is the most important one. In addition, betweenness centrality is also a measure of node's centrality in a network and can be used to measure the influence a node has over the spread of information through the network. It is equal to the number of the shortest paths from all vertices to all others that pass through that node.

In Figure 5, we plot the correlation between node betweenness centralities and degrees. Then, we compute that the average of node betweenness centralities is 33.62 and the correlation coefficient is 0.96, indicating that there are

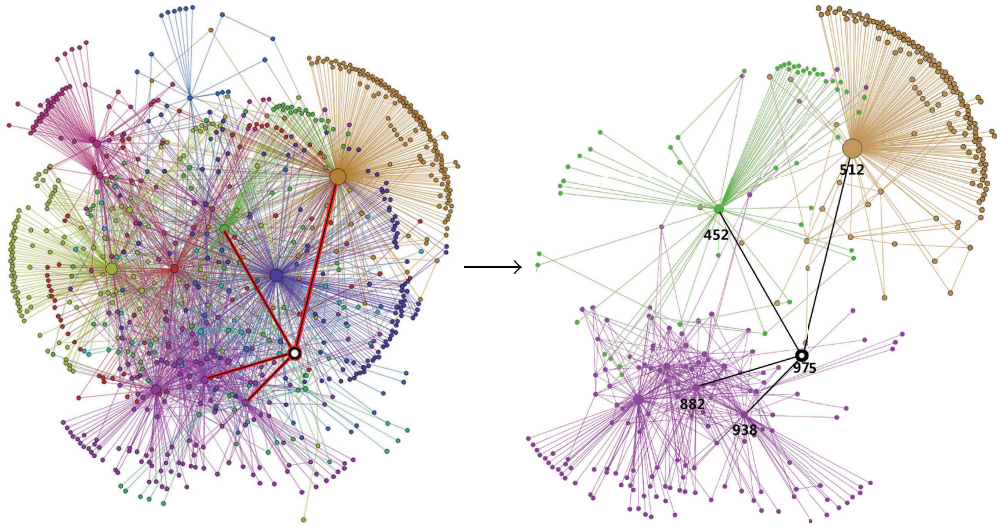


Figure 4: The network connections of the node ID 975.

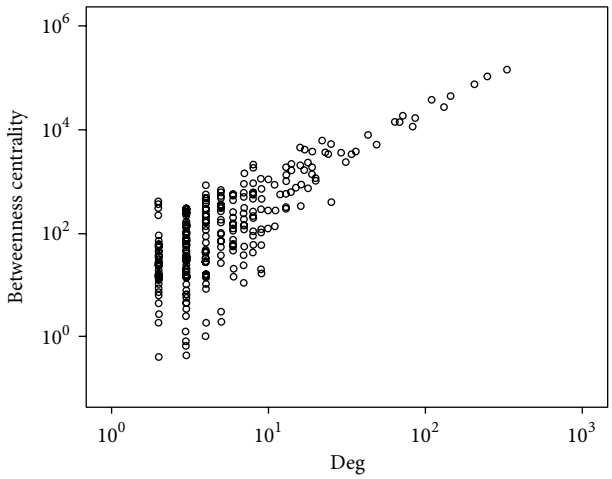


Figure 5: Correlations between node betweenness centralities and degrees.

strong and positive correlations between node betweenness centralities and degrees. However, the special case discussed here is the one in which the node connecting to several groups has high betweenness centrality. We rank users by betweenness centrality and find three nodes both in the top 5% list and with a degree less than 10. They connect with all communities in the network and act as bridges between the different communities. The result is consistent well with the structural holes theory that was advanced by sociologist Ronald Burt in real-life social network study.

6. Conclusions

In this paper, we have studied the structural properties of Microblogging ever created (839 active Sina Microblogging users and their 2112 social relations) from several viewpoints.

First of all, we have found a power-law distribution, a short average length, and a high clustering coefficient in its topology analysis, which are all compatible with known characteristics of other online social networks and real-life social networks. In order to illuminate the mechanisms for the formation of small-world property, we have studied the difference between Sina Microblogging and traditional social networks from the aspect of users' behavior and the users' link mode and found that Sina Microblogging can easily form the user centric We Media. Therefore, it possesses small-world property that can facilitate the flow of information. Then, we calculated the correlation between clustering coefficients and degrees and showed that Sina Microblogging has apparent hierarchical structure, and we have found that Sina Microblogging trends to be disassortative network, which all mark a deviation from real-life social networks. Moreover, we analyzed the betweenness centralities of intermediary nodes and confirmed that the intermediary nodes can control the spread of information. Last but not least, our work is only the first step towards exploring the difference between the online and real-life social networks. Much work still remains.

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 Science Foundation of China under Grant no. 70903016 and the National Science & Technology Support Program under Grant no. 2012BAH81F03.

References

[1] R. Corten, "Composition and structure of a large online social network in the netherlands," *PLoS ONE*, vol. 7, no. 4, Article ID e34760, 2012.

[2] R. E. Wilson and S. D. Gosling, "A review of Facebook research in the social sciences," *Perspectives on Psychological Science*, vol. 7, no. 3, pp. 203–220, 2012.

[3] D. J. Watts and S. H. Strogatz, "Collective dynamics of "small-world" networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[4] R. Albert, H. Jeong, and A.-L. Barabási, "Diameter of the world-wide web," *Nature*, vol. 401, no. 6749, pp. 130–131, 1999.

[5] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[6] M. Barthélemy, "Spatial networks," *Physics Reports*, vol. 499, no. 1, pp. 1–101, 2011.

[7] D. Centola, "The spread of behavior in an online social network experiment," *Science*, vol. 329, no. 5996, pp. 1194–1197, 2010.

[8] Y.-Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong, "Analysis of topological characteristics of huge online social networking services," in *Proceedings of the 16th International World Wide Web Conference (WWW '07)*, pp. 835–844, May 2007.

[9] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a social network or a news media?" in *Proceedings of the19th International World Wide Web Conference (WWW '10)*, pp. 591–600, April 2010.

[10] Q. Yan, L. R. Wu, and L. Zheng, "Social network based microblog user behavior analysis," *Physica A: Statistical Mechanics and Its Applications*, vol. 392, no. 7, pp. 1712–1723, 2013.

[11] W. G. Yuan, Y. Liu, J. J. Cheng, and F. Xiong, "Empirical analysis of microblog centrality and spread influence based on Bi-directional connection," *Acta Physica Sinica*, vol. 62, no. 3, Article ID 038901, 2013.

[12] L. Backstrom and P. Boldi, "Four degrees of separation," in *Proceedings of the 3rd Annual ACM Web Science Conference*, pp. 33–42, 2012.

[13] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: densification and shrinking diameters," *ACM Transactions on Knowledge Discovery from Data*, vol. 1, no. 1, Article ID 1217301, 2007.

[14] B. Gonçalves, N. Perra, and A. Vespignani, "Modeling users' activity on twitter networks: validation of Dunbar's number," *PloS ONE*, vol. 6, no. 8, 2011.

[15] R. I. M. Dunbar, "Social cognition on the Internet: testing constraints on social network size," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 367, no. 1599, pp. 2192–2201, 2012.

[16] G. Miller, "Social scientists wade into the tweet stream," *Science*, vol. 333, no. 6051, pp. 1814–1815, 2011.

[17] R. Lex and B. Kovacs, "A comparison of email networks and off-line social networks: a study of a medium-sized bank," *Social Networks*, vol. 34, no. 4, pp. 462–469, 2011.

[18] S. H. Lee, P.-J. Kim, and H. Jeong, "Statistical properties of sampled networks," *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 73, no. 1, Article ID 016102, 2006.

[19] Y.-Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong, "Analysis of topological characteristics of huge online social networking services," in *Proceedings of the 16th International World Wide Web Conference (WWW '07)*, pp. 835–844, May 2007.

[20] F. Fu, X. Chen, L. Liu, and L. Wang, "Social dilemmas in an online social network: the structure and evolution of cooperation," *Physics Letters A: General, Atomic and Solid State Physics*, vol. 371, no. 1-2, pp. 58–64, 2007.

[21] K. Yuta and N. Ono, "gap in thecommunity-size distribution of a large-scale social networking site," Physics and Society, http://arxiv.org/abs/physics/0701168.

[22] M. Tomassini and L. Luthi, "Empirical analysis of the evolution of a scientific collaboration network," *Physica A: Statistical Mechanics and Its Applications*, vol. 385, no. 2, pp. 750–764, 2007.

[23] K.-I. Goh, Y.-H. Eom, H. Jeong, B. Kahng, and D. Kim, "Structure and evolution of online social relationships: heterogeneity in unrestricted discussions," *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 73, no. 6, Article ID 066123, 2006.

[24] H. Hu, D. Han, and X. Wang, "Individual popularity and activity in online social systems," *Physica A: Statistical Mechanics and Its Applications*, vol. 389, no. 5, pp. 1065–1070, 2010.

[25] P. A. Grabowicz, J. J. Ramasco, E. Moro, J. M. Pujol, and V. M. Eguiluz, "Social features of online networks: the strength of intermediary ties in online social media," *PLoS ONE*, vol. 7, no. 1, Article ID e29358, 2012.

[26] T. A. B. Snijders, G. G. van de Bunt, and C. E. G. Steglich, "Introduction to stochastic actor-based models for network dynamics," *Social Networks*, vol. 32, no. 1, pp. 44–60, 2010.

[27] J. Ugander and B. Karrer, "The anatomy of the facebook social graph," Social and Information Networks, http://arxiv.org/abs/1111.4503.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 653670, 12 pages
http://dx.doi.org/10.1155/2014/653670

Research Article

Local Community Detection in Complex Networks Based on Maximum Cliques Extension

Meng Fanrong, Zhu Mu, Zhou Yong, and Zhou Ranran

School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China

Correspondence should be addressed to Zhu Mu; zhumubest@163.com

Received 6 January 2014; Accepted 25 February 2014; Published 15 April 2014

Academic Editor: Guanghui Wen

Detecting local community structure in complex networks is an appealing problem that has attracted increasing attention in various domains. However, most of the current local community detection algorithms, on one hand, are influenced by the state of the source node and, on the other hand, cannot effectively identify the multiple communities linked with the overlapping nodes. We proposed a novel local community detection algorithm based on maximum clique extension called LCD-MC. The proposed method firstly finds the set of all the maximum cliques containing the source node and initializes them as the starting local communities; then, it extends each unclassified local community by greedy optimization until a certain objective is satisfied; finally, the expected local communities will be obtained until all maximum cliques are assigned into a community. An empirical evaluation using both synthetic and real datasets demonstrates that our algorithm has a superior performance to some of the state-of-the-art approaches.

1. Introduction

In recent years, more and more research has begun to pay attention to large complex networks, such as social networks, protein interaction networks, citation networks, and WWW. Extensive researches have indicated that community structure universally exists in complex networks and the connection between nodes in a community is closer than that between communities. Meanwhile, these nodes often have similar attributes or play a similar role. Therefore, community detection has become one of the basic tasks of complex network analysis and is of important theoretical significance and real value.

The research of community detection mostly has focused on detecting all community structures in a whole network from a global viewpoint [1–5]. However, the large scale of complex networks in many real applications is inconceivable. For example, the friend relation networks on Facebook and Twitter contain hundreds of millions of nodes [6], and the detection of the community structures in such huge complex networks will cost tremendous time and space. In addition, as the nodes and links of many complex networks are dynamically evolving [7, 8], it is often hard for us to acquire the complete network information, further increasing the difficulty in global community detection. Therefore, many scholars have begun to focus on local community detection of complex networks.

Different from the global community detection which classifies a total complex network, local community detection is only to inquire the community structure where a designated node (source node) is located in a network. A complex network is essentially divided into two parts, namely, the community where a designated node is located and the rest part. Furthermore, the local community where the node is located has a close internal connection within the community but a relatively loose relation with the outside. Local community detection need not know all information about a complex network in advance. It starts from a node, gradually extends from the node, and gradually acquires the local information around the current community during the extension process. The representative algorithms for local community detection include [9–13].

However, most of available local community detection algorithms have two restrictions: firstly, the method including direct start from a source node, continuous selection of the best nodes from candidate ones by greedy algorithm, and adding them into a local community till the local community detected satisfies all specified conditions, which makes it

easy to deviate from the real local community, thus reducing the accuracy of local community detection; secondly, in this way, finally, only a unique local community structure can be obtained and when the source node is an overlapped (hub) node connecting multiple communities, it is unable to obtain all local communities.

To solve the above two problems, we propose a local community detection algorithm based on maximum cliques extension (LCD-MC), which includes finding maximum cliques and extending local communities. Its main advantages are shown as below.

(i) Instead of taking the source node as input directly, finding all maximum cliques containing the source node is made as the start of local community extension, thus increasing the stability of local community detection.

(ii) The flexibility in identifying overlapped node-involved local communities is realized by extending all maximum cliques satisfying certain conditions, respectively.

(iii) The experimental results on both synthetic and real networks demonstrate that, compared with the state-of-the-art local community detection algorithms, LCD-MC, on one side, can obtain better local community quality and, on the other side, can effectively identify multiple local community structures connected with the overlapped node.

2. Related Work

Let a complex network $G = (V, E)$, where V represents the node set, E the edge set, and n and m number of nodes and number of edges in the network, respectively. Different from global algorithms which divide G into a number of closely connected community structures, local community detection designates a node v_0 and explores the community structures in close relation with the node v_0.

Clauset firstly proposed the formal definition of local community detection [9]. Assume that we have known a community structure C (initially, C contains only the node v_0) composed of some nodes; set U is connected with nodes in C but does not belong to the node set of community C. The process of local community detection is to continuously select nodes from U and add them to the current community C till the predefined local modularity R reaches the maximum value. To define the objective function R, Clauset also defined the boundary B of community C, that is, node sets in C that have at least one node connecting with U, as shown in Figure 1.

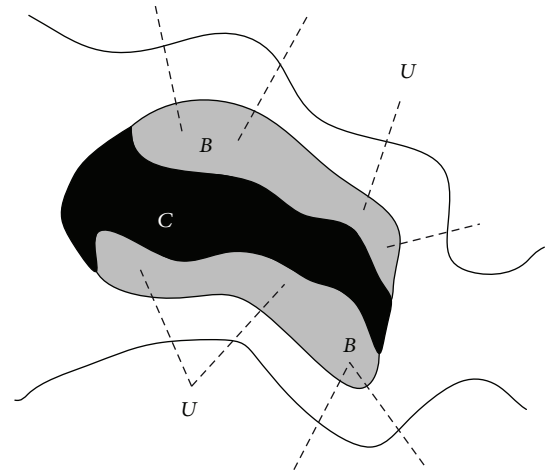


Figure 1: Local community detection model.

Assume that the given C, B, and U are known; R is defined by

$$R = \frac{\sum_{i,j} B_{ij}\delta(i,j)}{\sum_{i,j} B_{ij}}, \tag{1}$$

where $B_{ij} = 1$ if $(i, j) \in E$ and either node v_i or node v_j exists in B; otherwise, $B_{ij} = 0$. When either node v_i or node v_j exists in B and the other node exists in C, then $\delta(i, j) = 1$ and, otherwise, $\delta(i, j) = 0$. The larger the R value is, the better the local community structure detected will be. Initially, $C = \{v_0\}$, and Clauset used a greedy optimization algorithm for the local modularity R to find the local community structure where the designated v_0 is located.

Similar to Clauset algorithm, Luo et al. proposed LWP algorithm [10], in which R is replaced by a new local modularity M, as shown in the following:

$$M = \frac{M_{\text{in}}}{M_{\text{out}}} = \frac{(1/2)\sum_{ij} A_{ij}\theta(i,j)}{\sum_{ij} A_{ij}\lambda(i,j)}, \tag{2}$$

where $\theta(i, j) = 1$ if both node v_i and node v_j exist in community C; otherwise, $\theta(i, j) = 0$. And $\lambda(i, j) = 1$ means that only one, either node v_i or node v_j, exists in community C. In addition to a different objective function, LWP algorithm also includes addition and deletion operations, making it possible to add into or delete from community C the nodes that can increase M value. Besides, LWP algorithm need not predefine the size of community C in advance.

LMF [11] is a local-global community detection algorithm, which proposes a fitness function, as shown in the following:

$$F = \frac{k_{\text{in}}^C}{(k_{\text{in}}^C + k_{\text{ex}}^C)^{\alpha}}, \tag{3}$$

where k_{in}^C and k_{ex}^C refer to the sums of the internal node degrees and external node degrees of community C, respectively, and α is a resolution parameter used for controlling the size of local community. This algorithm is similar to LWP algorithm in that, according to the given α, it achieves the objective of making the fitness function F reach the maximum local value by addition and deletion.

Wu et al. [12] proposed a local community detection algorithm based on link similarity (LS). The algorithm firstly defines the similarity between a single node and a local community and then carries out local community detection

in a decrease sequence of the calculated similarity values. In addition, this algorithm's search process is composed of greedy clustering, optimization, and trimming.

Chen et al. [13] proposed a local community detection algorithm based on local degree center node (LMD). Though the objective of local community detection is to find the community structure where the given node is located, it was held by the authors that, for some given nodes, the detection directly starting from v_0 may not necessarily obtain ideal results. Therefore, to increase the robustness of local community detection, instead of starting the search from the given node, LMD starts from finding a local degree center node nearest to the given node and then extends local communities starting from this local degree center node with R, M, and F as objective functions. Here, the degree of the local degree center node is greater than or equal to that of all neighbor nodes.

3. Algorithm

In this section, we introduce the proposed local community detection algorithm based on maximum cliques extension (LCD-MC), which is mainly composed of two parts, namely, algorithm FindMC for finding the maximum cliques of a node and algorithm LCD for local community extension corresponding to the maximum cliques.

3.1. FindMC Algorithm for Finding Maximum Cliques

Definition 1. Given an undirected graph $G = (V, E)$, if $U \subseteq V$, for random $u, v \subseteq U$, and $(u, v) \subseteq E$, then U is called G's complete subgraph.

Definition 2. G's complete subgraph U is G's maximum clique if and only if U is not included in the G's larger complete subgraphs.

In FindMC, we mainly adopt the concept of the Bron-Kerbosch algorithm [19] to find the maximum cliques where the given source node v_0 is located. That means mainly using three sets, namely, R, P, and X, whose functions are, respectively, explained as follows:

(1) R is used for storing the already acquired nodes forming the current clique structures;

(2) P is used for storing all the candidate nodes edge-connected with the nodes in R, which can be used to extend all the clique structures already found and will be added to R;

(3) X is used for storing the candidate nodes that have been used in P. If a certain node in P can cluster with nodes in R to form a larger clique structure, this node will be added into both set R and set X but deleted from set P.

FindMC algorithm starts from node v_0 and constructs recursively a search tree as the nodes in set P are continuously added to set R. On this search tree, each internal node corresponds to a state or is called a candidate clique structure, while a leaf node represents a corresponding maximum clique.

Algorithm 1 gives the pseudocode of FindMC algorithm for finding maximum clique. In the initialization phase, FindMC stores only node v_0 in set R and only v_0's neighbours in set P. This is because the nodes in the maximum clique where node v_0 is located can only be v_0's neighbours and any search outside this will be invalid (line 01). In addition, to protect the updating of set P, the nodes in P are copied to nodeList to be extended, where the nodes are arranged in a degree-decreasing sequence. This is because that node with a larger degree is easier to form a maximum clique. Therefore, they are given a priority so as to improve the algorithm efficiency (lines 02~03). After the initialization is completed, FindMC algorithm, at its second phase, will execute the conventional Bron-Kerbosch algorithm recursively according to sets R, P, and X and store the acquired maximum cliques in MCS (maximum clique set).

Algorithm 2 is the conventional Bron-Kerbosch algorithm. In this algorithm, if both sets P and X are null sets, then the nodes in R satisfy the condition for maximum clique and will be added to the maximum clique set MCS (lines 01–03). Otherwise, it will select a pivot node u from sets P and X and continue to make self-recursive call for each node v except u in P. When all subprograms of v are ended, v will be deleted from set P and added into set X (lines 04–09).

3.2. LCD Algorithm for Extending Local Community Structure.

A maximum clique is a very closely connected node set; however, the requirement on full connection is too strict to the definition of community structure. Therefore, the second step of LCD-MC is to further extend the local community structure according to the MCS obtained in Algorithm 1, as shown in Algorithm 3 (LCD). LCD algorithm carries out the following operation for each unclassified maximum clique (no node in such clique has been allocated to any local community):

(1) initialization, which will add each node in the current clique to the local community LC and add all nodes connected with but not belonging to LC into set U (lines 04–08);

(2) extending local community LC, which will select a node t from U that can make the greatest increase of the objective function, add it to the local community LC, and update the corresponding nodes in U till there is not any node that can increase the value of the objective value (lines 09–21);

(3) finally, return the required local community set LCS, each community of which containing the initial node v_0 (line 23).

On line 12 of Algorithm 3, function CalculateDeltaValue (u) calculates the incremental value of the objective function after node u is added to the current local community. Here, the objective function can be any objective function mentioned in Section 2. To improve the algorithm efficiency, the incremental value of objective function can be calculated

```
Input: G = (V, E), v_0
Output: Maximum Clique Set MCS
Begin:
(1) R ← {v_0}, P ← {w | (v_0, w) ∈ E}, X ← Ø, MCS ← Ø
(2) nodeList ← P
(3) sort the nodes in nodeList based on their degree descending
(4) foreach v in nodeList
(5)     Bron_Kerbosch (R ∪ {v}, P ∩ N(v), X ∩ N(v), MCS);
(6)     P ← P \ {v};
(7)     X ← X ∪ {v};
(8) endfor
(9) retrun MCS
End
```

Algorithm 1: FindMC.

```
Input: R, P, X, MCS
Output: MCS
Begin:
(1) if P = Ø and X = Ø
(2)       MCS ← MCS ∪ R;
(3) else
(4)       u ← a pivot node in P ∪ X
(5)       foreach v in P \ {u}
(6)           BronKerbosch (R ∪ {v}, P ∩ N(v), X ∩ N(v), MCS);
(7)           P ← P \ {v};
(8)           X ← X ∪ {v};
(9)       endfor
(10) endif
(11) retrun MCS
End
```

Algorithm 2: Bron-Kerbosch.

according to the assumed change value generated after the addition of node u into the local community. Take the objective function M in (2) as an example. Assume the number of edges in the current local community LC is E_{in}, the number of the edges between LC and U is E_{out}, where x represents the delta value of edge number in LC due to the addition of u, and y represents the number of edges of nodes in LC connecting U after the addition of u into LC; then, the delta value of M is

$$\text{delta} = \frac{xE_{\text{out}} + (x - y)E_{\text{in}}}{(E_{\text{out}} - x + y)E_{\text{in}}}. \tag{4}$$

3.3. Time Complexity. Let us analyze the time complexity of LCD-MC from its two steps, respectively. As the worst time complexity of Bron-Kerbosch algorithm is $O(3^{n/3})$ [20], but LCD-MC only needs to find cliques containing the initial node, it essentially operates Bron-Kerbosch algorithm in the subgraphs formed by the initial node and its neighbors. In this way, the worst time complexity of FindMC, the first step in LCD-MC, is $O(3^{k/3})$, where k is the degree of the initial node, and that of LCD, the second step in LCD-MC, is $O(|C|^2)$, where C is the local community detected. It should be noted that either k in FindMC or $|C|$ in LCD is far smaller than the overall size of the network. Therefore, LCD-MC indicates satisfied time efficiency.

4. Evaluation

In this section, we make the verification comparison of LCD-MC proposed in this paper with several representative local community detection algorithms. We conduct all the experiments on a Pentium Core2 Duo 2.8 GHz PC with 2 GBytes of main memory, running on Windows 7. We implement our algorithm in C#, using Microsoft Visual Studio 2008.

4.1. Experimental Setup. We compared LCD-MC with several representative local community detection algorithms including Clauset [9], LWP [10], LS [12], and LMD [13]. Of them, Clauset was the first to propose local community detection, and, therefore, we take it as the basic algorithm. Both

```
Input: G = (V, E), MCS
Output: Local Communities LCS
Begin:
(1) LCS ← Ø, U ← Ø
(2) foreach unclassified MC in MCS
(4)    LC ← Ø, U ← Ø
(5)    foreach v in MC
(6)       LC ← LC ∪ {v}
(7)       Initialize U;
(8)    endfor
(9)    while true
(10)      max ← ∞
(11)      foreach u in U
(12)         delta ← CalculateDeltaValue (u);
(13)         t ← u, max ← delta;
(14)         if max > 0
(15)            LC ← LC ∪ {t};
(16)            Update N;
(17)         else
(18)            break;
(19)         endif
(20)      endfor
(21)   LCS ← LCS ∪ {LC}
(22) endfor
(23) retrun LCS
End
```

Algorithm 3: LCD.

Table 1: Information of benchmark networks.

ID	N	k	max k	min c	max c	mu
S1	1000	10	50	10	50	0.1–0.9
S2	1000	10	50	20	100	0.1–0.9
S3	5000	10	50	10	50	0.1–0.9
S4	5000	10	50	10	50	0.1–0.9

Table 2: Information of real networks.

Name	Nodes	Edges	Communities	Reference
Football	115	616	12	[14]
Polbooks	105	441	3	[15]
Karate	34	78	2	[16]
Adjnoun	112	425	2	[17]
Polblogs	1490	19090	2	[18]

LWP and LS make self-inspection and delete the nodes not satisfying certain conditions while making local community extension. LWP takes M in (2) as the objective function, while LS takes node similarity factor into comprehensive consideration and can adopt any objective function. Both LMD and LCD-MC start local community detection from a node other than the source node. LMD starts from the local degree center node nearest to the source node, while LCD-MC starts from the maximum cliques where the source node is located. Both LMD and LCD-MC are suitable for different objective functions. In our experiment, LS, LMD, and LCD-MC all took M as the objective function.

We firstly compared the quality of local communities found by these algorithms. As these algorithms, except LCD-MC, have no ability to identify local communities connected by overlapping nodes, in experimental data, we selected LFR benchmarks [21, 22] with nonoverlapped structure and several labeled real networks, whose information is as shown in Tables 1 and 2. The meaning of the parameters in Table 1 is described as follows: N, the number of nodes; k, the average degree; max k, the maximum degree; min c, the minimum for the community sizes; max c, maximum for the community sizes; mu, a mixing parameter, the probability of nodes connected with nodes of external community. It should be pointed out that, to find the only local community structure, in the second step of LCD-MC, the maximum clique with the most number of nodes was selected for extension.

To evaluate the quality of the local communities generated by various methods, we adopt F-Measure score (FM) and normalized mutual information (NMI) [23] as the evaluation indexes.

FM is a commonly used measure for community detection algorithms. Assume T is the set of node pairs (i, j), where nodes i and j belong to the same classes in the ground truth,

and S is the set of node pairs that belong to the same communities generated by an algorithm. Then FM is computed from both the precision and the recall synthetically:

$$\mathrm{FM} = \frac{2 \times \mathrm{precision} \times \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}, \tag{5}$$

where precision and recall are written as (6). Consider that

$$\mathrm{precision} = \frac{|S \cap T|}{|S|}, \qquad \mathrm{recall} = \frac{|S \cap T|}{|T|}. \tag{6}$$

NMI is another widely used criterion for measuring the performance of community detection algorithms. Formally, the measurement of NMI can be defined as

$$\mathrm{NMI} = \frac{-2 \sum_{i=1}^{r} \sum_{j=1}^{k} N_{ij} \log\left(N_{ij} N / N_{i.} N_{.j}\right)}{\sum_{i=1}^{r} N_{i.} \log\left(N_{i.}/n\right) + \sum_{j=1}^{k} N_{.j} \log\left(\left(N_{.j}/N\right)\right)}, \tag{7}$$

where N is the confusion matrix, N_{ij} is the number of nodes both in the ith class and the jth cluster, r and k are the number of classes and clusters, respectively, and $N_{i.}$ and $N_{.j}$ are the number of nodes in the ith class and the jth cluster, respectively.

4.2. Synthetic Networks. Figures 2, 3, 4, and 5 are the experimental results on the synthetic networks S1, S2, S3, and S4 conducted by the five local community detection algorithms. Each figure includes four subgraphs and the abscissa of each subgraph represents the value of mixed parameter *mu*. The larger the value, the less distinct the community structure; meanwhile it also shows the greater difficulty in finding corresponding community. The ordinates of each subgraph represent the evaluation indexes, which, from left to right and from upper to lower, are precision, recall, FM, and NMI.

From the experimental results, we can acquire the following observation.

(1) Comparison of LWP and LS with Clauset. It can be seen that, compared with the basic algorithm, both LWP and LS can only focus on either precision or recall. That means that when a high precision can be achieved in local community detection, the recall is often relatively low, and vice versa. Viewing from FM and NMI comprehensive evaluation indexes, neither algorithm can guarantee finding a better local community than that by Clauset. Therefore, it can be known that the critical factor in local community detection is not the self-inspection of the community detected.

(2) Comparison of LCD-MC and LMD with Clauset. It can be seen that, in terms of the four indexes, both these algorithms can achieve better results than those by Clauset to various extents, indicating that there are really certain restrictions in starting local community detection from the initial node. Viewing from the experimental results, compared with LMD algorithm which starts the detection from local degree center point, the LCD-MC algorithm that starts from node maximum clique is more effective.

(3) As a whole, LCD-MC achieves the best results of all four evaluation indexes. On both small community networks (S1 and S3) with a mixed parameter mu smaller than 0.6 and large community networks (S2 and S4) with a mixed parameter mu smaller than 0.5, LCD-MC could find out the local community structure of each node almost fully correctly. On a highly mixed network, for example, with a mu of 0.8 or 0.9, neither LCD-MC nor the other algorithms could obtain ideal results. This just conforms to the real condition that network community structure is not distinct.

Table 3: Results on real networks.

Networks	Index	Clauset	LWP	LS	LMD	LCD-MC
Karate	Precision	0.905	0.601	0.852	**0.924**	0.908
	Recall	0.585	0.278	0.375	0.599	**0.658**
	FM	0.675	0.329	0.435	0.694	**0.744**
	NMI	0.31	0.125	0.157	0.343	**0.397**
Football	Precision	0.665	0.588	0.59	0.723	**0.9**
	Recall	0.743	0.543	0.554	0.848	**0.895**
	FM	0.691	0.475	0.483	0.77	**0.896**
	NMI	0.594	0.301	0.308	0.665	**0.841**
Polbooks	Precision	0.772	**0.824**	0.797	0.768	0.787
	Recall	0.477	0.203	0.221	**0.784**	0.691
	FM	0.52	0.24	0.259	**0.756**	0.696
	NMI	0.312	0.138	0.14	**0.459**	0.429
Adjnoun	Precision	0.541	0.409	0.278	0.478	**0.524**
	Recall	0.24	0.15	0.187	0.615	**0.669**
	FM	0.224	0.128	0.155	0.431	**0.48**
	NMI	0.01	**0.047**	0.044	0.005	0.003
Polblogs	Precision	0.87	0.694	0.616	0.913	**0.917**
	Recall	0.045	0.205	0.207	0.681	**0.722**
	FM	0.062	0.211	0.212	0.744	**0.788**
	NMI	0.058	0.075	0.073	0.431	**0.454**

To sum up, LCD-MC can find local communities with better quality on synthetic networks compared with the other representative local community detection algorithms.

4.3. Real Networks. To further verify the performance of LCD-MC, we compare it with the other algorithms on real networks and show the comparison results in Table 3. The bold digits are the maximum value of local community quality of each algorithm for the related evaluation index. Of them, precision and recall only reflect one aspect of algorithm performance, while FM and NMI take algorithm performance into a comprehensive consideration and, therefore,

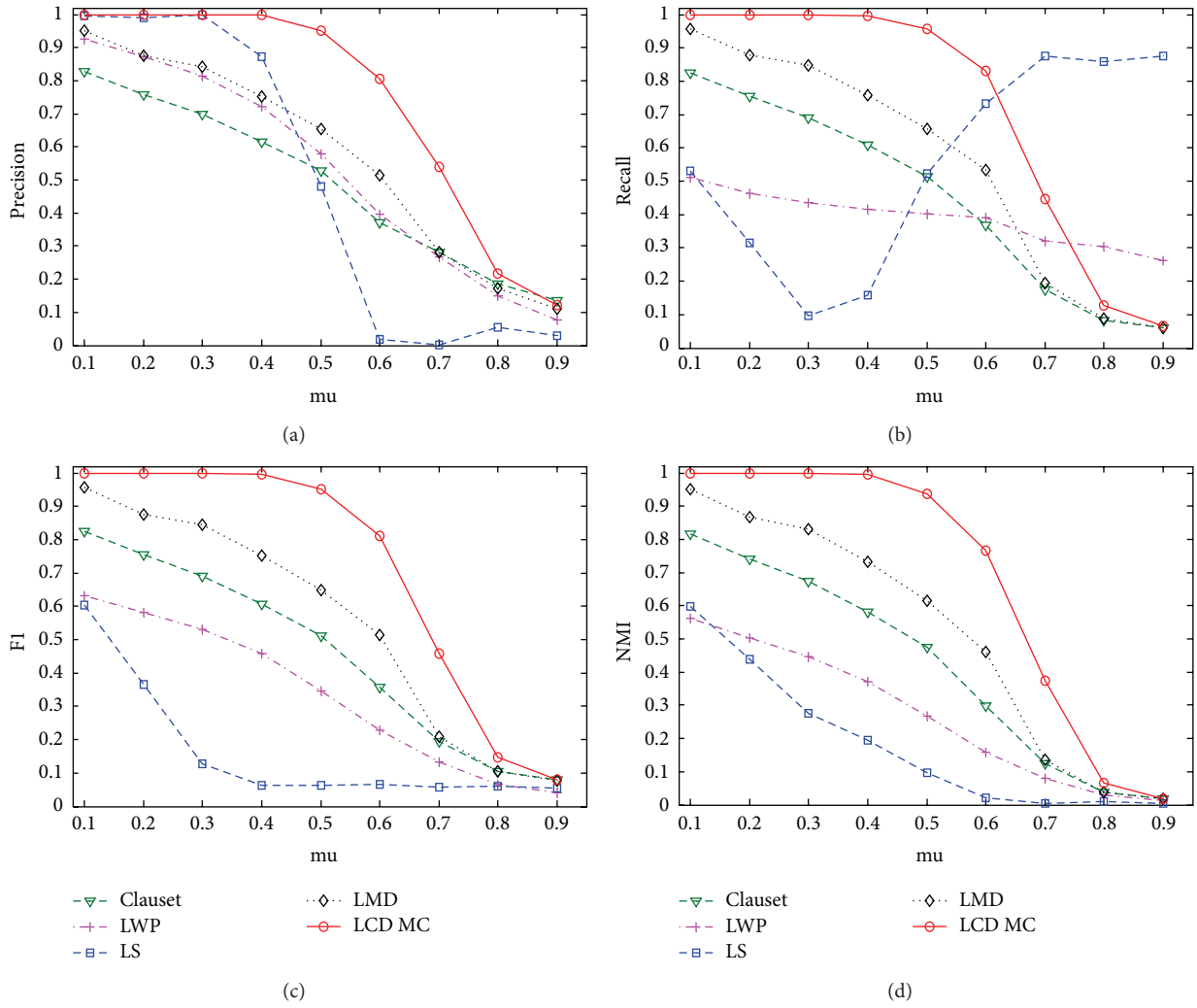


Figure 2: Results on S1.

Table 4: The real community of the synthetic community.

Community ID	Community elements
1	{1, 10, 15, 20, 21, 28, 35, 38, 46, 49, 51, 59, 61, 63, 66, 69, 73, 74, 80, 82, 87}
2	{5, 8, 22, 24, 25, 26, 29, 31, 32, 34, 37, 39, 42, 45, 47, 55, 56, 57, 58, 62, 70, 84, 86, 89, 96, 97}
3	{0, 2, 3, 4, 6, 9, 11, 13, 14, 30, 41, 44, 48, 50, 53, 54, 56, 64, 68, 76, 81, 83, 90, 91, 94, 95, 98}
4	{2, 7, 12, 16, 17, 18, 19, 23, 27, 28, 33, 36, 37, 40, 43, 52, 60, 65, 67, 71, 72, 75, 77, 78, 79, 85, 88, 92, 93, 99}

Table 5: Local community result of Clauset on the synthetic network.

Node ID	Community elements
2	{0, 2, 3, 4, 6, 9, 11, 13, 14, 30, 41, 44, 48, 50, 53, 54, 56, 64, 68, 76, 81, 83, 90, 91, 94, 95, 98}
28	{5, 8, 22, 24, 25, 26, 28, 29, 31, 32, 34, 37, 39, 42, 45, 47, 51, 55, 56, 57, 58, 62, 70, 84, 86, 89}
37	{5, 8, 22, 24, 25, 26, 29, 31, 32, 34, 37, 39, 42, 45, 47, 55, 56, 57, 58, 62, 70, 84, 86, 89, 96, 97}
56	{5, 8, 22, 24, 25, 26, 29, 31, 32, 34, 37, 39, 42, 45, 47, 55, 56, 57, 58, 62, 70, 84, 86, 89, 96, 97}

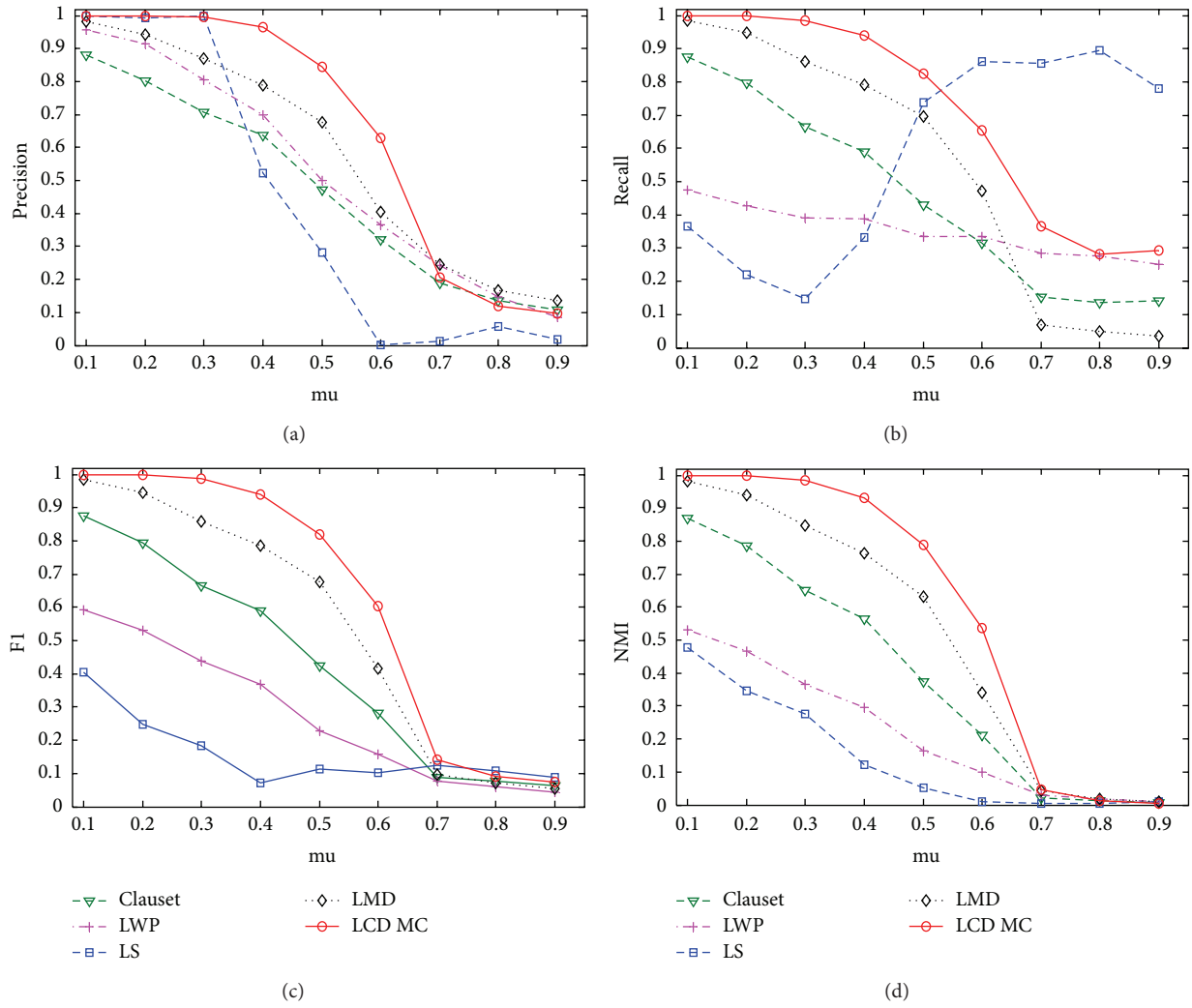


Figure 3: Results on S2.

Table 6: Local community result of LCD-MC on the synthetic network.

Node ID	Community elements
2	{2, 7, 12, 16, 17, 18, 19, 23, 27, 33, 36, 40, 43, 52, 60, 65, 67, 71, 72, 75, 77, 78, 79, 85, 88, 92, 93, 99} {0, 2, 3, 4, 6, 9, 11, 13, 14, 30, 41, 44, 48, 50, 53, 54, 56, 64, 68, 76, 81, 83, 90, 91, 94, 95, 98}
28	{1, 10, 15, 20, 21, 28, 35, 38, 46, 49, 51, 59, 61, 63, 66, 69, 73, 74, 80, 82, 87} {2, 7, 12, 16, 17, 18, 19, 23, 27, 28, 33, 36, 40, 43, 52, 60, 65, 67, 71, 72, 75, 77, 78, 79, 85, 88, 92, 93, 99}
37	{5, 8, 22, 24, 25, 26, 29, 31, 32, 34, 37, 39, 42, 45, 47, 55, 56, 57, 58, 62, 70, 84, 86, 89, 96, 97} {2, 7, 12, 16, 17, 18, 19, 23, 27, 33, 36, 37, 40, 43, 52, 60, 65, 67, 71, 72, 75, 77, 78, 79, 85, 88, 92, 93, 99}
56	{5, 8, 22, 24, 25, 26, 29, 31, 32, 34, 37, 39, 42, 45, 47, 55, 56, 57, 58, 62, 70, 84, 86, 89, 96, 97} {0, 2, 3, 4, 6, 9, 11, 13, 14, 30, 41, 44, 48, 50, 53, 54, 56, 64, 68, 76, 81, 83, 90, 91, 94, 95, 98}

are of more comparison significance. We can see that LCD-MC achieved the best results on Karate, Football, Polblogs, and Adjnoun networks (all algorithms achieved a very low NMI value on Adjnoun, which can be neglected). Though, on Polbooks network, LCD-MC results are not the best, they are only slightly inferior to the best results achieved by LMD algorithm.

To sum up, LCD-MAC also achieved better experimental results on real networks compared with the other local community detection algorithms.

4.4. Local Communities of Overlapped Node. We verified the ability of LCD-MC to identify multiple local communities where overlapped nodes are located on a LFR synthetic

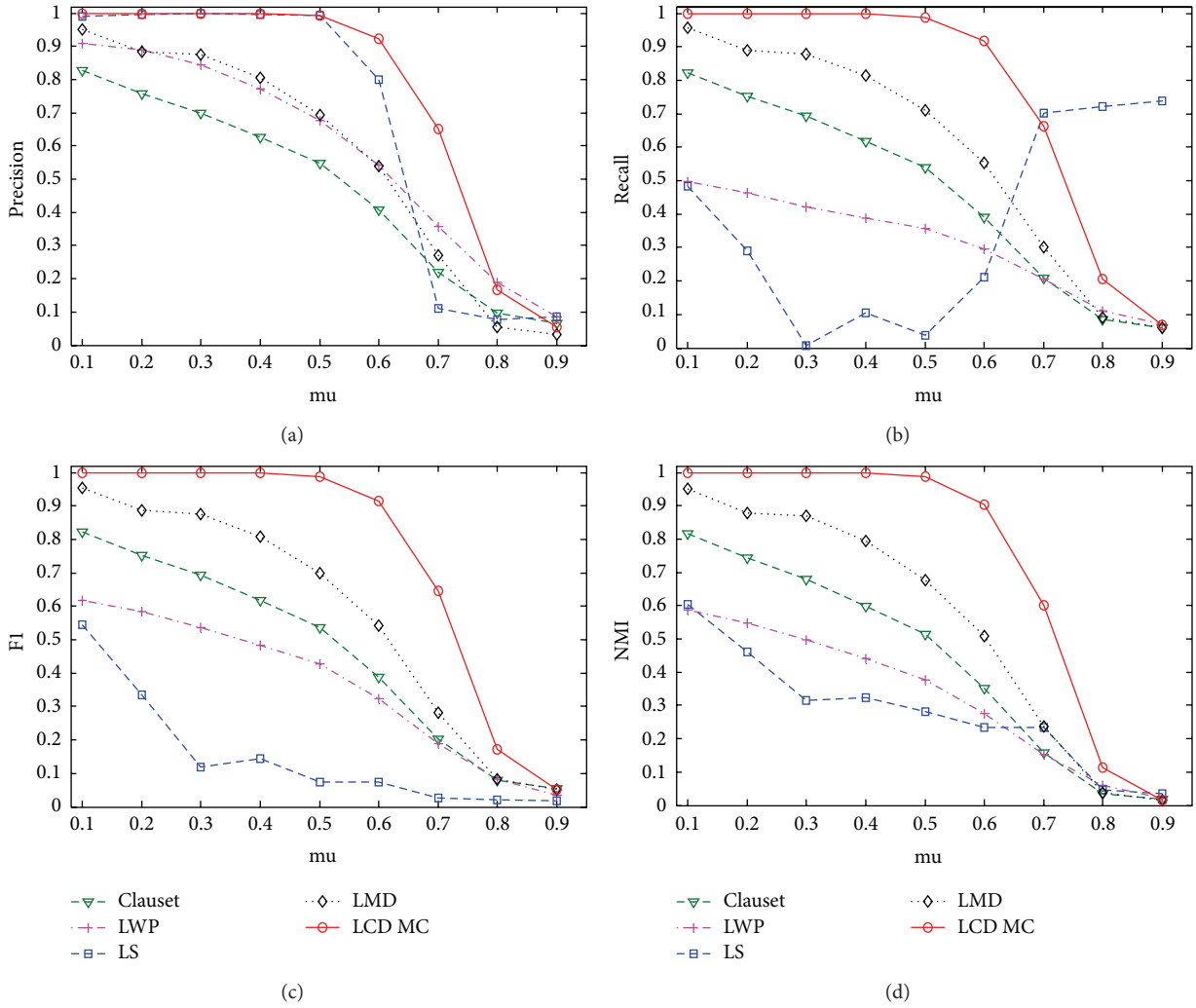


Figure 4: Results on S3.

network. The configuration parameters of the LFR synthetic network are $N = 100$, $k = 10$, $\max k = 20$, $\min c = 20$, $\max c = 30$, mu = 0.1, on = 4, and om = 2, where on represents the number of the overlapping nodes, and om represents the number of memberships of the overlapping nodes. The corresponding network layout is as shown in Figure 6. Table 4 shows the corresponding community distribution, in which 2, 28, 37, and 56 are overlapped nodes and each node connects two communities.

As Clauset, LWP, LS, and LMD algorithms can obtain only one local community from each node; we selected Clauset as their representative in comparison with LCD-MC algorithm. The results of local community detection on the network depicted in Figure 6 by Clauset and LCD-MC are shown in Tables 5 and 6, respectively. It can be seen that Clauset only found out one local community from each overlapped node, because Clauset algorithm can only extend the current only local community according to objective function R. LCD-MC effectively found out two local communities from each overlapped node. This is because, in an initialization, LCD-MC uses maximum cliques as candidate local communities instead of only one node. Take node 2 as an example. In initialization, its maximum clique sets included $\{2, 79, 93\}$, $\{2, 79, 99\}$, and $\{2, 83, 94\}$. Of them, the maximum clique $\{2, 79, 93\}$ was used as the initial local community for extension and, as a result, community 4 was found, as shown in Table 4. At this stage, the maximum clique $\{2, 79, 99\}$ was already included in this local community, but the maximum clique $\{2, 83, 94\}$ was still outside it. Therefore, LCD-MC started from $\{2, 83, 94\}$ to continue the search for new local community and, then, found two local community structures connected with node 2. Therefore, LCD-MC has the ability to find out multiple local communities connected by overlapped nodes.

5. Conclusion

In this paper, we propose a novel local community detection algorithm for large complex networks based on maximum cliques extension (LCD-MC). This algorithm firstly adopts

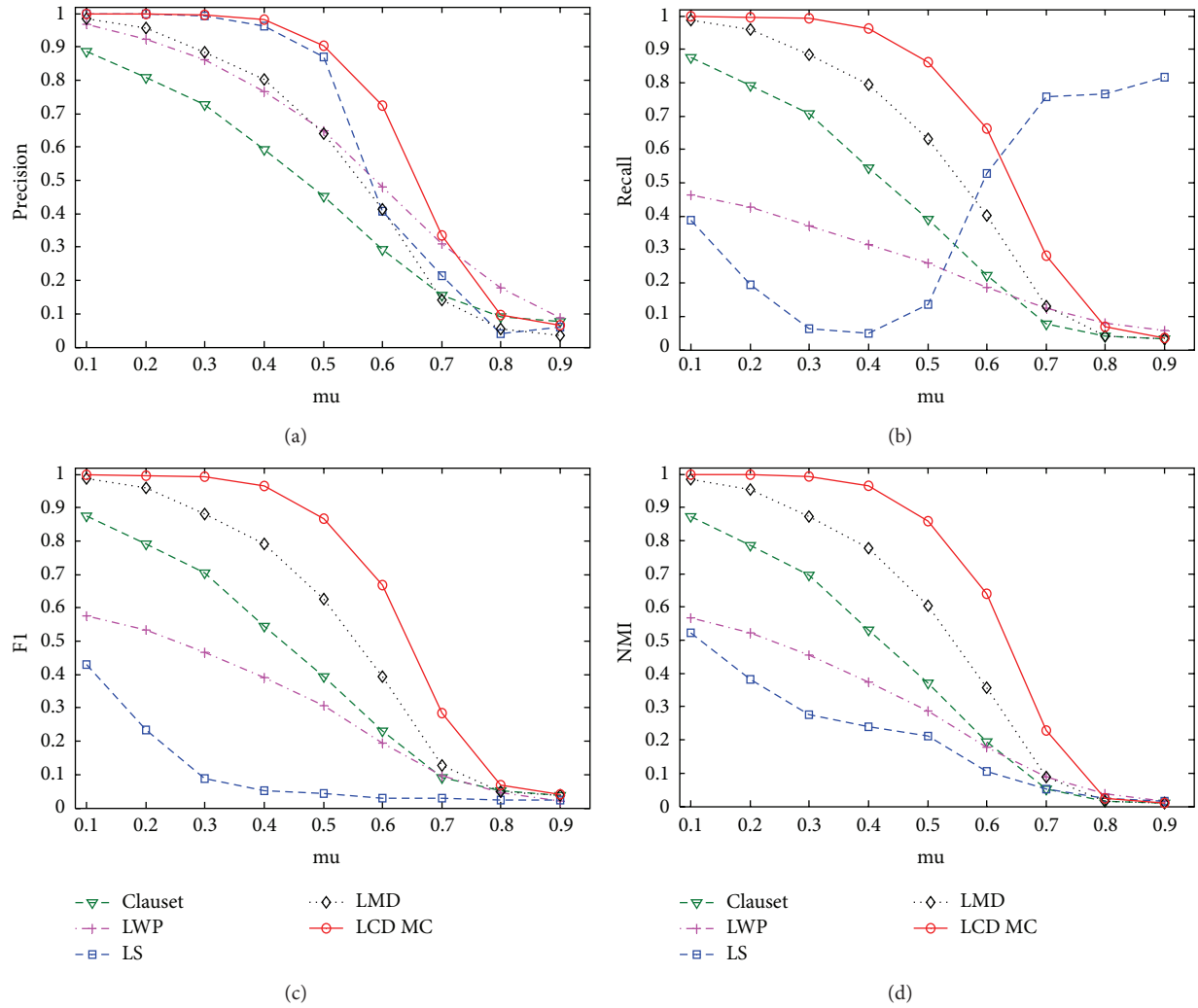


Figure 5: Results on S4.

the idea of the Bron-Kerbosch algorithm to find out all maximum cliques containing the source node in the network by recursion, then, takes an arbitrary maximum clique satisfying certain conditions as the initial local community, and, by continuous exploring the neighbor area around the current local community, continuously adds conforming nodes to the local community structure till there is not any conforming node. LCD-MC algorithm is most characterized by starting the extension from the maximum clique of a given node instead of starting directly from the given node. In this way, it avoids the deviation of community extension and can identify multiple local community structures connected by an overlapped node. We compared LCD-MC with some representative algorithms such as Clauset, LWP, LS, and LMD on various synthetic and real networks. The experimental results demonstrate that LCD-MC algorithm can find local communities with better quality on both synthetic and real networks. Moreover, the experimental results on a synthetic network with known overlapped community structures indicate that the LCD-MC algorithm has the ability to identify multiple local communities where the overlapped node is located.

Conflict of Interests

The authors declare that they have no financial and personal relationships with other people or organizations that can inappropriately influence their work; there is no professional or other personal interest of any nature or kind in any product, service, or company that could be construed as influencing the position presented in, or the review of, the paper.

Acknowledgments

This work was supported by the National High Technology Research and Development Program of China (Grant no. 2012AA0622022 and Grant no. 2012AA011004), the Doctoral Fund of Ministry of Education of China (Grant no. 20100095110003 and Grant no. 20110095110010), the State

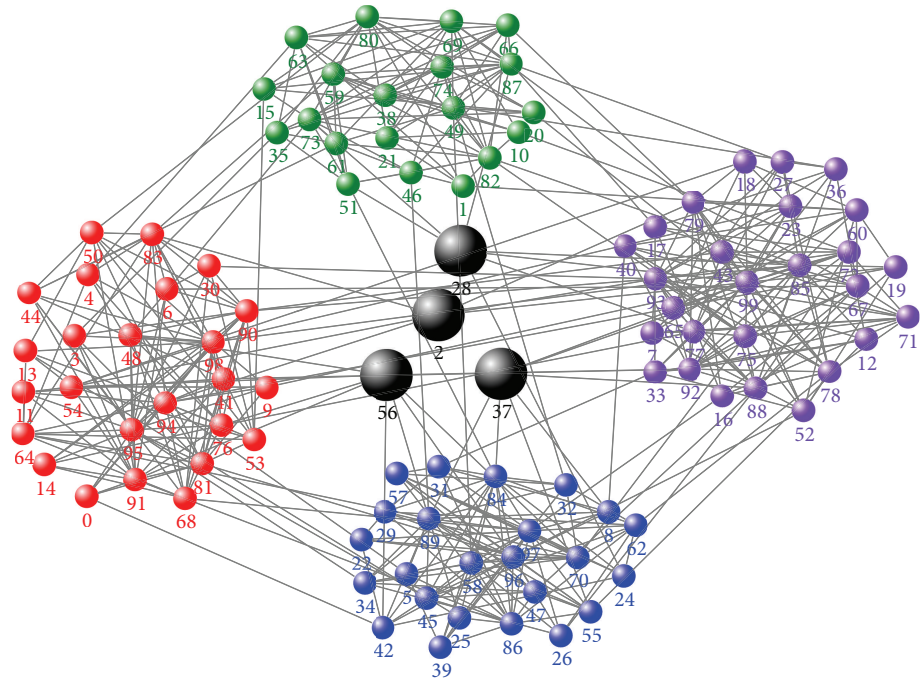


Figure 6: An example of synthetic network. Created with NodeXL (http://nodexl.codeplex.com/).

863 projects (Grant no. 2012AA011004), and the Graduate Research and Innovation Projects in Jiangsu Province (Grant no. CXZZ12_0934).

References

[1] M. Coscia, F. Giannotti, and D. Pedreschi, "A classification for community discovery methods in complex networks," *Statistical Analysis and Data Mining*, vol. 4, no. 5, pp. 512–546, 2011.

[2] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3–5, pp. 75–174, 2010.

[3] J. Leskovec, K. J. Lang, and M. Mahoney, "Empirical comparison of algorithms for network community detection," in *Proceedings of the 19th International World Wide Web Conference (WWW '10)*, pp. 631–640, April 2010.

[4] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks the state of the art and comparative Study," *ACM Computing Surveys*, vol. 45, no. 4, pp. 1–37, 2013.

[5] D. Liu, D. Jin, D. He, J. Yang, and B. Yang, "Community mining in complex networks," *Journel of Computer Research and Development*, vol. 50, no. 10, pp. 2140–2154, 2013.

[6] M. Coscia, G. Rossetti, and F. Giannotti, "A local-first discovery method for overlapping communities," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '12)*, pp. 615–623, ACM, August 2012.

[7] G. Wen, Z. Duan, G. Chen, and X. Geng, "A weighted local-world evolving network model with aging nodes," *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 21-22, pp. 4012–4026, 2011.

[8] X. Geng and G. Wen, "Weighted evolving networks with intrinsic strength," *International Journal of Modern Physics C*, vol. 18, no. 9, pp. 1435–1442, 2007.

[9] A. Clauset, "Finding local community structure in networks," *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 72, no. 2, Article ID 026132, 2005.

[10] F. Luo, J. Z. Wang, and E. Promislow, "Exploring local community structures in large networks," *Web Intelligence and Agent Systems*, vol. 6, no. 4, pp. 387–400, 2008.

[11] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New Journal of Physics*, vol. 11, no. 3, Article ID 033015, 2009.

[12] Y. J. Wu, H. Huang, Z. F. Hao et al., "Local community detection using link similarity," *Journal of Computer Science and Technology*, vol. 27, no. 6, pp. 1261–1268, 2012.

[13] Q. Chen, T. T. Wu, and M. Fang, "Detecting local community structure in complex networks based on local degree central nodes," *Physica A*, vol. 392, no. 3, pp. 529–5537, 2013.

[14] 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.

[15] http://www.orgnet.com/.

[16] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, vol. 33, no. 4, pp. 452–473, 1977.

[17] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 74, no. 3, Article ID 036104, 2006.

[18] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 US election: divided they blog," in *Proceedings of the 3rd International Workshop on Link Discovery*, pp. 36–43, ACM, August 2005.

[19] C. Bron and J. Kerbosch, "Algorithm 457: finding all cliques of an undirected graph," *Communications of the ACM*, vol. 16, no. 9, pp. 575–577, 1973.

[20] E. Tomita, A. Tanaka, and H. Takahashi, "The worst-case time complexity for generating all maximal cliques and computational experiments," *Theoretical Computer Science*, vol. 363, no. 1, pp. 28–42, 2006.

[21] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 78, no. 4, Article ID 046110, 2008.

[22] A. Lancichinetti and S. Fortunato, "Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities," *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 80, no. 1, Article ID 016118, 2009.

[23] M. Zhu, F. Meng, and Y. Zhou, "Semisupervised clustering for networks based on fast affinity propagation," *Mathematical Problems in Engineering*, vol. 2013, Article ID 385265, 13 pages, 2013.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 107246, 12 pages
http://dx.doi.org/10.1155/2014/107246

Research Article

Reorganizing Complex Network to Improve Large-Scale Multiagent Teamwork

Yang Xu, Pengfei Liu, and Xiang Li

School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Correspondence should be addressed to Yang Xu; xuyang@uestc.edu.cn

Received 5 February 2014; Accepted 26 February 2014; Published 14 April 2014

Academic Editor: Guoqiang Hu

Large-scale multiagent teamwork has been popular in various domains. Similar to human society infrastructure, agents only coordinate with some of the others, with a peer-to-peer complex network structure. Their organization has been proven as a key factor to influence their performance. To expedite team performance, we have analyzed that there are three key factors. First, complex network effects may be able to promote team performance. Second, coordination interactions coming from their sources are always trying to be routed to capable agents. Although they could be transferred across the network via different paths, their sources and sinks depend on the intrinsic nature of the team which is irrelevant to the network connections. In addition, the agents involved in the same plan often form a subteam and communicate with each other more frequently. Therefore, if the interactions between agents can be statistically recorded, we are able to set up an integrated network adjustment algorithm by combining the three key factors. Based on our abstracted teamwork simulations and the coordination statistics, we implemented the adaptive reorganization algorithm. The experimental results briefly support our design that the reorganized network is more capable of coordinating heterogeneous agents.

1. Introduction

Cooperative multiagent and multirobot teams are promising in the domain of distributed artificial intelligence, such as controlling of a large number of unmanned aerial vehicles [1], coordinating soldiers, agents and robots in battlefield [2], and searching and rescuing in disasters [3]. In those teams, agents work together toward their common goal, and similar to the infrastructure of human society, they only closely coordinate with a few teammates either logically or under the constraints of physical communications, with a peer-to-peer associate network structure. For example, in a multi-UAV team, members communicate through a wireless ad hoc network structure. When the team scales up, the network presents the complex network structure in the dynamic team coordination [4].

How to improve the team performance by adjusting the network topology has been extensively studied. Some special complex networks attributes are found to be important to the social network performance. For example, Gaston found that the tight coupled network structure will weaken the team performance because of the excessive consumption [5]. Scerri has analyzed how the network structure affects the efficiency of the multiagent communication network [6]. Following those discoveries, structure-oriented approaches [7] take the advantages of the discoveries of complex network effects, such as the scale-free, small-world property and the community structure on the team performance [8, 9]. On the other hand, actor-oriented approaches [7] focus on analyzing the characters of agents' behaviors on the networked coordination and making use of agents' role features to adjust team organization based on a given network structure such as hierarchical or grid-based networks [7]. For example, Jiang et al. [10] proposed a task assignment algorithm for heterogeneous agents with an understanding of its "actor" in the social network. Although significant progress has been made in both structure and actor-oriented approaches, no previous research combines both advantages to build an integrated team reorganization algorithm in real multiagent coordination domains [7].

In this paper, to combine both structure and actor-oriented approaches, we made three efforts. First, we analyze

how different network topologies with different complex attributes may influence the agents' networked coordination by Markov chain model. The discovery is exciting because potential scale-free and small-world effects help the team a lot especially when coordination interaction can be intelligently routed. Second, we found that agents always try to connect each other directly or within very short distance if they are closely cooperated. However, the cooperative relationship is the intrinsic character of the team and independent of how team is organized, for example, the capabilities of heterogeneous agents and the subteams in which agents closely cooperate for joint activities [11]. Similar to human society, people closely coordinate or communicate because some kinds of relationships exist. For example, a letter will be delivered from the mother to her daughter, or bread will be sent from the bakery to a hungry person. Human society will keep routing between those pairs of persons no matter how far they are. Therefore, if we can reorganize the team according to this nature and connect those closely coordinated agents with less links, the team will be benefited most.

To make our team organization optimization approach possible, we simulated the peer-to-peer coordination of large-scale heterogeneous multiagent systems. In our simulation, the resources, tasks, and other coordination information are abstracted and encapsulated into messages, which can be passed from the source agent to the destination via the coordination network. This method is easy to track the messages so that we can infer which pairs of agents are closely coordinated. To find the closely cooperative relationships, we built a connection matrix and recorded messages' resources and destinations. In addition, the subteam model is used to record the agents who are involved in the same joint activity.

As the last part of our design, we set up the adaptive reorganization algorithm to integrate the three key factors we have analyzed. In the network construction phase, by building a probability model for each pair of agents, agents recursively pick their neighbors based on their probabilities. The probability of connecting each pair of agents is measured by how closely cooperated they are, within a subteam, or how much they help to promote the scale-free and small-world effects if they are directly connected. Therefore, in our algorithm, agents always pick neighbors who are closely cooperative, with higher degrees and shorter distances, but with different important factors. Our experiment results briefly match our expectations. By reorganizing the team as a small-world and scale-free network and connecting closely cooperative agents with shorter distances, the network can significantly expedite team performance.

2. Scalable Team Coordination

In this section, we build the scalable multiagent team network model, in which pairs of agents are defined as connected only when they are able to interact with each other directly and each agent only maintains its peer-to-peer connection with a few of others. The objective of their interactions is to jointly perform their tasks so that their complex team goal can be achieved.

2.1. Large Team Organization. The organizational topology is described as an undirected graph: $G = (A, E)$, where multiagent A is the node set of G, $A = \{a_1, \ldots, a_k\}$ and E is the set of edges; if there is $e_k = \langle i, j \rangle$, then coordination messages could be transmitted between a_i and a_j directly; that is, they are neighbors to each other. $N(i)$ is defined as the set of a_i's neighbors.

G could be organized based on the properties of complex networks. In this paper, we are mainly interested in four of the topologies: random network, grid network, small-world network, and scale-free network. Preliminary studies [12] found that each topology encodes the following different fundamental properties.

(i) Degree: the degree of agent a_i is $d(i) = |n(i)|$.

(ii) Average degree: $\overline{d} = (1/N) \sum_{i \in V} |n(i)|$ is the average number of neighbors of all agents, for any complex network $\overline{d} \ll |v|$.

(iii) Degree distribution: $p(k) = \Pr[d = k]$ is defined as a fraction of agents (the number of such agents is d) with the degree k.

(iv) Distance: $distance(i, j)$ defines the least number of hops to communicate between agents a_i and a_j. Specifically, $distance(i, j) = 1$, if $\langle i, j \rangle \in E$.

(v) Average distance is the average distance between any pair of agents:

$$l = \frac{1}{N(N-1)} \sum_{\forall a_i, a_j \in A} distance(i, j). \quad (1)$$

Different complex network topologies can be described according to the properties. In this paper, we mainly consider the two effects and used *degree distribution* to express the scale-free effect and *average distance* to express the small-world effect.

2.2. Multiagent Team Coordination. A large multiagent team coordination can be briefly described as follows: agents are cooperative on a joint goal. It can be decomposed into discrete subgoals $g_1, \ldots, g_i$. To achieve the subgoals, the corresponding tasks $\alpha_1, \ldots, \alpha_i$ are typically performed by individuals. Agents must perform the individual tasks α, when they are applicable, for the team to receive reward. An amount of reward will be received by the team when an agent performs a task. The reward depends on the agent and task, the capability of the agent, and the resources that the agent has. Specifically,

$$Reward(a, \alpha, Capability(a, \alpha), Holds(a)) \longrightarrow \mathcal{R}. \quad (2)$$

The function $Assigned(a, \alpha) = 1$, if agent a is assigned a task α; otherwise it is equal to 0. A given task is only allowed to be assigned to one agent at any time; that is, $\sum_{a \in A} Assigned(a, \alpha) \leq 1$. However, agents may expect different utilities on the same task based on their capabilities. For example, we should expect higher reward for a fireman to do fire fighting than a nonexperienced civilian. The function

Capability(a, α) projects a real value to denote the expected utility that agent a performs α.

Agents always require sharable resources to perform tasks. These resources, $Res = \{res_1, \ldots, res_m\}$, are discrete and nonconsumable. Agent a has the exclusive access to resources $Holds(a) \subseteq Res$. Only one agent may hold a resource at any time; that is, $\forall a, b \in A$, $a \neq b$, $Holds(a) \cap Holds(b) = \emptyset$.

The teamwork is to maximize the reward for the team, while minimizing the *costs of coordination*. The overall reward is

$$\sum_{i=1}^{n} \sum_{a \in A} Assigned\,(a, \alpha_i)\, Reward \times (a, \alpha_i, Capability\,(a, \alpha)\,, Holds\,(a))\,. \tag{3}$$

The costs of coordination are very general and in some cases hard to be defined. Here we are mainly concerned with the volume of communication.

2.3. Coordination Decision Process. The objective of agents' interactions is to jointly coordinate themselves so that their common goal could be achieved. In large team coordination, similar to human society, agents always forward the incapable tasks and resources across the network. Once an agent accepts a task or resource according to its capability and what it is performing, it will execute the task or make use of the resource. The key of the coordination is to optimize their coordination so that the best capable agents could be reached to as fast so that agents' communication and assignment delay can be minimized.

Specifically, in our abstracted coordination simulator, initiated tasks or resources are encapsulated into messages; each agent executes Algorithm 1, which describes a general way of agents' coordination. Agents firstly check whether new tasks become applicable. If it is, the agent will encapsulate the task into a message and add it into its message queue so that the messages can be processed (lines 3–7). Next, the agent will merge all the messages passed from other agents to its message queue (line 8). It then processes all the messages in the queue. If a message represents a task, the agent will accept the task when its capability to perform that task is higher than message's threshold (lines 11–14); otherwise, the agent will choose a neighbor to pass that message to (line 17). If the message encapsulated a resource and the agent's need for that resource to perform its waiting tasks is higher than message's current threshold [13], this resource will be held; otherwise it is passed to a neighbor (lines 19–27). Note that when a message is sent, the message will be removed from that agent's list. Finally, the agent will check whether any pending tasks can now be executed (line 30) and release any resources from completed tasks (lines 32–37).

3. Coordination Efficiency over Different Network Topologies

In this section, we briefly analyze how team organization can make the team coordination performance different. We model the team coordination messages' routing over the network as a finite Markov chain, which is briefly illustrated as in Figure 1. For a specific message movement, we can define different states. In Figure 1, s_i defines the state that the message moves to an agent with the shortest distance of i to the sink agent. The transition probability $P_{i,j}$ defines that of the message being passed from state i to j. Because there is only one step move for each message at any horizon, $P_{i,j} = 0$ except for $j \in \{i-1, i, i+1\}$. Therefore, for a state $s_i \neq s_0$, the message may move closer to the destination ($P_{i,i-1}$), stay on the same level ($P_{i,i}$), or move far away ($P_{i,i+1}$) as the three statuses shown in Figure 2. When the message reaches state s_0, it will be kept at the destination and $P_{0,0} = 1$. If we suppose that u is the initial probability distribution of the message being in state s, according to the theory of Markov chains [14], the probability that the message reaches the sink agent after n steps can be calculated as

$$P_s^n = u \times P^n. \tag{4}$$

As the agents transmit the messages randomly to anyone of their neighbors, there will be different distances between the source and destination. Figure 2 shows the relative rates of $P(s_i, s_{i-1})$ (marked as "Close"), $P(s_i, s_i)$ (marked as "Same"), and $P(s_i, s_{i+1})$ (marked as "Further") for scale-free and random networks [6]. Notice that we average $P(s_i, s_j)$ over each node at distance i, though this will vary from node to node in different cases. The x-axis shows the distance from a node to the target node, that is, the target agent i. The y-axis shows the proportion of the states of "Further," "Same," and "Closer," and different areas represent the corresponding proportions with a sum of 100%. In general, the closer the agent is to the sink agent, the more likely the random movement is to lead the message further from it. Conversely, the further the agent is from the sink agent, the more likely the random movement is to lead the message closer to it. Figure 2 shows different complex network probability distributions with different messages' movement probabilities.

Figure 2(a) shows the messages' state probability transition matrix P with a typical scale-free network organization:

$$\begin{pmatrix} 0.02 & 0.98 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.01 & 0.1 & 0.89 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.05 & 0.25 & 0.7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.2 & 0.35 & 0.45 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0.25 & 0.25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.75 & 0.1 & 0.15 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.89 & 0.01 & 0.1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{5}$$

Figure 2(b) shows that, with a typical random network organization,

$$\begin{pmatrix} 0.01 & 0.99 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.01 & 0.03 & 0.96 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.01 & 0.24 & 0.75 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.15 & 0.5 & 0.35 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.6 & 0.25 & 0.15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.8 & 0.05 & 0.15 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.88 & 0.02 & 0.1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{6}$$

```
(1) ApplicableTasks = [], OwnTasks = [], Holds = [], Messages = [];
(2) while (true) do
(3)   for (α ∈ agent a_i & α ∉ ApplicableTasks) do
(4)     if (Applicable(α)) then
(5)       ApplicableTasks.append(α);
(6)       Messages.append(CreateMessages(α));
(7)     end if
(8)     Messages.append(recvMessages());
(9)   end for
(10)  for (m ∈ Messages) do
(11)    if (m is TaskMessages(α)) then
(12)      if (GetCap(α) > m.threshold) then
(13)        if (α ∉ OwnTasks) then
(14)          OwnTasks.append(α);
(15)        end if
(16)      else
(17)        SendToNeighbour(m);
(18)      end if
(19)    else if (m is ResourceMessages(r)) then
(20)      m.threshold += δ;
(21)      if (GetNeed(r) > m.threshold) then
(22)        if (r ∉ Holds) then
(23)          Holds.append(r);
(24)        end if
(25)      else
(26)        m.threshold −= δ;
(27)        SendToNeighbour(m);
(28)      end if
(29)    end if
(30)    CheckExecution(OwnTasks, Holds);
(31)  end for
(32)  for (α ∈ OwnTasks) do
(33)    if (α is complete) then
(34)      OwnTask.remove(α);
(35)      for (r ∈ ChkUnneed(OwnTask, Holds)) do
(36)        Hold.remove(r);
(37)        SendToNeighbour(CreateMessages(r));
(38)      end for
(39)    end if
(40)  end for
(41) end while
```

Algorithm 1: Coordination decision process.

Suppose that the same message's initial distribution is $u = [0.01\ 0.15\ 0.15\ 0.10\ 0.10\ 0.17\ 0.16\ 0.16]$ and after given steps of message's random movements, the state probability distribution is listed as in Table 1. For example, after 1000 steps, the state probability distribution for a scale-free network is $[0.858\ 0.003\ 0.016\ 0.050\ 0.056\ 0.016\ 0.001\ 0.000]$, where in 86% of cases this message has reached the sink agent. On the other hand, the state probability distribution for a random network after 1000 steps is $[0.653\ 0.003\ 0.020\ 0.106\ 0.182\ 0.036\ 0.000\ 0.000]$, where in only about 65% cases this message has reached the sink agent. The efficiency of information transmission in a scale-free network is significantly higher than in a random network.

Table 1: Random walk in complex networks.

	Steps	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7
Scale-free	100	0.310	0.012	0.08	0.244	0.272	0.078	0.004	0.0
	200	0.421	0.010	0.067	0.204	0.228	0.065	0.004	0.0
	500	0.658	0.006	0.040	0.121	0.135	0.039	0.002	0.0
	1000	0.858	0.003	0.016	0.050	0.056	0.016	0.001	0.0
Random	100	0.248	0.007	0.042	0.229	0.394	0.079	0.001	0.0
	200	0.310	0.006	0.039	0.211	0.362	0.073	0.001	0.0
	500	0.467	0.005	0.030	0.163	0.280	0.056	0.001	0.0
	1000	0.653	0.003	0.020	0.106	0.182	0.036	0.000	0.0

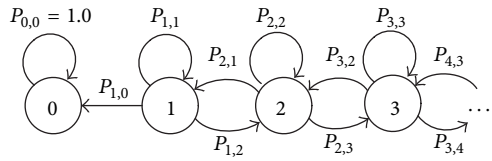


Figure 1: Markov chains model on messages' movement between agents.

4. Team Reorganization Approach

The theoretical analysis demonstrates that the organization structure of large-scale multiagent team does influence its performance efficiency. In this section, we will introduce our team adjusting approach to construct better team organization so that team coordination efficiency could be improved. Taking both the intrinsic characters of agents' coordination and the complex network effects into consideration, we have proposed an integrated adaptive algorithm and built two heuristic models to learn the agents' closely coordinative relationships.

Our previous study has found that communication efficiency of the team, which is measured by the number of hops a message goes across the team, will be improved if messages are forwarded down the link with higher probability towards the destination agent. Therefore, we can build a data structure to learn from the sampled team coordination performance data and transferred into the reorganization algorithm.

The basic process of our reorganization approach is described as in Figure 3. In the learning process, we will sample the interactions between agents from the coordination simulations and build two heuristic models: *connection matrix* and *subteam* to find closely coordinative relationships for each pair of agents. Next, by taking the complex network attributes, *connection matrix* and *subteam* models, into consideration, we build a probability model to measure the connection between each pair of agents. The high probability will be applied for agents who are closely cooperated, within a subteam, or promote the scale-free and small-world effects. Therefore, we can transform the probability model into an adaptive reorganization algorithm, where agents recursively pick their neighbors based on their probabilities.

4.1. Connection Matrix. The coordination between agents is somewhat similar to human society, in which groups of people closely coordinate and communicate according to their common interests. Therefore, tasks or resources always come from their source to the agents who are capable of performing the task or using the resource encapsulated. For example, a piece of information about finding a hostile tank always starts from an unmanned scout good at information gaining and is useful to the UAVs who can attack tanks. This cooperative relationship is an intrinsic character of the team and is independent of how the team is organized or how the coordination is delivered.

If the messages coming from the source can be delivered to their destinations with shorter paths, the coordination is improved. In addition, the pairs of the source and target agents are always fixed according to their characters such as their capabilities which have been predefined before the coordination. Therefore, if we can learn those partners who always closely coordinate, we are able to reorganize the team so that the average distance between those partners could be shortened.

According to this idea, we use a matrix called *source .target* to record messages' movements. Each element of $source.target[i][j]$ records the number of messages whose sender is agent a_i and receiver is agent a_j. In our learning process, when a message has been accepted (as shown in lines 11–15 and 21–24 in Algorithm 1), we are able to record its source (as a_i) and sink agent (as a_j) and $source.target[i][j]$ is accumulated by one. Based on the learned *source.target* matrix, it can be easily transformed into *connection* matrix:

$$connection\,[i]\,[j] = source.target\,[i]\,[j] + source.target\,[j]\,[i]\,. \tag{7}$$

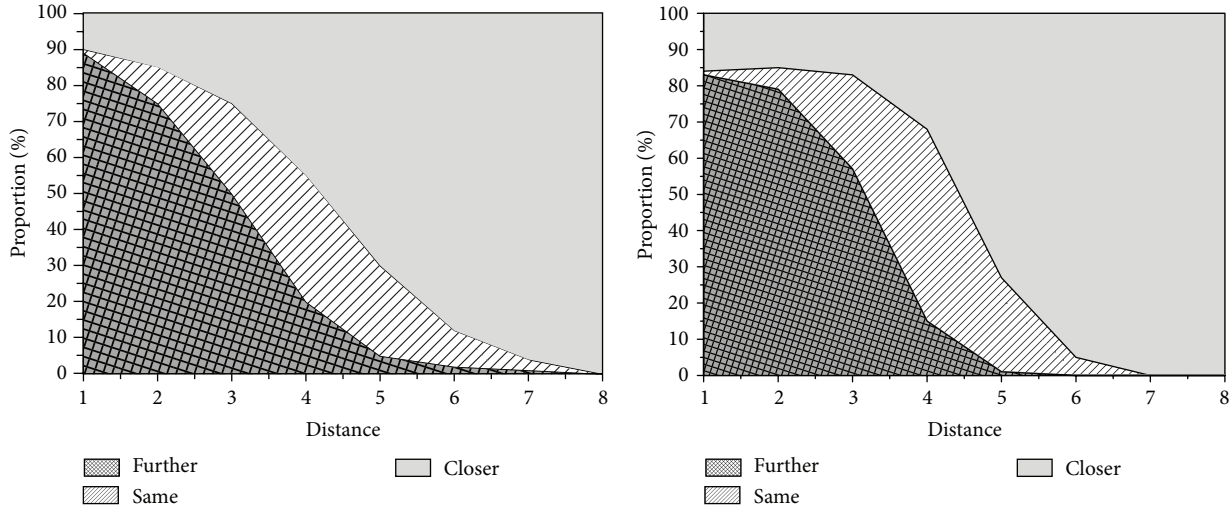
The *connection* matrix is symmetric and records the coordination between any pair of agents.

4.2. Subteams. Inspired by the clusters formed by closely cooperative individuals in human society, we build the *subteam* model to describe the similar group activities across the network. The *subteam* model, which is described in Figure 4 [11], is according to the mechanism of coordinative task planning. In the team coordination model we have built, the common goal is broken into subgoals $g_1, \ldots, g_i$, which can be executed by individual agents. Hence, agents can follow the planning mechanism to coordinate and achieve their subgoals.

Firstly, we predefined a number of plan templates in the library for agents to instantiate their plans. For example, when there is a fire in a building, the plan will be instantiated because it matches a template for disaster response. Each subgoal is addressed with a plan, $plan_i = \langle g_i, recipe_i, roles_i, d_i \rangle$, and thus the overall team plans, $Plans = \{plan_1, \ldots, plan_i\}$. g_i is the subgoal, and $recipe_i$ describes the way the subgoal will be achieved. $roles_i = \{r_1, \ldots, r_i\}$ are individual activities that must be performed to execute $recipe_i$, and d_i is the domain specific information pertinent to the plan.

Distributed plan creation is implemented by individual agents on behalf of the team, and we allow any member to commit the team to executing a plan when it detects that subgoal g_i is relevant. The subteams formation process commences when an individual agent detects all the appropriate preconditions that match a plan template in the library and subsequently instantiates a plan, $plan_i$. The $roles_i$ in $plan_i$ is embedded in the messages which are forwarded across the network until an agent finally accepts the role. Once accepted, the agent becomes a member of the subteam and makes a temporary commitment to perform the role toward the subgoal with the other subteam members. With the completion of the plan, the subteam will be dismissed so that new subteam for a new plan can be formed.

This algorithm can be described as an extended part of task allocation in team coordination. Algorithm 2 briefly



(a) In a scale-free network (b) In a random network

Figure 2: The relative proportions of links that lead closer to the sink agent, keep the same distance, or move further from the sink agent, as the distance to it is varied [6].

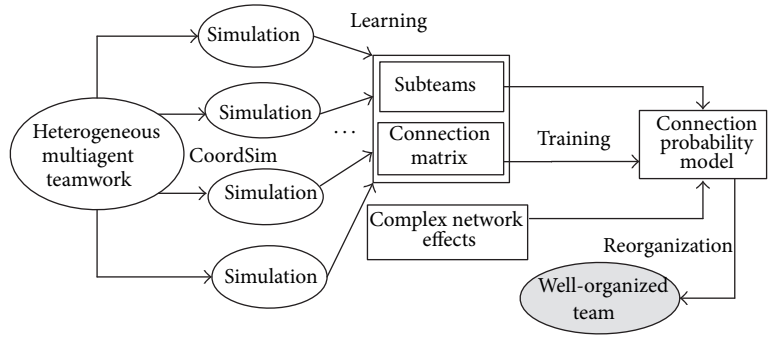


Figure 3: The team reorganization approach.

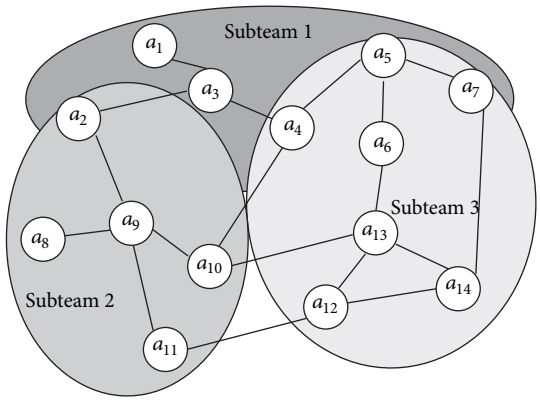


Figure 4: An example of the subteam model.

describes the formation of subteams. Although agents dynamically form and dismiss subteams, when agents form a subteam, they communicate frequently for their common interest. It should be learned and modeled in our adaptive algorithm so that the agents who were always in the same subteams should be connected closely to improve the team performance.

Based on the intrinsic character of the subteams, we consider it as an important factor for the team reorganization. In the coordination process, when agents use *JointIntentionMessage* to inform their neighbors to join a subteam, we record their subteam ID and mark it as $subID[i]$. Thus, we can learn the number of the subteams that agent i joined, written as $subteam[i]$. The overlapping subteam where agent i and agent j have joined together is written as $subteam[i, j]$. Therefore, the close relationship between i and j in the subteam model can be formalized as $sub[i][j]$:

$$sub\,[i]\,[j] = \frac{subteam\,[i, j] \times 2}{subteam\,[i] + subteam\,[j]}. \tag{8}$$

4.3. Integrated Reorganization Algorithm. In this section, we propose our adaptive reorganization algorithm. The key is to connect each pair of closely cooperated agents and promote

```
(1) for (m ∈ Messages) do
(2)   if (m is TaskMessages(α)) then
(3)     if (GetCap(α) > m.threshold) then
(4)       if (α ∉ OwnTasks) then
(5)         OwnTasks.append(α);
(6)         SendJointIntentionMessage(α.plan);
(7)         FormSubteams(α.plan);
(8)       end if
(9)     end if
(10)  end if
(11) end for
(12) for (α ∈ OwnTasks) do
(13)  if (α is complete) then
(14)    OwnTask.remove(α);
(15)    DismissSubteams(α.plan);
(16)  end if
(17) end for
```

Algorithm 2: Agents coordination (modified part).

```
(1) E = Φ;
(2) for (i = 1 to No_of_agents) do
(3)   for (j = 1 to Avg_degree/2) do
(4)     repeat
(5)       dest ← UniformRandom(Pr(i));
(6)     until Not(⟨i, dest⟩ ∈ E)
(7)     E.add(⟨i, dest⟩);
(8)   end for
(9) end for
(10) E.add(⟨i, dest⟩);
```

Algorithm 3: Integrated reorganization algorithm.

the helpful complex network attributes to design an integrated algorithm. In our algorithm, the closely coordinated relationships are defined according to the *connection matrix* and *subteams*. To build the integrated probability model, each probability of connecting agents a_i and a_j directly is written as $\Pr[i][j]$ ($\Pr[i][j] = \Pr[j][i]$). It is correlated with *connection matrix* and *subteams* according to the closely cooperative relationship, $d(j)$ according to the scale-free effect, and their distance distance(i, j) according to the small-world effect. Please note that we put a very small positive value ε as default in the probability model to guarantee that, although less likely, any agent is still able to directly connect with other agents. Specifically, we write $\Pr(i) = \{\Pr[i][j] \mid a_j \in A\}$ as the probability vector that a_i connects with any of the others.

The network reorganization algorithm is expressed as in Algorithm 3. In this algorithm, the network starts from empty link set (line 1). Each agent picks half of the predefined average degree of the network (line 3) and connects with them if they have not been connected (lines 5-6). The function *UniformRandom*$(\Pr(i))$ helps to pick agent a_i's neighbor based on its probability vector, which is the key to the

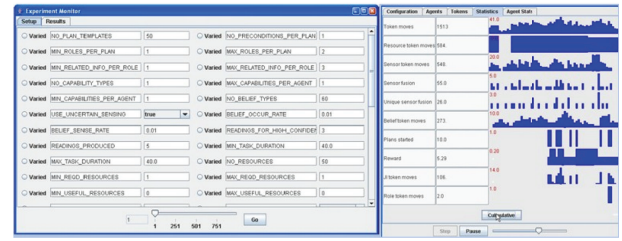


Figure 5: Screenshots of CoordSim simulator.

algorithm. In this paper, we briefly consider four key factors to build the probability $\Pr(i, j)$:

(i) $connection[i][j] / \sum_{a_k \in A} connection[i][k]$ models the preference of agent i directly connecting j according to the connection matrix;

(ii) $sub[i][j] / \sum_{a_k \in A} sub[i][k]$ models the preference of agent i directly connecting j according to the *subteam* model;

(iii) $d(j) / \sum_{a_k \in A} d(k)$ models j's degree distribution to infer whether i should connect to j to promote scale-free effect. The higher the degrees of j are, the more likely the i will connect to j;

(iv) $distance(i, j) / \sum_{a_k \in A} distance(i, k)$ models the small word effects. The further the distance between i and j is, the more likely they will be directly connected.

If agent a_i starts a new connection, the probability of connecting with a_j is defined as

$$\begin{aligned}\Pr(i, j) &= \beta \times \left(\frac{connection[i][j]}{\sum_{a_k \in A} connection[i][k]} + \frac{sub[i][j]}{\sum_{a_k \in A} sub[i][k]} \right) \\ &\quad + \gamma \times \frac{d(j)}{\sum_{a_k \in A} d(k)} + \lambda \times \frac{distance(i, j)}{\sum_{a_k \in A} distance(i, k)},\end{aligned} \tag{9}$$

where normalization should be applied. $\{\beta, \gamma, \lambda\}$ are the important factors and $\beta + \gamma + \lambda = 1$. Please note that if $\{\beta = 1, \gamma = 0, \lambda = 0\}$, we only take the character of closely coordinative relationships into consideration; if $\{\beta = 0, \gamma = 1, \lambda = 0, \}$, we will set up a standard scale-free network; and if $\{\beta = 0, \gamma = 0, \lambda = 1\}$, we will set up a standard small-world network [12].

5. Simulations and Results

To simulate the real team coordination, we use our abstract simulator called CoordSim [15]. This simulator is capable of simulating the major aspects of coordination, including task assignment and resource allocation. CoordSim abstracts the environment by simulating only its effects on the team. According to the team coordination process in Section 2.3, reward is simulated as being received by the team when a task is allocated. CoordSim allows a large number of parameters

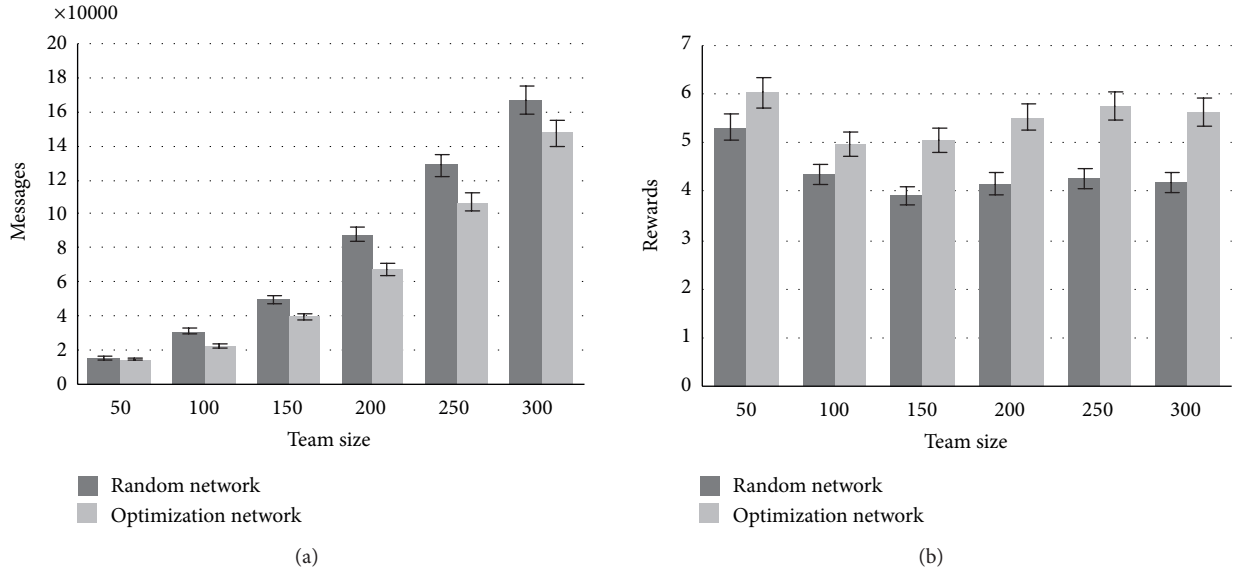


Figure 6: The reorganization algorithm is scalable on different sizes of teams.

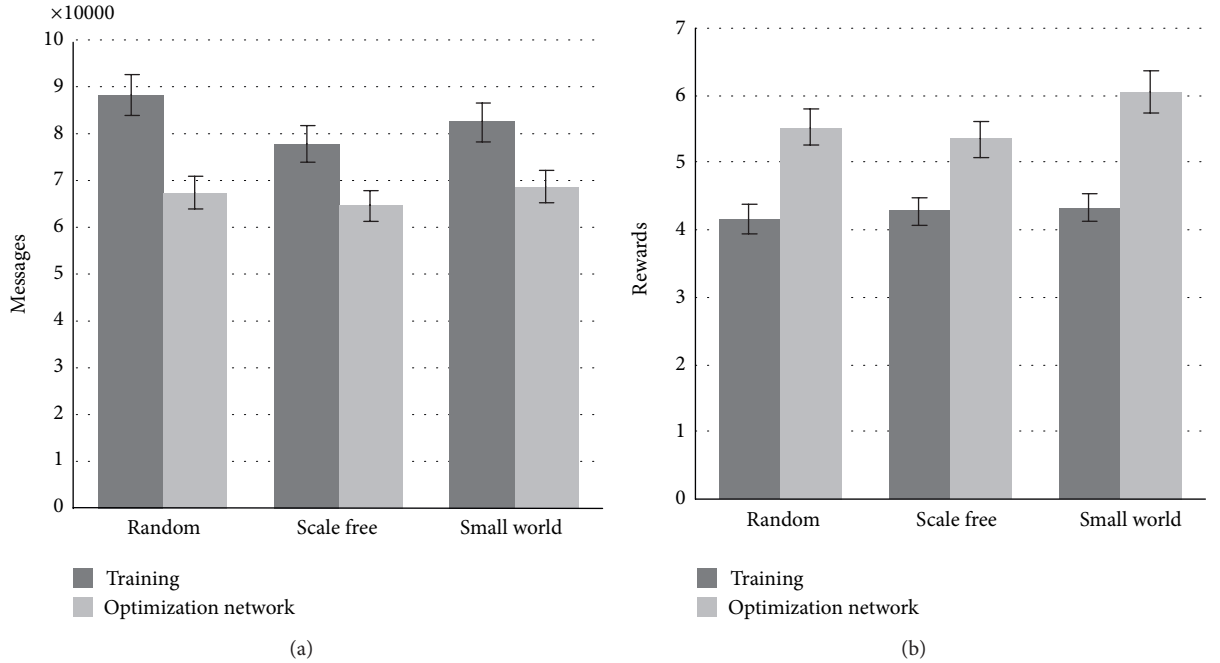


Figure 7: The reorganization algorithm improves the team performances when the teams originally organized as different network structures.

to be varied and also allows statistics to be recorded, such as the number of rewards and message movements, which is important to our approach. Two interfaces of this simulator are shown in Figure 5. There are more than 20 parameters that can be varied, covering the major aspects of the large heterogeneous coordination.

If not otherwise stated, the experiments are configured as follows. There are 100 agents deployed in a 500 × 500 environment to perform 100 tasks with 100 resources. Each task requires at least one resource. In the default setup, the heterogeneous team has five different types of capabilities. Task and resource messages are allowed to move unless accepted. "Reward" is the sum of the rewards received by each agent. "Messages" is the number of times that agents communicated for coordination. The objective of the team coordination is to gain the best tradeoff between maximizing team rewards and minimizing messages. To build intelligent coordination between agents, we used the literature [13] as the coordination schema in the simulations. Simulation will last for 500 time steps. All of the experimental results are based

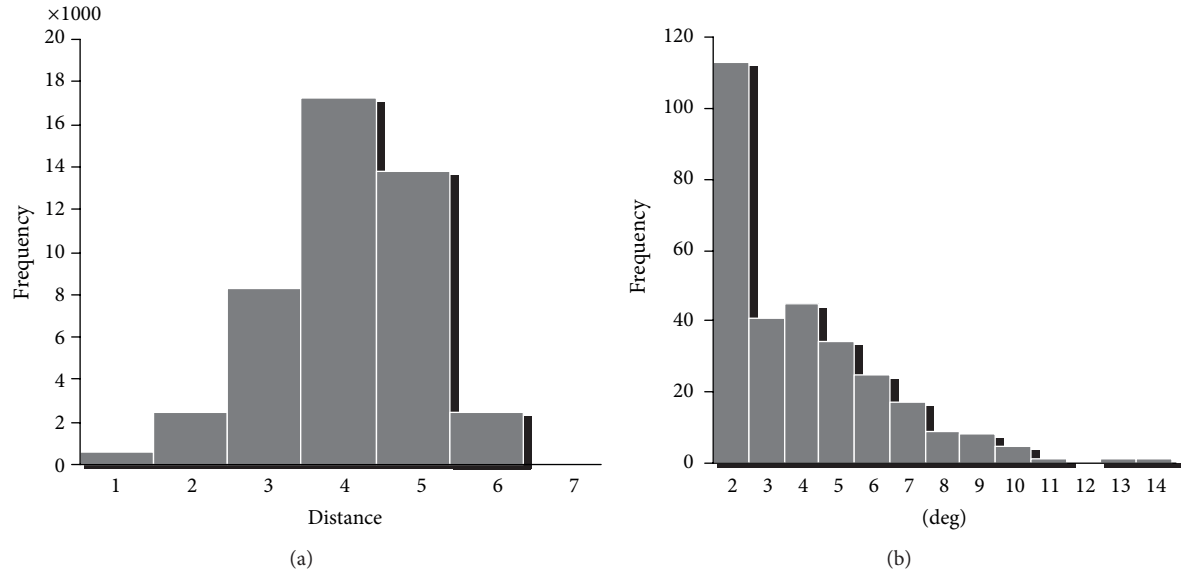


Figure 8: After reorganization, the team has small-world and scale-free effects.

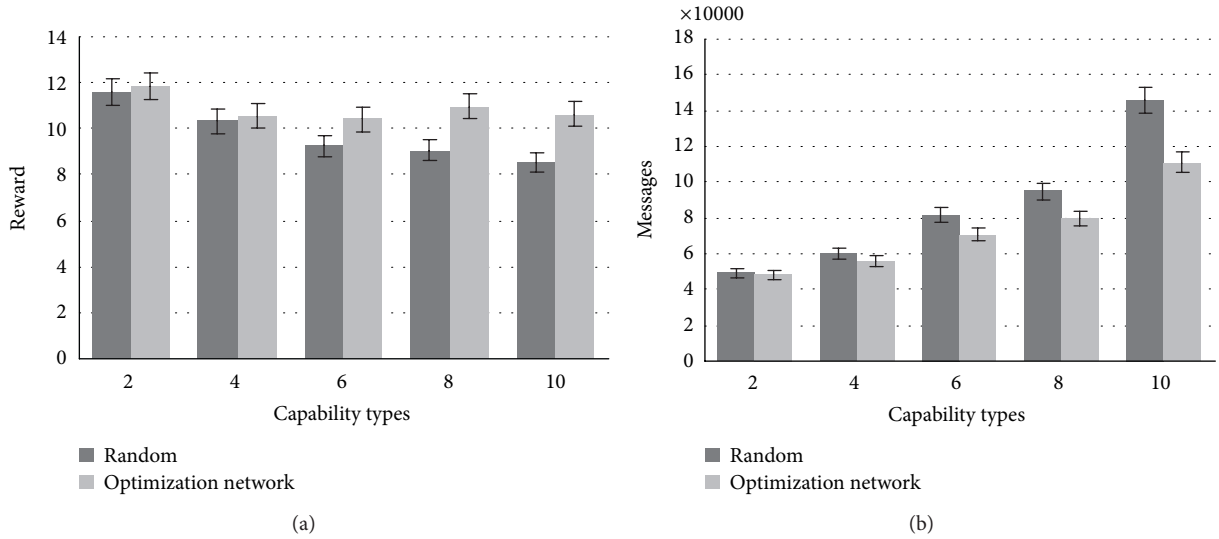


Figure 9: The reorganization algorithm improves the heterogeneous teams more significantly.

on 100 runs. To build the *connection matrix* and *subteam*, we sampled the team with 100 runs as well. In each run, all the agents and environmental settings keep the same, but their organization networks are different.

We evaluated the efficiency of our team reorganization algorithm in five experiments. In the first experiment, we first organized the team as random networks and set $\{\beta = 1, \gamma = 0, \lambda = 0\}$ that the team should be reorganized according to the *connection matrix* and *subteam* only. We varied the team size from 50 to 300, and to be fairly compared, tasks and resources in large teams were more to keep the same tasks per agent and resources per agent. The experimental results in Figure 6 show that, no matter what the team sizes are, the reorganized network outperforms the random team organization with higher rewards and less communication costs.

In the second experiment shown in Figure 7, we organized the original network with three different structures: random, small world, and scale free. The team size is 100 and we set $\{\beta = 1, \gamma = 0, \lambda = 0\}$ that the team will be reorganized according to the connection matrix and subteam model only. As we expected, sine we have taken the team's intrinsic closely cooperative relationship between heterogeneous agents into consideration, the team performance is greatly improved no matter what the original network is.

In Figure 8, we investigate whether the reorganized networks have the complex network attributes. The reorganized

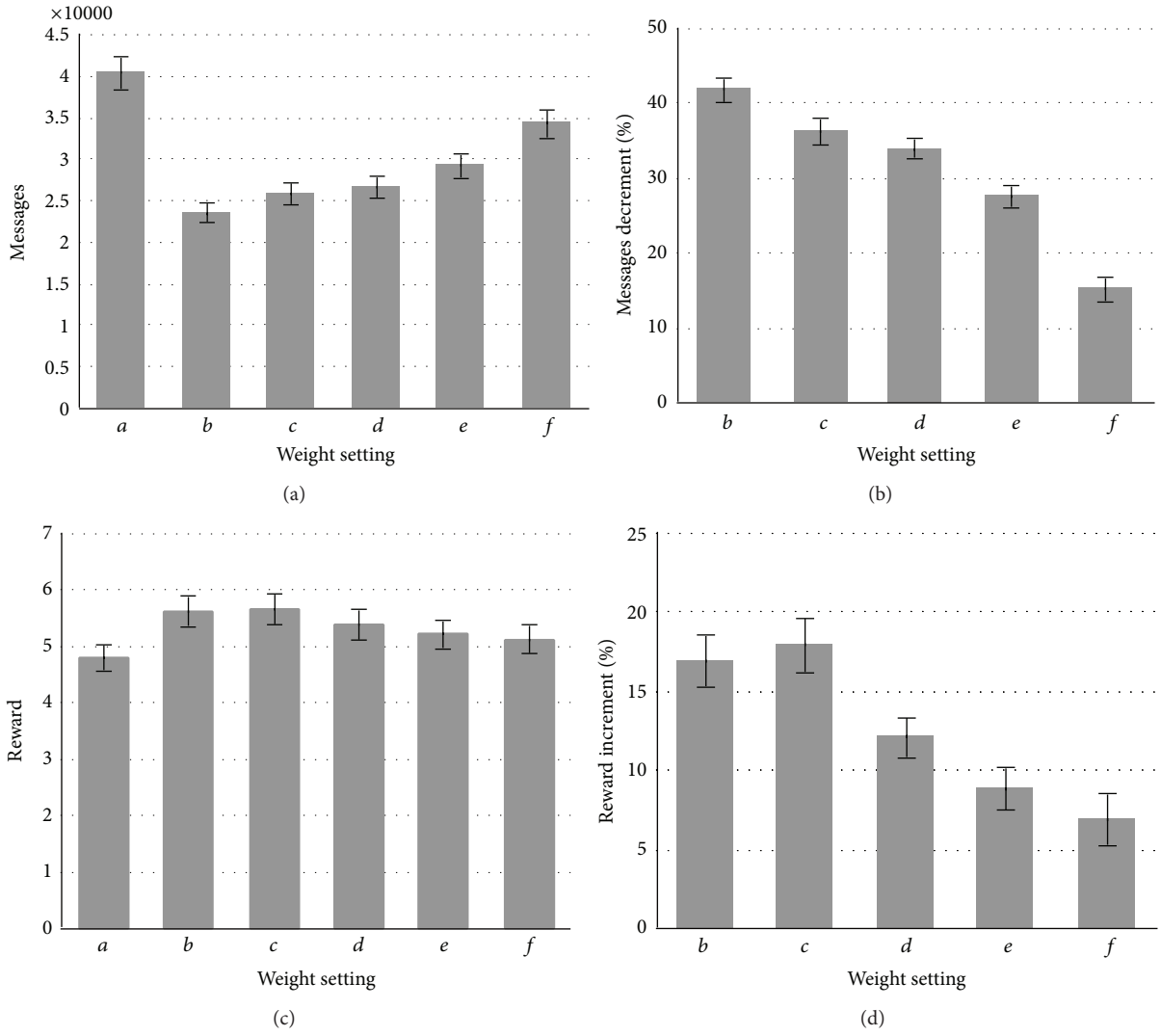


Figure 10: The integrated algorithm improves the team performance when β is higher than the others.

network is learned from a random network and the team size is 100. We set $\{\beta = 1, \gamma = 0, \lambda = 0\}$ as well. The distribution of distance and degree is shown in Figure 8. We can see that the reorganized network has the small-world and scale-free effects.

In the fourth experiment, we verify our design in different heterogeneous teams. The team size is 100 and the original network is random network. We set $\{\beta = 1, \gamma = 0, \lambda = 0\}$ as before. Agents' capability types are set from 2 to 10 to make the team more and more heterogeneous. Note that when team becomes more heterogeneous, less agents are capable of accepting a given task message. It will make the coordination very hard. As the experiment results in Figure 9 show, the reward gains keep decreasing in the original random networks; however, it is not the case in our reorganized networks. We hypothesize that when team becomes more heterogeneous, the closely cooperative relationship is more prominent to be caught, which makes our design more efficient.

In the last experiment, we set up the original network as random networks and team size is 100. We have six settings: $a = \{\beta = 0.0, \gamma = 0.0, \lambda = 0.0\}$, $b = \{\beta = 1.0, \gamma = 0.0, \lambda = 0.0\}$, $c = \{\beta = 0.7, \gamma = 0.02, \lambda = 0.28\}$, $d = \{\beta = 0.5, \gamma = 0.05, \lambda = 0.45\}$, $e = \{\beta = 0.3, \gamma = 0.05, \lambda = 0.65\}$, and $f = \{\beta = 0, \gamma = 0.05, \lambda = 0.95\}$ so that the team organization is set according to all the three factors, but with higher portions to small-world and scale-free effects. Experimental results in Figure 10 show that, comparing with a which is not reorganized, by integrating all three factors, the team performance with reorganization is improved. However, to gain the best performance, $c = \{\beta = 0.7, \gamma = 0.02, \lambda = 0.28\}$ works the best. When $f = \{\beta = 0, \gamma = 0.05, \lambda = 0.95\}$, the performance is worse with the lack of closely cooperative relationship in organization. We explain that little portions

of complex network effects help the team, but discovering the closely cooperative relationship contributes the most.

6. Related Work

Many researchers have demonstrated that the organizational design in multiagent systems has a significant effect on its performances [16], and the properties of small-world network can enhance network's signal propagation speed, computational power, and synchronization [17]. Glinton et al. [18] have found that the team is of high performance and rapid convergence when the scale-free organization was formed, and limited links per agent in the complex network improve team performance.

A range of organizational strategies have been proposed to improve multiagent team. In literature [19], it presented a distributed scenario for team formation in multiagent systems and concluded that the direct interconnections among agents were determined by the agent interaction topology. Kota et al. [20] provided a self-organization method which enabled agents to modify the structural relations to achieve a better allocation of tasks. Keogh and Sonenberg [21] designed a flexible, coordinated organization-based agent system in which agents can adjust their own attitudes to fit in others in a changing situation by having the access to organizational information that they can change. A composite self-organization mechanism in a multiagent network is proposed in [22]. It enables agents to dynamically adapt team organization by using a trust model and the former task allocation to assist agents to decide whom they should connect with. Terauchi et al. [23] proposed an agent organization management system and provided context based organizational information for problem solving. It is only concerned with improving system scalability and flexibility but neglected the coordination efficiency.

7. Conclusion and Future Works

In this paper, we have made an initial effort on finding how to adjust team organization to expedite team performance. We have proved that team organization is a key factor to the team performance. Based on scalable team coordination schema, we designed an adaptive team organization algorithm by incorporating the nature of agents' cooperation relationship and the attributes of complex network. Our experiments have been proven to the validity of our design.

While this work represents an important step in this regard, much work remains to be done. First of all, we have only designed an offline learning algorithm to adjust the team, while online adjustment may be available. Secondly, our approach is based on the organization of associated network which is based on a P2P coordination infrastructure, while in many application domains, the coordination is based on broadcast.

Conflict of Interests

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

Acknowledgments

This research has been sponsored by NSFC 61370151, 60905042, and 61202211, National Science and Technology Support Program of China 2012BAI22B05, and Central University Basic Research Funds Foundation ZYGX2011X013.

References

[1] J. Han, C. Wang, and G. Yi, "Cooperative control of UAV based on multi-agent system," in *Proceedings of the 8th IEEE Conference on Industrial Electronics and Applications*, 2013.

[2] T. Yang, M. Wang, and M. Fang, "The military strategy threat of cognitive system based on agent," in *Proceedings of the International Conference of Modern Computer Science and Applications Advances in Intelligent Systems and Computing*, vol. 191, 2013.

[3] M. Mala, "RCAIMS: a reactive multi-agent system based incident/emergency management system," *International Journal of Computational Intelligence Studies*, vol. 2, no. 3-4, pp. 300–313, 2013.

[4] V. Gorodetsky, O. Karsaev, V. Samoylov, and S. Serebryakov, "P2P agent platform: implementation and testing," in *Agents and Peer-to-Peer Computing*, vol. 5319 of *Lecture Notes in Computer Science*, pp. 41–54, Springer, Berlin, Germany, 2010.

[5] M. E. Gaston and M. DesJardins, "Agent-organized networks for dynamic team formation," in *Proceedings of the 4th International Conference on Autonomous Agents and Multi agent Systems (AAMAS '05)*, pp. 375–382, July 2005.

[6] P. Scerri and K. Sycara, "Social networks for effective teams," *Cooperative Networks: Control and Optimization*, 2008.

[7] Y. C. Jiang and J. C. Jiang, "Understanding social networks from a multi-agent coordination perspective," *IEEE Transactions on Parallel and Distributed Systems*, 2013.

[8] J. Pitt, D. Ramirez-Cano, M. Draief, and A. Artikis, "Interleaving multi-agent systems and social networks for organized adaptation," *Computational and Mathematical Organization Theory*, vol. 17, no. 4, pp. 344–378, 2011.

[9] Q. Han, T. Arentze, H. Timmermans, D. Janssens, and G. Wets, "The effects of social networks on choice set dynamics: Results of numerical simulations using an agent-based approach," *Transportation Research A: Policy and Practice*, vol. 45, no. 4, pp. 310–322, 2011.

[10] Y. Jiang, Y. Zhou, and W. Wang, "Task allocation for undependable multiagent systems in social networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 8, pp. 1671–1681, 2013.

[11] E. Liao, P. Scerri, and K. Sycara, "A framework for very large teams," in *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems Workshop on Coalitions and Teams*, 2004.

[12] R. Albert, H. Jeong, and A. L. Barabási, "Diameter of the world-wide web," *Nature*, vol. 401, no. 6749, pp. 130–131, 1999.

[13] Y. Xu, P. Scerri, B. Yu, S. Okamoto, M. Lewis, and K. Sycara, "An integrated token-based algorithm for scalable coordination," in *Proceedings of the 4th International Conference on Autonomous Agents and Multi agent Systems*, pp. 543–550, July 2005.

[14] L. Lovasz, "Random walks on graphs: a survey," in *Combinatorics, Bolyai Mathematical Society*, 1993.

[15] Y. Xu, P. Scerri, K. Sycara, and M. Lewis, "Comparing market and token-based coordination," in *Proceedings of the 5th*

International Conference on Autonomous Agents and Multiagent Systems, pp. 1113–1115, May 2006.

[16] S. van Segbroeck, S. de Jong, A. Nowé, F. C. Santos, and T. Lenaerts, "Learning to coordinate in complex networks," *Adaptive Behavior*, vol. 18, no. 5, pp. 416–427, 2010.

[17] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[18] R. Glinton, K. Sycara, and P. Scerri, "Agent organized networks redux," in *Proceedings of the AAAI Conference on Artificial Intelligence and the 20th Innovative Applications of Artificial Intelligence Conference*, pp. 83–88, July 2008.

[19] L. Coviello and M. Franceschetti, "Distributed team formation in multi-agent systems: stability and approximation," in *Proceedings of the 51st IEEE Conference on Decision and Control*, 2012.

[20] R. Kota, N. Gibbins, and N. R. Jennings, "Self-organising agent organisations," in *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems*, 2009.

[21] K. Keogh and L. Sonenberg, "Adaptive coordination in distributed and dynamic agent organizations," in *Coordination, Organizations, Institutions, and Norms in Agent System VII*, vol. 7254 of *Lecture Notes in Computer Science*, pp. 38–57, Springer, Berlin, Germany, 2012.

[22] D. Ye, M. Zhang, and D. Sutanto, "Self-organization in an agent network: a mechanism and a potential application," *Decision Support Systems*, vol. 53, no. 3, pp. 406–417, 2012.

[23] A. Terauchi, O. Akashi, M. Maruyama et al., "ARTISTE: agent organization management system for multi-agent systems," in *Multi-Agent Systems for Society*, vol. 4078 of *Lecture Notes in Computer Science*, pp. 207–221, Springer, Berlin, Germany, 2009.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 946560, 8 pages
http://dx.doi.org/10.1155/2014/946560

Research Article

Pinning Lur'e Complex Networks via Output Feedback Control

Fang Liu,[1] Qiang Song,[1] Jinde Cao,[2,3] and Jianquan Lu[2]

[1] *School of Information Engineering, Huanghuai University, Henan 463000, China*
[2] *Department of Mathematics, Southeast University, Nanjing 210096, China*
[3] *Department of Mathematics, Faculty of Science, King Abdulaziz University, Jeddah 21589, Saudi Arabia*

Correspondence should be addressed to Qiang Song; qsongseu@gmail.com

Received 6 February 2014; Accepted 5 March 2014; Published 13 April 2014

Academic Editor: Guanghui Wen

Without requiring the full-state information of network nodes, this paper studies the pinning synchronization in a network of Lur'e dynamical systems based on the output feedback control strategy. Some simple pinning conditions are established for both undirected and directed Lur'e networks by using M-matrix theory and S-procedure technique. With the derived stability criteria, the pinning synchronization problem of large-scale Lur'e networks can be transformed to the test of a low-dimensional linear matrix inequality. Some remarks are further given to address the selection of pinned nodes and the design of pinning feedback gains. Numerical results are provided to demonstrate the effectiveness of the theoretical analysis.

1. Introduction

Lur'e systems refer to class of nonlinear dynamical systems which are formed by a linear system and a nonlinear feedback loop satisfying a sector condition. Actually, many nonlinear systems, such as Chua's circuit system and some hyperchaotic systems, can be described by Lur'e systems. Over the past few decades, much effort has been devoted to the research on Lur'e systems. The stability problem of Lur'e systems has been intensively studied, yielding some fundamental results such as the circle and Popov criteria [1–6]. The synchronization problem of Lur'e systems has also been deeply investigated for the purpose of secure communication and engineering applications [7–12].

Recently, the synchronization phenomena in complex dynamical networks with each node being a Lur'e system have attracted increasing attention. Liu et al. [13] addressed the global synchronization in coupled Lur'e systems based on the absolute stability theory. Ding and Han [14] studied the effect of the communication delay on the synchronization in Lur'e networks. Ji et al. [15] considered the synchronization problem for complex networks composed of time-delayed Lur'e systems. It is worth noting that the Lur'e networks in [13–15] achieved synchronization by local interactions among network nodes without involving any external force. When a Lur'e network cannot reach synchronization by itself, some appropriate controllers may be designed such that the entire network can be synchronized to some desired trajectory.

For a complex network with a large number of nodes, it is literally impossible to apply control actions to all nodes due to the high control cost. Current studies have shown that the pinning control strategy [16–22] can be utilized to synchronize a network to a homogenous state, where local feedback injections are only placed on a small fraction of network nodes. The pinning control problem for Lur'e networks with undirected topologies has been investigated by some researchers [23–25]. More recently, Song et al. [26] developed some simple pinning conditions for Lur'e complex networks with directed topologies by using M-matrix [27, 28] and algebraic graph theories.

In the pinning control of complex networks, how to choose a set of pinned nodes is one of the most difficult problems. For an undirected and connected network, it is well-known that the network can be synchronized if a subset of nodes is either specifically or randomly pinned [16, 17]. As for a directed network, it has been proved that pinning control should be applied to the roots of a minimum number of directed spanning forests of the network topology [18, 19, 21]. Song and Cao [22] showed that the nodes whose out-degrees are bigger than their in-degrees must be chosen

as pinned candidates. More recently, by using M-matrix theory, Song et al. [26, 29, 30] and Wen et al. [31] derived some stability criteria for pinning networked systems with directed topologies. In particular, some M-matrix strategies were developed to discuss several challenging problems in the pinning control of networked systems [30].

In most literature, the synchronization of complex networks is usually reached by using the full states of network nodes which may not always be available in many practical cases [32–35]. Note that some observer-based algorithms have been developed for achieving synchronization in complex networks [36, 37]. In this paper, we investigate the pinning synchronization of Lur'e complex networks by utilizing the observed states, that is, output states, of network nodes. Without requiring full state information of network nodes, we propose a distributed output feedback control approach to pin Lur'e networks with node dynamics satisfying sector conditions. Then, by using M-matrix and algebraic graph theories, we derive some simple stability criteria to convert the pinning control problem of Lur'e networks into the test of a linear matrix inequality whose dimension is just determined by a single Lur'e node. Moreover, we discuss the selection of pinned nodes and the design of pinning feedback gains for both undirected and directed Lur'e complex networks.

The rest of this paper is organized as follows. In Section 2, some preliminaries are provided. Section 3 formulates the pinning control problem of Lur'e networks with output feedback coupling. Sections 4 and 5 derive some pinning conditions for undirected and directed Lur'e networks, respectively. In Section 6, numerical results are given to validate the theoretical analysis. Finally, some concluding remarks are stated in Section 7.

2. Preliminaries

In this section, we provide some mathematical preliminaries and some supporting lemmas to derive the main results of this paper.

2.1. Notations. The standard notations are used throughout this paper. Let $\mathbb{R}$ and $\mathbb{C}$ denote the sets of real and complex numbers, respectively. For $z \in \mathbb{C}$, $\mathrm{Re}(z)$ represents its real part. Let I_n be the n-dimensional identity matrix and let $1_n \in \mathbb{R}^n$ ($0_n \in \mathbb{R}^n$) be the vector of all ones (zeros). For a matrix $A \in \mathbb{R}^{n\times n}$, let A^T be its transpose, A^{-1} its inverse, $A_s = (A + A^T)/2$ its symmetric part, and $\lambda_i(A)$ the ith eigenvalue, and let $\mathcal{J}(A) = \min_{1\le i\le n} \mathrm{Re}(\lambda_i(A))$ denote the minimum real part of all its eigenvalues. For a real symmetric matrix $X \in \mathbb{R}^{n\times n}$, let $\lambda_{\min}(X)$ and $\lambda_{\max}(X)$ be its minimum and maximum eigenvalues, respectively, and write $X > 0$ ($X < 0$) if X is positive (negative) definite. The symbol $\otimes$ denotes the Kronecker product [27].

2.2. Graph Theory. The information interaction in a networked system can be described by a weighted graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$ which is composed of a node set $\mathcal{V} = \{1, \ldots, N\}$, an edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, and an adjacency matrix $\mathcal{A}$. A directed edge between two different nodes is denoted by (j, i) indicating that node i can access the information from node j. The (i, j)th entry of the adjacency matrix $\mathcal{A}$ is defined by $a_{ij} > 0 \Leftrightarrow (j, i) \in \mathcal{E} (i \neq j)$ [29–31, 38, 39]. In this paper, it is always assumed that $a_{ii} = 0$ for all $i \in \mathcal{V}$. The elements of the Laplacian matrix $L = (l_{ij})_{N\times N}$ associated with the adjacency matrix $\mathcal{A}$ are defined as follows:

$$l_{ij} = -a_{ij} \le 0 \quad \text{for } i \neq j, \qquad l_{ii} = \sum_{k=1, k\neq i}^{N} a_{ik}. \tag{1}$$

A directed path is a sequence of directed edges with distinct nodes. A digraph with N nodes is called a directed tree if it has $N - 1$ edges and there exists a node with directed paths to every other node. A digraph is called strongly connected if for any two different nodes j and i, one can always find a directed path from node j to node i. A digraph is said to have or contain a directed spanning tree if there exists at least one node having a directed path to every other node [29–31, 38, 39].

2.3. M-Matrix Theory. Some results related to M-matrix will be used to study the pinning synchronization of Lur'e complex networks.

Definition 1 (see [27, 28]). A nonsingular matrix $A = (a_{ij}) \in \mathbb{R}^{n\times n}$ is called an M-matrix if $a_{ij} \le 0$ whenever $i \neq j$ and all elements of A^{-1} are nonnegative.

Lemma 2 (see [27, 28]). *For a nonsingular matrix $A = (a_{ij})_{n\times n} \in \mathbb{R}^{n\times n}$ with $a_{ij} \le 0$ $(i \neq j)$, the following statements are equivalent:*

(1) *A is an M-matrix;*

(2) *all eigenvalues of A have positive real parts; that is, $\mathrm{Re}(\lambda_i(A)) > 0$ for all $i = 1, \ldots, n$;*

(3) *there exists a positive definite diagonal matrix $\Xi = \mathrm{diag}(\xi_1, \ldots, \xi_n) > 0$ such that $\Xi A + A^T \Xi$ is positive definite.*

2.4. Some Supporting Lemmas. Some properties of the Kronecker product are listed as follows.

Lemma 3 (see [27]). *For matrices A, B, C, and D with appropriate dimensions, one has*

(1) $(A \otimes B)^T = A^T \otimes B^T$;

(2) $(A + B) \otimes C = A \otimes C + B \otimes C$;

(3) $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$.

Lemma 4 (see [40]). *Let $A \in \mathbb{R}^{n\times n}$ be symmetric. One has $\lambda_{\min}(A)x^T x \le x^T Ax \le \lambda_{\max}(A)x^T x$, forall $x \in R^n$.*

The S-procedure technique [1] which is widely used to study the stability problem of Lur'e dynamical systems can be stated as follows.

Lemma 5 (see [1]). *Let $V_0(x), V_1(x), \ldots, V_m(x)$ be quadratic forms over $x \in \mathbb{R}^n$. One has $V_0(x) < 0$ for all $x \neq 0_n$ satisfying*

$V_i(x) \le 0, i = 1 \ldots, m$, *if there exist some nonnegative numbers* $\tau_1, \ldots, \tau_m \ge 0$ *such that* $V_0(x) - \sum_{i=1}^{m} \tau_i V_i(x) < 0$.

Lemma 6 (Schur complement [1]). *The following linear matrix inequality (LMI)*

$$\begin{pmatrix} Q(x) & S(x) \\ S(x)^T & R(x) \end{pmatrix} > 0, \tag{2}$$

where $Q(x) = Q(x)^T$, $R(x) = R(x)^T$ *is equivalent to either of the following conditions:*

(1) $Q(x) > 0$, $R(x) - S(x)^T Q(x)^{-1} S(x) > 0$;
(2) $R(x) > 0$, $Q(x) - S(x) R(x)^{-1} S(x)^T > 0$.

3. Output Feedback Control Algorithm for Pinning Lur'e Networks

Consider a complex network with each node being a Lur'e dynamical system described by

$$\begin{aligned} \dot{x}_i(t) &= Ax_i + Bf(Cx_i), \\ y_i(t) &= Cx_i(t), \quad i = 1, \ldots, N, \end{aligned} \tag{3}$$

where $x_i = (x_{i1}, \ldots, x_{in})^T$ is the state variable of the ith node, $y_i = (y_{i1}, \ldots, y_{im})^T$ is the output state of the ith node, and $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{m\times n}$ with C_k^T being its kth row; the nonlinear vector-valued function $f(Cx_i) = (f_1(C_1^T x_i), \ldots, f_m(C_m^T x_i))^T \in \mathbb{R}^m$ satisfies the following sector conditions:

$$0 \le \frac{f_k(b) - f_k(a)}{b - a} \le \delta_k, \quad \forall a, b \in \mathbb{R},\ a \ne b,\ k = 1, \ldots, m. \tag{4}$$

In many practical cases, the full states of network nodes are not always available. By utilizing the output states of network nodes, we consider the following Lur'e network model:

$$\begin{aligned} \dot{x}_i(t) &= Ax_i + Bf(Cx_i) - \sigma \sum_{j=1}^{N} l_{ij} F y_j + u_i, \\ y_i(t) &= Cx_i(t), \quad i = 1, \ldots, N, \end{aligned} \tag{5}$$

where $\sigma > 0$ is the coupling strength, l_{ij} is the (i, j)th entry of Laplacian matrix $L \in \mathbb{R}^{N\times N}$, $F \in \mathbb{R}^{n\times m}$ is the output feedback gain matrix, and u_i is the control input to be designed.

Note that the isolated node (or leader node) for Lur'e complex network (5) is given by

$$\begin{aligned} \dot{x}_r(t) &= Ax_r(t) + Bf(Cx_r), \\ y_r(t) &= Cx_r(t), \end{aligned} \tag{6}$$

where $x_r = (x_{r1}, \ldots, x_{rn})^T$ and $y_r = (y_{r1}, \ldots, y_{rm})^T$.

Definition 7. The Lur'e complex network (5) is said to be globally asymptotically synchronized to the isolated node (6); that is, $x_i(t) \to x_r(t)$, $i = 1, \ldots, N$, as $t \to \infty$, for any initial condition.

To reduce the number of controllers, one can adopt the pinning control strategy to synchronize complex network (5) to the isolated node (6). For Lur'e network (5), let $\mathcal{V} = \{1, \ldots, N\}$ and $\mathcal{V}_{\text{pin}} = \{i_1, \ldots, i_l\} \subset \mathcal{V}$ be the sets of total and pinned nodes, respectively, where $1 \le l < N$. Applying pinning control to network (5) yields

$$\begin{aligned} \dot{x}_i(t) &= Ax_i + Bf(Cx_i) - \sigma \sum_{j=1}^{N} l_{ij} F y_j - \sigma d_i F(y_i - y_r), \\ y_i(t) &= Cx_i(t), \quad i = 1, \ldots, N, \end{aligned} \tag{7}$$

where the pinning feedback gains are defined as follows:

$$d_i > 0 \quad \text{if } i \in \mathcal{V}_{\text{pin}}, \qquad d_i = 0 \quad \text{if } i \in \mathcal{V} \setminus \mathcal{V}_{\text{pin}}. \tag{8}$$

Suppose that the matrix pair (A, C) in complex network (7) is *detectable*, and the output feedback gain matrix is designed as $F = P^{-1}C^T$, where $P \in \mathbb{R}^{n\times n}$ is a positive definite matrix to be determined later. Then the pinning-controlled Lur'e network (7) becomes

$$\begin{aligned} \dot{x}_i(t) &= Ax_i(t) + Bf(Cx_i) - \sigma \sum_{j=1}^{N} l_{ij} P^{-1} C^T C x_j \\ &\quad - \sigma d_i P^{-1} C^T C (x_i - x_r), \\ y_i(t) &= Cx_i(t), \quad i = 1, \ldots, N. \end{aligned} \tag{9}$$

Remark 8. If the matrix pair (A, C) is detectable, there always exists a matrix $F \in \mathbb{R}^{n\times m}$ to ensure that $A - FC$ is a Hurwitz matrix. Moreover, one can always find a positive definite matrix $P \in \mathbb{R}^{n\times n}$ and a positive scalar $\tau > 0$ such that $PA + A^T P - \tau C^T C < 0$ [1].

Remark 9. In most literature, the pinning control approaches for complex networks usually require full states of network nodes [18–22, 29, 30]. Note that the pinning control algorithm (9) is actually implemented by utilizing the output states of network nodes.

For the convenience of our discussions in this paper, let L be the Laplacian matrix of network (5), $D = \text{diag}(d_1, \ldots, d_N)$ the matrix of pinning feedback gains defined by (8), and $\Delta = \text{diag}(\delta_1, \ldots, \delta_m)$ the matrix describing the sector condition (4).

Let $e_i(t) = x_i(t) - x_r(t)$ and $\eta_i = (\eta_{i1}, \ldots, \eta_{im})^T = f(Cx_i) - f(Cx_r)$, $i = 1, \ldots, N$. Then by (4), we have $\eta_{ik} = f_k(C_k^T x_i) - f_k(C_k^T x_r)$ satisfying

$$0 \le \frac{\eta_{ik}}{C_k^T e_i} \le \delta_k, \quad k = 1, \ldots, m, \tag{10}$$

which is equivalent to

$$\eta_{ik}(\eta_{ik} - \delta_k C_k^T e_i) \le 0, \quad k = 1, \ldots, m. \tag{11}$$

From (6) and (9), we obtain the following error system:

$$\dot{e}_i(t) = Ae_i + B\eta_i - \sigma\sum_{j=1}^{N} l_{ij}P^{-1}C^TCe_j - \sigma d_iP^{-1}C^TCe_i, \quad i = 1,\ldots,N. \tag{12}$$

Let $e(t) = (e_1^T(t),\ldots,e_N^T(t))^T$ and $\eta = (\eta_1^T,\ldots,\eta_N^T)^T$. Rewrite (12) in the matrix form as

$$\dot{e}(t) = (I_N \otimes A)e + (I_N \otimes B)\eta - \left(\sigma(L+D)\otimes\left(P^{-1}C^TC\right)\right)e. \tag{13}$$

By using M-matrix theory, Song et al. [29, 30] have shown that $\mathcal{J}(L+D)$, that is, $\min_{1\le i\le N}\mathrm{Re}(\lambda_i(L+D))$ plays an important role in investigating the pinning control of networked systems. The following result is useful to study the synchronization problem of Lur'e network (9).

Lemma 10 (see [30]). *Let $\widetilde{\mathcal{G}}$ be the augmented digraph formed by the nodes of Lur'e network* (5) *and the isolated node* (6). *One has $\mathcal{J}(L+D) > 0$ if and only if $\widetilde{\mathcal{G}}$ contains a directed spanning tree, which means that the isolated node* (6) *has a directed path to every other network node.*

Remark 11. By Lemmas 2 and 10 and the definition of L in (1), one knows that $L+D$ is an M-matrix if and only if $\widetilde{\mathcal{G}}$ contains a directed spanning tree. Moreover, $L+D$ is positive definite if L is symmetric.

4. Pinning Criteria for Undirected Lur'e Network

In this section, we consider the pinning synchronization of Lur'e network (9) with undirected topology and derive some simple pinning conditions for the network by using linear matrix inequality (LMI) and the S-procedure [1, 5, 8].

Theorem 12. *Suppose that $L = L^T$ and there exist a positive definite matrix $P \in \mathbb{R}^{n\times n}$ and a matrix $T = \mathrm{diag}(\tau_1,\ldots,\tau_m) > 0$ such that*

$$Q = \begin{pmatrix} PA + A^TP - 2\sigma\lambda_1C^TC & PB + C^T\Delta T \\ B^TP + T\Delta C & -2T \end{pmatrix} < 0, \tag{14}$$

where $\lambda_1 = \lambda_{\min}(L+D) > 0$. Then, the pinning-controlled Lur'e network (9) *globally asymptotically synchronizes to the isolated node* (6).

Proof. Since $L+D$ is a real symmetric matrix, one knows that all eigenvalues of $L+D$ are real. It follows from the condition of the theorem that the minimum eigenvalue of $L+D$ is a positive number; that is, $\lambda_1 = \lambda_{\min}(L+D) > 0$. Then we see that $L+D$ is a positive definite matrix. By Lemma 4, we have $L+D \ge \lambda_1 I_N > 0$.

From sector condition (11) and $\tau_k > 0$, $k = 1,\ldots,m$, we obtain

$$\sum_{k=1}^{m}\tau_k\eta_{ik}\left(\eta_{ik} - \delta_kC_k^Te_i\right) \le 0, \tag{15}$$

which is equivalent to

$$\eta_i^T\left(T\eta_i - T\Delta Ce_i\right) \le 0, \quad \forall i \in \{1,\ldots,N\}. \tag{16}$$

Take the Lyapunov function candidate as

$$V(t) = \frac{1}{2}e^T(t)\left(I_N \otimes P\right)e(t), \tag{17}$$

where $P > 0$ satisfies condition (14).

Considering Lemma 3, inequality (16), and the S-procedure stated in Lemma 5, we calculate the time derivative of $V(t)$ along the trajectory of error system (13) as follows:

$$\begin{aligned}
\dot{V}(t) &= \sum_{i=1}^{N}e_i^TPAe_i + \sum_{i=1}^{N}e_i^TPB\eta_i - e^T\left[(\sigma(L+D))\otimes\left(C^TC\right)\right]e \\
&\le \sum_{i=1}^{N}e_i^TPAe_i + \sum_{i=1}^{N}e_i^TPB\eta_i \\
&\quad - e^T\left[(\sigma(L+D))\otimes\left(C^TC\right)\right]e - \sum_{i=1}^{N}\eta_i^T\left(T\eta_i - T\Delta Ce_i\right) \\
&\le \sum_{i=1}^{N}\left(e_i^TPAe_i + e_i^TPB\eta_i\right) - e^T\left[(\sigma\lambda_1I_N)\otimes\left(C^TC\right)\right]e \\
&\quad - \sum_{i=1}^{N}\eta_i^T\left(T\eta_i - T\Delta Ce_i\right) \\
&= \sum_{i=1}^{N}\left(e_i^TPAe_i - \sigma\lambda_1e_i^TC^TCe_i + e_i^TPB\eta_i\right. \\
&\quad \left. + e_i^TC^T\Delta T\eta_i - \eta_i^TT\eta_i\right) \\
&= \frac{1}{2}\sum_{i=1}^{N}z_i^TQz_i = \frac{1}{2}z^T(t)\left(I_N\otimes Q\right)z(t),
\end{aligned} \tag{18}$$

where $z_i = (e_i^T,\eta_i^T)^T$, $z(t) = (z_1^T(t),\ldots,z_N^T(t))^T$, and Q is defined in (14).

By (14) and (18), we know that $\dot{V}(t) \le 0$ and $\dot{V}(t) \equiv 0$ if and only if $z(t) = 0_{N(n+m)}$. Thus, the set $\mathcal{M} = \{z(t) \mid z(t) = 0_{N(n+m)}\}$ is the largest invariant set contained in the set $\mathcal{D} = \{z(t) \mid \dot{V}(t) \equiv 0\}$ for error system (13). According to LaSalle's invariance principle [5], starting from any initial condition, every solution of error system (13) approaches $\mathcal{M}$ as $t \to \infty$, which indicates $e_i(t) \to 0$, $i = 1,\ldots,N$. Then the error system (13) is asymptotically stable at the origin. Therefore, the pinning-controlled Lur'e network (9) globally asymptotically synchronizes to the isolated node (6). □

Remark 13. Note that the dimension of LMI condition (14) is equal to $n+m$ determined by the single Lur'e node.

Remark 14. In Remark 8, we have pointed out that one can find a positive definite matrix $P \in \mathbb{R}^{n\times n}$ and a scalar $\tau > 0$ such that $PA + A^T P - \tau C^T C < 0$ if the pair (A, C) is detectable. In view of Lemma 6, we see that the detectability of (A, C) provides a necessary condition for the solvability of LMI (14).

Remark 15. From condition (14) in Theorem 12, we know that the quantity $\lambda_{\min}(L + D)$ is very important to study the synchronization in undirected Lur'e network (9). Suppose that the network topology is composed of several disconnected components. By Lemma 10 and Remark 11, at least one node in each component should be pinned such that $L + D$ is positive definite, and the pinned nodes can be randomly or specifically chosen. For better pinning control performance, one can pin the most highly connected nodes to achieve larger $\lambda_{\min}(L + D)$ such that condition (14) can be satisfied more easily.

5. Pinning Criteria for Directed Lur'e Network

In this section, we study the pinning synchronization of Lur'e network (9) with directed topology. Due to asymmetric coupling, the pinning synchronization problem of directed complex networks is more difficult than that of undirected networks. M-matrix theory will be adopted to derive some pinning conditions for directed Lur'e network (9).

Theorem 16. *Suppose that L is not symmetric. Assume that a positive definite matrix $P \in \mathbb{R}^{n\times n}$ and a matrix $T = \mathrm{diag}(\tau_1, \ldots, \tau_m) > 0$ can be found such that*

$$R = \begin{pmatrix} PA + A^T P - 2\rho C^T C & PB + C^T \Delta T \\ B^T P + T\Delta C & -2T \end{pmatrix} < 0, \tag{19}$$

where ρ is a positive scalar subject to

$$0 < \rho < \sigma \mathcal{J}(L + D). \tag{20}$$

Then, Lur'e network (9) *globally asymptotically synchronizes to the isolated node* (6).

Proof. Let λ_i be the ith eigenvalue of $L + D$. It is easy to verify that $\sigma\lambda_i - \rho$ is the ith eigenvalue of $\sigma(L + D) - \rho I_N$. It follows from condition (20) that $\mathrm{Re}(\sigma\lambda_i - \rho) > 0$ holds for all $i = 1, \ldots, N$. Then, by the definition of L in (1) and Lemma 2, we know that $\sigma(L + D) - \rho I_N$ is an M-matrix and there exists a positive definite diagonal matrix $\Xi = \mathrm{diag}(\xi_1, \ldots, \xi_N) > 0$ such that

$$\left[\Xi\left(\sigma(L + D) - \rho I_N\right)\right]_s > 0. \tag{21}$$

Construct the following Lyapunov function candidate:

$$V(t) = \frac{1}{2} e^T(t)(\Xi \otimes P) e(t), \tag{22}$$

where P satisfies condition (19).

The time derivative of $V(t)$ along the trajectory of error system (13) yields

$$\begin{aligned}
\dot{V}(t) &= \sum_{i=1}^{N} \xi_i e_i^T PAe_i + \sum_{i=1}^{N} \xi_i e_i^T PB\eta_i \\
&\quad - e^T\left[(\sigma\Xi(L + D)) \otimes \left(C^T C\right)\right] e \\
&\le \sum_{i=1}^{N} \xi_i e_i^T PAe_i + \sum_{i=1}^{N} \xi_i e_i^T PB\eta_i \\
&\quad - e^T\left[(\sigma\Xi(L + D)) \otimes \left(C^T C\right)\right] e \\
&\quad - \sum_{i=1}^{N} \xi_i \eta_i^T \left(T\eta_i - T\Delta C e_i\right) \\
&= \sum_{i=1}^{N} \xi_i \left(e_i^T PAe_i + e_i^T PB\eta_i + e_i^T C^T \Delta T\eta_i - \eta_i^T T\eta_i\right) \\
&\quad - \rho\sum_{i=1}^{N} \xi_i e_i^T C^T Ce_i + \rho\sum_{i=1}^{N} \xi_i e_i^T C^T Ce_i \\
&\quad - e^T\left[(\sigma\Xi(L + D)) \otimes \left(C^T C\right)\right] e \\
&= \frac{1}{2}\sum_{i=1}^{N} \xi_i z_i^T Rz_i \\
&\quad - e^T\left(\left[\Xi\left(\sigma(L + D) - \rho I_N\right)\right]_s \otimes \left(C^T C\right)\right) e \\
&\le \frac{1}{2} z^T(t)(\Xi \otimes R) e(t),
\end{aligned} \tag{23}$$

where the first inequality is obtained by applying inequality (16), the last inequality follows (21) and the fact $C^T C \ge 0$, $z_i = (e_i^T, \eta_i^T)^T$, $z(t) = (z_1^T(t), \ldots, z_N^T(t))^T$, and R is defined in (19).

By LaSalle's invariance principle, we can show that $e_i(t) \to 0$, $i = 1, \ldots, N$, which indicates that the pinning-controlled Lur'e network (9) globally asymptotically synchronizes to the isolated node (6). □

Remark 17. From condition (20) in Theorem 16, we know that a set of pinned nodes should be selected to ensure $\mathcal{J}(L + D) > 0$. For a directed network, suppose that the multiplicity of the zero eigenvalue of L is equal to p. Song et al. [30] decomposed the network topology into p components, where each component has a directed tree. By Lemma 10, at least one root node in each component should be pinned such that $L + D$ is an M-matrix, which means that the minimum number of pinned nodes is p. It is worth mentioning that the remaining $N - p$ nodes can be rearranged in descending order according to the differences of their out-degrees and in-degrees [29]. When the network topology contains a directed spanning tree, the Laplacian matrix L has a simple zero eigenvalue [38]; that is, $p = 1$. Then it is possible to pin Lur'e network (9) if the coupling strength σ is sufficiently large in view of conditions (19) and (20), which also confirms the pioneering work in [18].

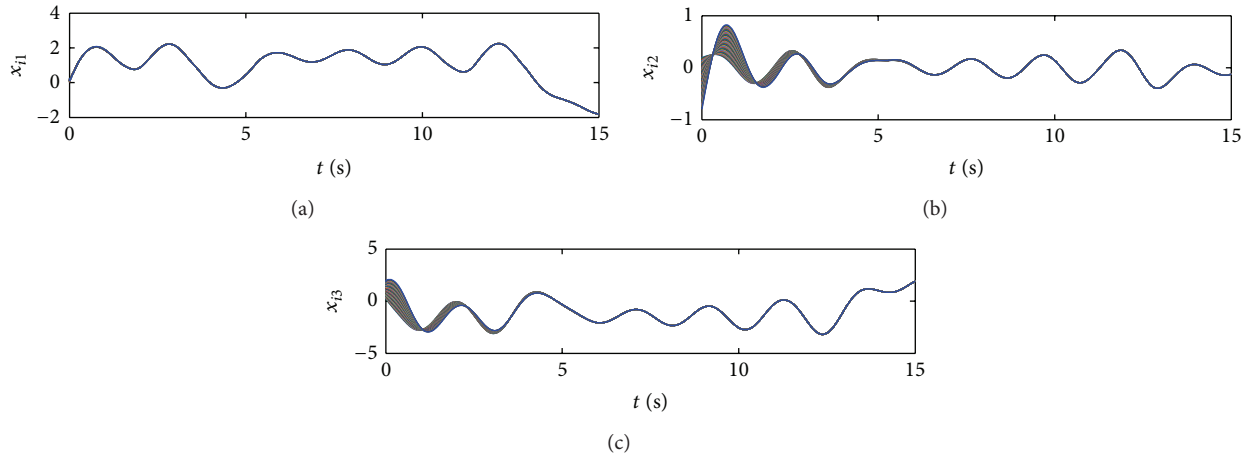


Figure 1: State evolutions of pinning-controlled Lur'e network.

Remark 18. Song et al. [30] have shown that $\mathcal{J}(L + D)$ monotonically increases with respect to the number of pinned nodes or pinning feedback gains. However, $\mathcal{J}(L + D) \leq \mathcal{J}(L(\mathcal{V} \setminus \mathcal{V}_{\text{pin}}))$ always holds even if the pinning feedback gains are sufficiently large. Therefore, it would be desirable to synchronize Lur'e network (9) by pinning a least number of nodes with relatively lower control gains.

Remark 19. One may use $\lambda_{\min}(\Xi(L + D) + (L + D)^T\Xi)$ to replace the parameter ρ in (19) to derive pinning condition for directed Lur'e networks, which will involve the computation of the matrix Ξ. Moreover, it will be quite difficult to discuss the selection of pinned nodes and the design of pinning feedback gains. Thus, we intentionally introduce the parameter ρ in Theorem 16 as a transitional variable to derive condition (19). Treating ρ as a scalar matrix, we can solve LMIs (19) and (20) with Matlab LMI toolbox to obtain appropriate parameters ρ and P. If no feasible solution is found, we can gradually increase the pinning feedback gains or continually add more nodes to $\mathcal{V}_{\text{pin}}$ to increase $\mathcal{J}(L + D)$ until conditions (19) and (20) are satisfied.

6. Numerical Results

In this section, some simulation results are given to verify our theoretical analysis. For a complex network with N nodes, the quantity $E(t) = \sqrt{(1/N)\sum_{i=1}^{N} e_i^T(t)e_i(t)}$ is used to measure the quality of the pinning process [18, 22].

Consider a Lur'e complex network in the form of (5) composed of fifty nodes:

$$\dot{x}_i(t) = Ax_i + Bf(Cx_i) - \sigma\sum_{j=1}^{50} l_{ij}Fy_j + u_i,$$
$$y_i(t) = Cx_i(t), \quad i = 1, \ldots, 50, \tag{24}$$

where $x_i = (x_{i1}, x_{i2}, x_{i3})^T$ is the state variable of the ith node, $\sigma = 5$,

$$A = \begin{pmatrix} -am_1 & a & 0 \\ 1 & -1 & 1 \\ 0 & -b & 0 \end{pmatrix}, \quad B = \begin{pmatrix} -a(m_0 - m_1) \\ 0 \\ 0 \end{pmatrix},$$
$$C = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}, \tag{25}$$

with $a = 9, b = 14.28, m_0 = -1/7$ and $m_1 = 2/7$; the nonlinear function is given by

$$f(Cx_i) = \frac{1}{2}\left(|x_{i1} + 1| - |x_{i1} - 1|\right). \tag{26}$$

The interaction digraph $\mathcal{G}$ of Lur'e network (24) is determined by a directed scale-free network generated by using the techniques in [20]. According to [20], $\mathcal{G}$ is strongly connected and thus contains a directed spanning tree. Based on Remark 17, we know that the minimum number of pinned nodes is one and any node can be chosen to be pinned. When node 2 is pinned with control gain $d_2 = 1.0$, we have

$$\mathcal{J}(L + D) = 0.0912,$$
$$\sigma\mathcal{J}(L + D) = 5 \times 0.0912 = 0.4560. \tag{27}$$

It is easy to verify that the matrix pair (A, C) is detectable, and the nonlinear function f belongs to the sector $[0, 1]$ which means that $\Delta = 1$. Solving LMIs (19) and (20) in Theorem 16, we obtain the following feasible solution:

$$\rho = 0.3470, \qquad T = 0.8556, \tag{28}$$

$$P = \begin{pmatrix} 0.1141 & -0.1179 & 0.0944 \\ -0.1179 & 0.8047 & -0.0776 \\ 0.0944 & -0.0776 & 0.1110 \end{pmatrix}. \tag{29}$$

With the output feedback control algorithm (9), we apply pinning control to node 2 of Lur'e network (24). The time evolutions of network state and synchronization error are shown in Figures 1 and 2, respectively. We can clearly see that the Lur'e network with fifty nodes is successfully synchronized to a homogenous state by pinning a *single* network node.

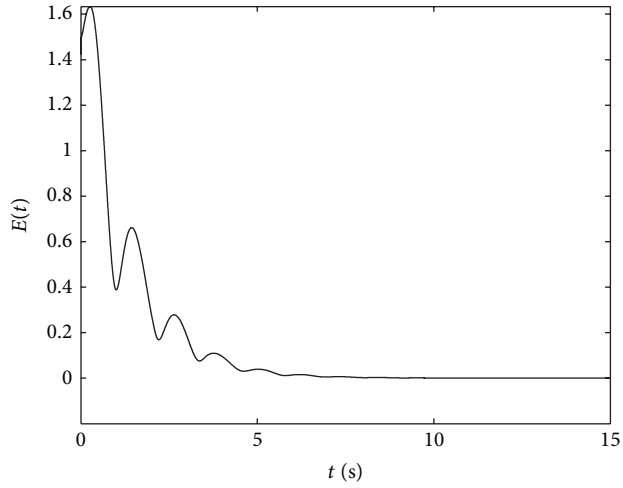


Figure 2: Pinning synchronization error of Lur'e network.

7. Conclusions

This paper proposes a distributed output feedback control approach to investigate the pinning synchronization in Lur'e complex networks. By using the tools from M-matrix theory, Lyapunov functional method, and observer techniques for nonlinear systems, some simple pinning criteria in terms of low-dimensional LMIs, whose dimensions are just determined by that of the single Lur'e node, have been derived for Lur'e networks with undirected and directed topologies, respectively. Numerical simulations have validated the effectiveness of the theoretical analysis. Since the time delay is inevitable in natural and physical systems, in the near future, it would be of interest to study the pinning control problem for Lur'e networks with both discrete and distributed time-delayed coupling on the basis of output feedback control strategies.

Conflict of Interests

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

Acknowledgments

This work was jointly supported by the National Science Foundation of China under Grants no. 61273218, 61304172, 61272530, and 61175119 and the Natural Science Foundation of Henan Province of China under Grant nos. 122102210027, 122300410220, and 12B480005.

References

[1] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, vol. 15, SIAM, Philadelphia, Pa, USA, 1994.

[2] W. M. Haddad and V. Kapila, "Absolute stability criteria for multiple slope-restricted monotonic nonlinearities," *IEEE Transactions on Automatic Control*, vol. 40, no. 2, pp. 361–365, 1995.

[3] P. Park, "A revisited Popov criterion for nonlinear Lur'e systems with sector-restrictions," *International Journal of Control*, vol. 68, no. 3, pp. 461–469, 1997.

[4] J. A. K. Suykens, J. Vandewalle, and B. De Moor, "An absolute stability criterion for the Lur'e problem with sector and slope restricted nonlinearities," *IEEE Transactions on Circuits and Systems I*, vol. 45, no. 9, pp. 1007–1009, 1998.

[5] H. K. Khalil, *Nonlinear Systems*, Prentice Hall, Englewood Cliffs, NJ, USA, 3rd edition, 2002.

[6] C. A. C. Gonzaga, M. Jungers, and J. Daafouz, "Stability analysis of discrete-time Lur'e systems," *Automatica*, vol. 48, no. 9, pp. 2277–2283, 2012.

[7] J. A. K. Suykens, P. F. Curran, and L. O. Chua, "Robust synthesis for master-slave synchronization of Lur'e systems," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 46, no. 7, pp. 841–850, 1999.

[8] M. E. Yalçin, J. A. K. Suykens, and J. Vandewalle, "Master-slave synchronization of Lur'e systems with time-delay," *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, vol. 11, no. 6, pp. 1707–1722, 2001.

[9] X. Huang, J. Cao, and D. W. C. Ho, "Synchronization criteria for Lur'e systems by dynamic output feedback with time-delay," *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, vol. 16, no. 8, pp. 2293–2307, 2006.

[10] H. Huijberts, H. Nijmeijer, and T. Oguchi, "Anticipating synchronization of chaotic Lur'e systems," *Chaos*, vol. 17, no. 1, Article ID 013117, 2007.

[11] J. G. Lu and D. J. Hill, "Impulsive synchronization of chaotic Lur'e systems by linear static measurement feedback: An LMI approach," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 54, no. 8, pp. 710–714, 2007.

[12] J. Lu, J. Cao, and D. W. C. Ho, "Adaptive stabilization and synchronization for chaotic Lur'e systems with time-varying delay," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, no. 5, pp. 1347–1356, 2008.

[13] X. Liu, J. Wang, and L. Huang, "Global synchronization for a class of dynamical complex networks," *Physica A: Statistical Mechanics and Its Applications*, vol. 386, no. 1, pp. 543–556, 2007.

[14] K. Ding and Q.-L. Han, "Effects of coupling delays on synchronization in Lur'e complex dynamical networks," *International Journal of Bifurcation and Chaos*, vol. 20, no. 11, pp. 3565–3584, 2010.

[15] D. H. Ji, J. H. Park, W. J. Yoo, S. C. Won, and S. M. Lee, "Synchronization criterion for Lur'e type complex dynamical networks with time-varying delay," *Physics Letters A: General, Atomic and Solid State Physics*, vol. 374, no. 10, pp. 1218–1227, 2010.

[16] X. F. Wang and G. Chen, "Pinning control of scale-free dynamical networks," *Physica A: Statistical Mechanics and Its Applications*, vol. 310, no. 3-4, pp. 521–531, 2002.

[17] X. Li, X. Wang, and G. Chen, "Pinning a complex dynamical network to its equilibrium," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 10, pp. 2074–2087, 2004.

[18] T. Chen, X. Liu, and W. Lu, "Pinning complex networks by a single controller," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 6, pp. 1317–1326, 2007.

[19] C. W. Wu, "Localization of effective pinning control in complex networks of dynamical systems," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '08)*, pp. 2530–2533, Seattle, Wash, USA, May 2008.

[20] Y. Y. Lu and X. F. Wang, "Pinning control of directed dynamical networks based on ControlRank," *International Journal of Computer Mathematics*, vol. 85, no. 8, pp. 1279–1286, 2008.

[21] J. Lu, D. W. C. Ho, and Z. Wang, "Pinning stabilization of linearly coupled stochastic neural networks via minimum number of controllers," *IEEE Transactions on Neural Networks*, vol. 20, no. 10, pp. 1617–1629, 2009.

[22] 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.

[23] Q. Z. Zhang and Z. K. Li, "Pinning control of complex Lur'e networks," *Chinese Physics B*, vol. 18, no. 6, pp. 2176–2183, 2009.

[24] Z. Li, Z. Duan, and G. Chen, "Global synchronised regions of linearly coupled Lur'e systems," *International Journal of Control*, vol. 84, no. 2, pp. 216–227, 2011.

[25] P. DeLellis and M. di Bernardo, "Adaptive pinning control of complex networks of Lur'e systems," in *Proceedings of the 51st IEEE Conference on Decision and Control*, pp. 6060–6064, Maui, Hawaii, USA, 2012.

[26] Q. Song, F. Liu, J. Cao, and J. Lu, "Some simple criteria for pinning a Lur'e network with directed topology," *IET Control Theory and Applications*, vol. 8, no. 2, pp. 131–138, 2014.

[27] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1991.

[28] A. Berman and R. J. Plemmons, *Nonnegative Matrices in the Mathematical Sciences*, vol. 9, SIAM, Philadelphia, Pa, USA, 1994.

[29] Q. Song, F. Liu, J. Cao, and W. Yu, "Pinning-controllability analysis of complex networks: an M-matrix approach," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 59, no. 11, pp. 2692–2701, 2012.

[30] Q. Song, F. Liu, J. Cao, and W. Yu, "M-matrix strategies for pinning-controlled leader-following consensus in multiagent systems with nonlinear dynamics," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1688–1697, 2013.

[31] G. Wen, Z. Duan, G. Chen, and W. Yu, "Consensus tracking of multi-agent systems with Lipschitz-type node dynamics and switching topologies," *IEEE Transactions on Circuits and Systems I*, vol. 61, no. 2, pp. 499–511, 2014.

[32] W. Lu and T. Chen, "New approach to synchronization analysis of linearly coupled ordinary differential systems," *Physica D: Nonlinear Phenomena*, vol. 213, no. 2, pp. 214–230, 2006.

[33] A. Arenas, A. Díaz-Guilera, J. Kurths, Y. Moreno, and C. Zhou, "Synchronization in complex networks," *Physics Reports*, vol. 469, no. 3, pp. 93–153, 2008.

[34] J. Lu, D. W. C. Ho, and J. Cao, "Synchronization in an array of nonlinearly coupled chaotic neural networks with delay coupling," *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, vol. 18, no. 10, pp. 3101–3111, 2008.

[35] J. Lu and D. W. C. Ho, "Stabilization of complex dynamical networks with noise disturbance under performance constraint," *Nonlinear Analysis: Real World Applications*, vol. 12, no. 4, pp. 1974–1984, 2011.

[36] G.-P. Jiang, W. K.-S. Tang, and G. Chen, "A state-observer-based approach for synchronization in complex dynamical networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 53, no. 12, pp. 2739–2745, 2006.

[37] J. Wu and L. Jiao, "Observer-based synchronization in complex dynamical networks with nonsymmetric coupling," *Physica A: Statistical Mechanics and Its Applications*, vol. 386, pp. 469–480, 2007.

[38] 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.

[39] G. Wen, Z. Duan, W. Yu, and G. Chen, "Consensus of multi-agent systems with nonlinear dynamics and sampled-data information: a delayed-input approach," *International Journal of Robust and Nonlinear Control*, vol. 23, no. 6, pp. 602–619, 2013.

[40] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1985.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 636784, 12 pages
http://dx.doi.org/10.1155/2014/636784

Research Article

Distributed Cooperative Current-Sharing Control of Parallel Chargers Using Feedback Linearization

Jiangang Liu,[1] Zhiwu Huang,[1] Jing Wang,[1,2] Jun Peng,[1] and Weirong Liu[1]

[1] *Hunan Engineering Laboratory for Advanced Control and Intelligent Automation, School of Information Science and Engineering, Central South University, Changsha, Hunan 410075, China*
[2] *School of Science, Engineering and Mathematics, Bethune-Cookman University, Daytona Beach, FL 32114, USA*

Correspondence should be addressed to Zhiwu Huang; hzw@csu.edu.cn

Received 16 December 2013; Accepted 19 February 2014; Published 7 April 2014

Academic Editor: Guoqiang Hu

We propose a distributed current-sharing scheme to address the output current imbalance problem for the parallel chargers in the energy storage type light rail vehicle system. By treating the parallel chargers as a group of agents with output information sharing through communication network, the current-sharing control problem is recast as the consensus tracking problem of multiagents. To facilitate the design, input-output feedback linearization is first applied to transform the nonidentical nonlinear charging system model into the first-order integrator. Then, a general saturation function is introduced to design the cooperative current-sharing control law which can guarantee the boundedness of the proposed control. The cooperative stability of the closed-loop system under fixed and dynamic communication topologies is rigorously proved with the aid of Lyapunov function and LaSalle invariant principle. Simulation using a multicharging test system further illustrates that the output currents of parallel chargers are balanced using the proposed control.

1. Introduction

As a new type of electric traction light rail transportation system, the energy storage type light rail vehicle, adopts supercapacitors as its power supply. With this energy storage technology, there is no need to construct a traction power grid and it is possible to recover the energy of regenerative braking. Energy storage type light rail vehicles need to be charged in seconds by the charging system when parking at the platform. Therefore, the charging system should provide a large enough output power to shorten the charging process. An effective method to solve this problem would be by connecting several chargers in parallel to increase system capacity [1], as shown in Figure 1. However, the challenge for the charging system is how to balance the charging current between the chargers.

If the charging current is not balanced, the charger with a higher output current has to bear a greater output power, which may lead to a large thermal stress and degrade the reliability and performance of the whole charging system. It is therefore necessary to design an effective current-sharing control strategy to balance the output current for charging system.

There are several conventional approaches for current-sharing problems, such as central current-sharing control method, droop control method [2, 3], and master-slave method [4, 5]. The central current-sharing control method needs a central controller to compute each module's output current according to the total load current. The central controller sends a unified command to balance the current. This method is relatively simple and easy to implement. However, if the central controller is faulty, this may result in the collapse of the whole charging system and degrade the whole system's reliability.

Under the droop control method, the output voltage decreases as the output current increases, effectively controlling and changing the output impedance to achieve balance [2, 3]. The method is simple and has been widely used in small-capacity parallel charging systems. However, this approach is not suitable for large-power paralleled charging

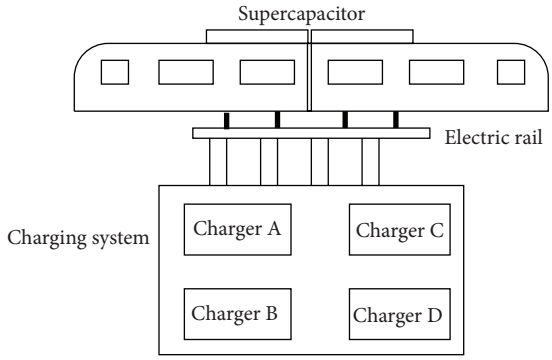


Figure 1: The schematic diagram of charging system for the energy storage type light rail vehicle system.

system and is not effective when the load has relatively large changes.

Master-slave current-sharing control is manually specifying one module among N parallel power modules as the main module and the remaining module's current follows the main module's allocation current [4, 5]. This method is suitable for a double closed-loop control system with an external voltage loop and an inner current loop. However, this method has the same drawback as the central current-sharing control approach, which means that if the main module is faulty, the whole system will collapse.

Central current-sharing control method, droop control method, and master-slave method are undistributed and restrictive and have some drawbacks and limitations, which are not the appropriate approach for the charging system of energy storage type light rail vehicles when the chargers are parallel with each other. Taking distributed chargers in the charging system as agents in multiagent systems, the current-sharing control design resembles a current consensus-tracking problem, where the charger's current tracks the reference. Several chargers are connected in parallel with each other in the charging system of an energy storage type light rail vehicle. The adjacent chargers interact with each other and exchange the state information such as the charging current. This method is fully distributed, each charger only requires its own information and the information of some neighbors, and one faulty node cannot cause the collapse of the whole charging system, which is an appropriate method for the charging system to solve the current-sharing problem.

Over the last decades, consensus and cooperative control of multiagent systems have received most of the attention [6], giving results in the areas such as flocking [7], formation control [8], distributed mobile sensor network [9], rendezvous in space, and autonomous vehicles [10]. Distributed consensus and cooperative control have also been recently introduced in power systems to regulate the output power of multiple photovoltaic generators [11], which has also been used to synchronize the output voltage of distributed generators in microgrids [12]. Although the cooperative control has been applied in power system, the control input constraints were not considered in detail.

However, the charging system has some physical constraints, such as the duty ratio. And there are inevitably components error and manufacturing error in each charger system; the dynamics of the chargers in the charging system are nonidentical. The charger has intrinsic nonlinear characteristics because of Buck DC/DC circuit's principle and super capacitor's feature. This paper seeks to address the challenge of how to design the cooperative current-sharing control law and take into account both the charger's nonidentical and nonlinear features.

The main contribution of this paper can be summarized as follows.

The current-sharing control of charging system for energy storage type light rail vehicles is implemented through the concept of distributed cooperative control of multiagent systems. The current-sharing problem is converted into the consensus-tracking problem of nonlinear and nonidentical multiagent systems.

Input-output feedback linearization is introduced to transform the nonidentical dynamics into first-order integrator. A general saturation function is introduced to design the cooperative current-sharing control law based on neighborhood rule. The Lyapunov function integrating LaSalle invariant principle is adopted to prove the cooperative stability of closed-loop systems.

The proposed current-sharing control scheme is implemented through a sparse communication network, which is essentially distributed. The communication network is modeled by a graph. Each charger only requires its own current information and that of its neighbors.

The rest of this paper is organized as follows: current-sharing problems of the charging system are set out in Section 2, the current-sharing control strategy based on distributed cooperative control of multiagent systems is proposed in Section 3, and cooperative stability is proved in Section 4. The proposed current-sharing strategy is verified in Section 5. Section 6 concludes this paper.

2. Charging System Problem Formulation

Considering the charging system for the energy storage type light rail vehicle, as shown in Figure 2, the charging system consists of a 10 kV AC supply grid, a 10 kV/900 V AC converter, and a network of parallel charger subsystems. All of the charger subsystems are able to exchange the state information through communications.

Each charger subsystem is composed of a three-phase bridge rectifier circuit and a chopper BUCK DC/DC circuit (Figure 3). The current dynamics in the BUCK circuit can be established based on Kirchhoff's Voltage Law. To this end, when the IGBT (Insulated Gate Bipolar Transistor) is ON, the dynamics of the current in the BUCK circuit are described as

$$\dot{I}_i = \frac{U_{d_i}}{L_i} D_i - \frac{I_i r_i}{L_i} - \frac{U_c}{L_i}, \tag{1}$$

and when the IGBT is OFF, we have

$$\dot{I}_i = -\frac{I_i r_i}{L_i} - \frac{U_c}{L_i}, \tag{2}$$

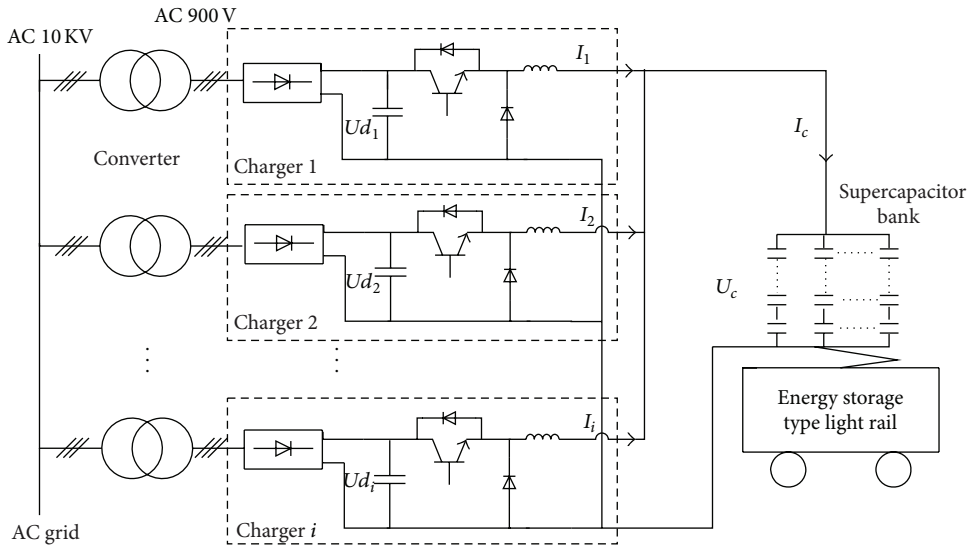


Figure 2: The main circuit schematic diagram of charging system for energy storage type light rail.

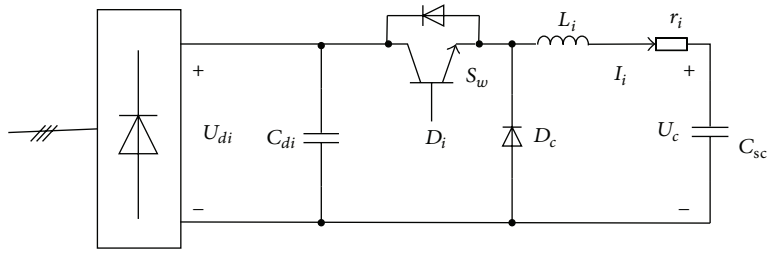


Figure 3: Buck circuit for each charger subsystem.

where I_i is the output current of the ith charging subsystem, U_c is supercapacitor's voltage, U_{d_i} is the DC input voltage which is obtained from the three-phase bridge rectifier, D_i is the IGBT duty ratio, L_i is the flux for the energy storage inductor, and r_i is the equivalent resistance of the circuit.

It can be seen from Figure 3 that supercapacitor is the key element in the energy storage light rail vehicle system. We use C_{sc} to denote its capacitance, and its value may ascend as the voltage increases. There exists electric charge redistribution process at the end of the charging stage, and a leakage current exists as well. Such features can be described by their equivalent three-branch RC circuit shown in Figure 4. The first branch is the instantaneous branch, which includes a resistor R_1, two capacitors C_0 and C_v. Voltage-dependent capacitance C_v relies on the terminal voltage U_c. The second branch is the delay branch, which includes a resistor R_2 and a capacitor C_2. The third branch is the long term branch, which includes a resistor R_3 and a capacitor C_3. R_{lea} is the leakage resistor. The time constants for the instantaneous branch, the delay branch, and the long term branch are different, and the values are in the level of seconds, minutes, and ten minutes, respectively. The charging process is mainly dependent on the first RC branch with the smallest time constant which meets the fast charging requirement for the energy storage type light rail vehicle system.

The relationship between capacitance and voltage for the super capacitor is given by

$$C_{sc} = C_0 + C_v U_c. \tag{3}$$

The voltage dynamic can be described by the following nonlinear differential equation:

$$\begin{aligned} \dot{U}_c &= \frac{I_c}{C_{sc}} \\ &= \frac{I_c}{C_0 + C_v U_c} \\ &= \frac{1}{C_0 + C_v U_c} \sum_{i=1}^{n} I_i, \end{aligned} \tag{4}$$

where I_c is the total charging current, I_i is the output current of the ith charging subsystem, and n is the number of the total charging subsystems.

To this end, it follows from (1)–(4) that the system model for the ith charging subsystem can be described as

$$\begin{aligned} \dot{x}_i &= f_i(x_1, x_2, \ldots, x_n) + g_i(x_i)\, u_i, \\ y_i &= h_i(x_i), \end{aligned} \tag{5}$$

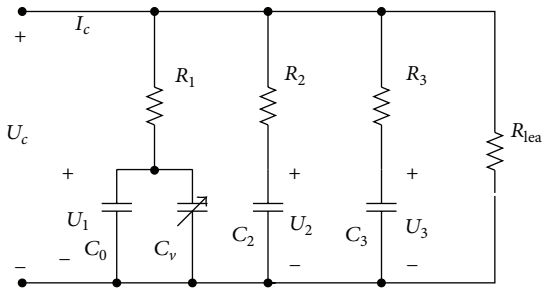


Figure 4: Equivalent three branches RC circuit model of the supercapacitor.

where $i = 1, 2, \ldots, n$, $x_i = [I_i, U_c]^T \in R^2$ is ith charger's state, $u_i(t) = U_{d_i} D_i \in R$ is the control input, and system functions $f_i(x_1, x_2, \ldots, x_n)$ and $g_i(x_i)$ are Lipschitz continuous and defined below

$$f_i(x_1, x_2, \ldots, x_n) = \begin{bmatrix} -\dfrac{r_i}{L_i} I_i & -\dfrac{U_c}{L_i} \\ \dfrac{1}{C_0 + C_v U_c} & \sum_{k=1}^{n} I_k \end{bmatrix},$$

$$g_i(x_i) = \begin{bmatrix} \dfrac{1}{L_i} \\ 0 \end{bmatrix}, \tag{6}$$

$$h_i(x_i) = I_i.$$

For the charging system of the energy storage light rail vehicle, there are several modeling methods. In this paper, the average value modeling method is adopted to describe the dynamic of the charger. The current I_i and voltage U_c are chosen as the charger's state x_i. $f_i(x_1, x_2, \ldots, x_n)$, $g_i(x_i)$ are the nonlinear functions, which describe the nonlinear and nonidentical charger and are related with the current $I_1, I_2, \ldots, I_n$ and voltage U_c. When we choose average value modeling method, the voltage $U_{d_i} D_i$ is chosen as the control input u_i, which is bounded because the duty ratio is within a range. The current I_i is considered as the output of the charger; that is, $y_i = h_i(x_i) = I_i$.

To ensure the charging performance for the energy storage rail vehicle system, it is expected that the output current I_i for all charging subsystems would be identical. However, in practice, it is unavoidable that the DC input voltages U_{d_i} for the charging subsystems may not be same, which results in the unbalanced output current I_i. To solve this problem, in this paper, the control objective is to design a cooperative control $u_i(t)$ such that

$$\lim_{t \to +\infty} |y_i(t) - y_0| = 0, \tag{7}$$

where y_0 is the reference current.

The reference current y_0 is a prespecified parameter. In this paper, we assume that this information is only available to one of the charging subsystems. Through information exchanges among the charging subsystems, the proposed cooperative control is designed to achieve the control objective defined in (7).

In real applications, y_0 is calculated by $y_0 = I_c/n$, given n parallel subsystems in the current charging process, where I_c represents the total charging current. Once the desired voltage $U_c(t_d)$ within the required charging time t_d, the initial capacitances $C_{\text{sc}}(0)$, and the initial residual voltage $U_c(0)$ of supercapacitors are given, I_c can be computed by $I_c = C_{\text{sc}}(0)(U_c(t_d) - U_c(0))/t_d$ in advance.

Each controller communicates with each other through RS-422 bus. And every controller can know whether other controllers are involved in the charging process or not. Therefore, the number n can be counted. When some charger cannot work, the desired setpoint reference current y_0 would be different accordingly as the total number n of working charging subsystems during that stage has changed. But y_0 cannot exceed the maximum current $I_{\max}$ that each charger can afford.

The cooperative current-sharing objective y_0 indeed depends on the number n and the total charging current I_c. It is worth mentioning that the messages about y_0 are only required to be configured to one of charging subsystems in the network. And due to the nature of the proposed cooperative control, all other subsystems in the network will eventually converge to the newly updated setpoint y_0.

3. Current-Sharing Strategy Based on Distributed Cooperative Control

To design the distributed cooperative current-sharing control for the charging system, we assume that the charging subsystems are able to exchange information through communication network. We use a graph to describe such a communication network. In this section, after some preliminaries on graph theory, we present the proposed cooperative control for distributed current-sharing control based on input-output feedback linearization technique.

3.1. Preliminary of Graph Theory. The communication topology of a multiagent system can be modeled by a graph [13]. A graph is usually denoted as $G(\nu, \varepsilon, A)$ with a nonempty finite set of n nodes $\nu = \{\nu_1, \nu_2, \ldots, \nu_n\}$, a set of edges or arcs $\varepsilon \subseteq \nu \times \nu$, and the associated adjacency matrix $A = [a_{ij}] \in R^{n \times n}$.

When the communication topology is fixed, the adjacency matrix A is constant; otherwise A is time-varying; that is, $A(t) = [a_{ij}(t)]_{n \times n}$. An edge from node j to node i is denoted by (ν_j, ν_i), which means that node i receives the information from node j. a_{ij} is the weight of edge (ν_j, ν_i), and $a_{ij} > 0$ if $(\nu_j, \nu_i) \in \varepsilon$; otherwise $a_{ij} = 0$. The graph is assumed to have no self-loops, which means $a_{ii} = 0$.

Node j is called a neighbor of node i if $(\nu_j, \nu_i) \in \varepsilon$. The set of neighbors of node i is denoted as $N_i = \{\nu_j \in \nu : (\nu_j, \nu_i) \in \varepsilon\}$.

Definition 1. A graph without loops is called simple graph [7], that is, the adjacency matrix diagonal elements $a_{ii} = 0$, and its Laplacian matrix is denoted as $L = [l_{ij}]_{n \times n}$:

$$l_{ij} = \begin{cases} \sum_{k=1, k \neq i}^{n} a_{ik}, & j = i \\ -a_{ij}, & j \neq i. \end{cases} \tag{8}$$

The in-degree and out-degree of node v_i are, respectively, defined as follows:

$$\deg_{\text{in}}(v_i) = \sum_{j \in N_i} a_{ij}, \qquad \deg_{\text{out}}(v_i) = \sum_{j \in N_i} a_{ji}. \tag{9}$$

Definition 2 (balanced graphs). The node i of a digraph $G(\nu, \varepsilon, A)$ is balanced if and only if its in-degree and out-degree are equal; that is, $\deg_{\text{in}}(v_i) = \deg_{\text{out}}(v_i)$. A graph $G(\nu, \varepsilon, A)$ is called balanced if and only if all of its nodes are balanced.

The in-degree matrix is defined as $D = \text{diag}\{d_i\} \in R^{n \times n}$ with $d_i = \sum_{j \in N_i} a_{ij}$. The Laplacian matrix can also be defined as $L = D - A$. All row sums of L are equal to zero; that is $L1_n = 0$, where 1_n is a column vector with n elements equal to 1.

A directed path from node i to node j is a sequence of edges, expressed as

$$\left\{(v_i, v_k), (v_k, v_l), \ldots, (v_m, v_j)\right\}. \tag{10}$$

A digraph is said to have a spanning tree, if there is a node i_r (called the root), such that there is a directed path from the root to every other node in the graph.

Remark 3. When the communication topology is dynamic, the associated adjacency matrix is $A(t) = [a_{ij}(t)]_{n \times n}$; the Laplacian matrix is denoted as $L(t) = [l_{ij}(t)]_{n \times n}$, where

$$l_{ij}(t) = \begin{cases} \sum_{k=1, k \neq i}^{n} a_{ik}(t), & j = i \\ -a_{ij}(t), & j \neq i. \end{cases} \tag{11}$$

3.2. Current-Sharing Controller Design under Fixed Topology. To facilitate the cooperative design, we first convert the nonlinear model in (5) into the linear one through input-output feedback linearization. It follows that the time derivative of the output y_i is

$$\begin{aligned} \dot{y}_i &= \frac{\partial h_i(x_i)}{\partial x_i} \dot{x}_i \\ &= \frac{\partial h_i(x_i)}{\partial x_i} \left(f_i(x_1, x_2, \ldots, x_n) + g_i(x_i) u_i\right) \\ &= L_{f_i} h_i + L_{g_i} h_i u_i, \end{aligned} \tag{12}$$

where $L_{f_i} h_i$ is the Lie derivative of h_i with respect to f_i and is defined by $L_{f_i} h_i = \nabla h_i f_i = (\partial h_i(x_i)/\partial x_i) f_i(x_1, x_2, \ldots, x_n)$ and $L_{g_i} h_i$ is the Lie derivative of h_i with respect to g_i and is defined by $L_{g_i} h_i = \nabla h_i g_i = (\partial h_i(x_i)/\partial x_i) g_i(x_i)$. Define an auxiliary control ϑ_i as follows:

$$\vartheta_i = L_{f_i} h_i + L_{g_i} h_i u_i. \tag{13}$$

Substituting (13) into (12), we have the first-order integrator model

$$\dot{y}_i = \vartheta_i. \tag{14}$$

Since the model of the charger is based on converter by using average value modeling method, the duty cycle D_i of Insulated Gate Bipolar Transistor is bounded and the voltage $U_{d_i} \cdot D_i$ is also bounded. Therefore, we introduce a saturation function $\phi(\cdot)$ in the proposed cooperative control to guarantee the boundedness of the control input.

Assumption 4. A general saturation function $\phi(\cdot)$ satisfies the following:

(1) $\phi(\cdot)$ is Lipschitz continuous,

(2) $\phi(z) = 0 \Leftrightarrow z = 0$,

(3) $z\phi(z) > 0, \forall z \neq 0$,

(4) $\phi_{\min} \leq \phi(z) \leq \phi_{\max}, \forall z \in R$.

Remark 5. One possible choice of saturation functions is hyperbolic tangent function as used in [14]. Nonetheless, any function satisfying Assumption 4 would work. To this end, once ϑ_i is bounded, the original control input u_i is also bounded based on inverse transformation of (13) since $f_i(\cdot)$ and $g_i(\cdot)$ are bounded.

The proposed auxiliary control ϑ_i can be designed as follows:

$$\vartheta_i = c_i \left(\sum_{j \in N_i} a_{ij} \phi\left(y_j - y_i\right) + \rho_i \phi\left(y_0 - y_i\right) \right), \tag{15}$$

where $\phi(z)$ satisfy Assumption 4, $N_i = \{v_j \in \nu : (v_i, v_j) \in \varepsilon\}$ the set of neighbors of v_i, $a_{ij} > 0$ means that for the ith charger, it can receive the information of the jth charger. $c_i > 0$ is the coupling strength. The control parameter c_i can be chosen appropriately to improve the response time [15]. ρ_i is the pinning gain and $\rho_i > 0$ for at least one i, which means that at least one charger knows the virtual charging current. From (15), we know that the current-sharing control law ϑ_i is only based on the ith charger's current information and the information of its neighbors.

Remark 6. In order to make the current-sharing controller designing much more flexible for the nonidentical charging system, the different coupling strength c_i is chosen in the updated cooperative control law for each subsystem instead of using the traditional common coupling strength c [15]. The convergence speed can be improved by tuning c_i. In addition, c_i does not depend on the whole network topology.

Accordingly, the cooperative control input u_i can be carried out by ϑ_i as

$$u_i = \frac{\vartheta_i - L_{f_i} h_i}{L_{g_i} h_i}. \tag{16}$$

When the communication topology is fixed, the final cooperative controller is formulated as follows by substituting (15) into (16):

$$u_i = \frac{c_i \left(\sum_{j \in N_i} a_{ij} \phi\left(y_j - y_i\right) + \rho_i \phi\left(y_0 - y_i\right) \right) - L_{f_i} h_i}{L_{g_i} h_i}. \tag{17}$$

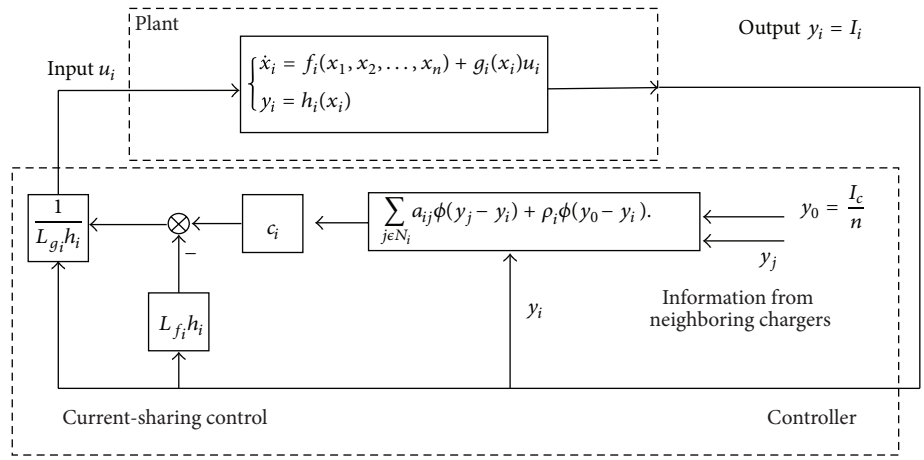


Figure 5: The block diagram of the proposed cooperative current-sharing control.

The block diagram of the proposed cooperative control is shown in Figure 5.

By substituting (17) into (12), we have each subsystem's closed-loop dynamics:

$$\dot{y}_i = c_i\left(\sum_{j\in N_i} a_{ij}\phi\left(y_j - y_i\right) + \rho_i\phi\left(y_0 - y_i\right)\right), \quad i = 1, 2, \ldots, n. \tag{18}$$

As RS-422 full duplex communication protocol is used between each charger, each charger can communicate with each other in a bidirectional way. The communication topology is an undirected graph. Therefore, we analyze the undirected case in our work.

Assumption 7. The communication network between each chargers is bidirectional information flow, which means that the directed graph $G(\nu, \varepsilon, A)$ for the fixed communication topology is symmetric; that is, $a_{ij} = a_{ji}$, $i, j = 1, 2, \ldots, n$.

To facilitate the analysis in Section 4, let us further define the new auxiliary state variable $\delta_i(t) = y_i(t) - y_0$. Furthermore, each auxiliary subsystem's state equation can be written as

$$\dot{\delta}_i = \dot{y}_i. \tag{19}$$

Furthermore, we can get

$$\dot{\delta}_i = -c_i\left(\sum_{j\in N_i} a_{ij}\phi\left(\delta_i - \delta_j\right) + \rho_i\phi\left(\delta_i\right)\right), \quad i = 1, 2, \ldots, n. \tag{20}$$

3.3. Current-Sharing Controller Design under Dynamical Topology. When the communication topology is dynamic, the distributed cooperative controller is formulated as follows by substituting (15) into (16):

$$u_i = \frac{c_i\left(\sum_{j\in N_i} a_{ij}(t)\,\phi\left(y_j - y_i\right) + \rho_i\phi\left(y_0 - y_i\right)\right) - L_{f_i}h_i}{L_{g_i}h_i}, \tag{21}$$

where $N_i = \{\nu_j \in \nu : (\nu_i, \nu_j) \in \varepsilon\}$ the set of neighbors of ν_i, $a_{ij}(t) > 0$ is the element of the time-varying adjacent matrix $A(t)$.

By substituting time varying cooperative control law (21) into system (12), we can get each subsystem's closed-loop dynamics:

$$\dot{y}_i = c_i\left(\sum_{j\in N_i} a_{ij}(t)\,\phi\left(y_j - y_i\right) + \rho_i\phi\left(y_0 - y_i\right)\right), \quad i = 1, 2, \ldots, n. \tag{22}$$

Being similar to the fixed topology case, we can get the each auxiliary subsystem's state equation under the dynamic topology:

$$\dot{\delta}_i = \dot{y}_i = -c_i\left(\sum_{j\in N_i} a_{ij}(t)\,\phi\left(\delta_i - \delta_j\right) + \rho_i\phi\left(\delta_i\right)\right), \quad i = 1, 2, \ldots, n. \tag{23}$$

Assumption 8. The directed dynamic graph $G_{s(t)}$ for the dynamic communication topology is symmetric at each time t; that is, $a_{ij}(t) = a_{ji}(t)$, $i, j = 1, 2, \ldots, n$.

4. Charging System Cooperative Stability Analysis

In this section, the cooperative stability analysis of the multicharging systems under fixed and dynamic topology with the aid of the Lyapunov function integrating LaSalle Invariant Principle is given.

Lemma 9. *Suppose $\xi_i \in R^m$, $\zeta_j \in R^m$, $\forall i, j = 1, 2, \ldots, n$ and $S = [s_{ij}]_{n\times n} \in R^{n\times n}$; if matrix S is symmetric, then*

$$\begin{aligned}&\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\left(\xi_i-\xi_j\right)^T\phi\left(\zeta_i-\zeta_j\right)\\&\quad=\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\xi_i^T\phi\left(\zeta_i-\zeta_j\right).\end{aligned}\tag{24}$$

Proof. Consider

$$\begin{aligned}&\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\left(\xi_i-\xi_j\right)^T\phi\left(\zeta_i-\zeta_j\right)\\&\quad=\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\xi_i^T\phi\left(\zeta_i-\zeta_j\right)\\&\quad\quad-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\xi_j^T\phi\left(\zeta_i-\zeta_j\right).\end{aligned}\tag{25}$$

Because the function $\phi(x)$ is a bounded odd function, the minus of the second term can be substituted into the brackets:

$$\begin{aligned}&\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\left(\xi_i-\xi_j\right)^T\phi\left(\zeta_i-\zeta_j\right)\\&\quad=\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\xi_i^T\phi\left(\zeta_i-\zeta_j\right)\\&\quad\quad+\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\xi_j^T\phi\left(\zeta_j-\zeta_i\right).\end{aligned}\tag{26}$$

Then exchanging i with j in the second term, we get

$$\begin{aligned}&\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\left(\xi_i-\xi_j\right)^T\phi\left(\zeta_i-\zeta_j\right)\\&\quad=\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\xi_i^T\phi\left(\zeta_i-\zeta_j\right)\\&\quad\quad+\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ji}\xi_i^T\phi\left(\zeta_i-\zeta_j\right).\end{aligned}\tag{27}$$

Since S is symmetric, we obtain

$$\begin{aligned}&\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\left(\xi_i-\xi_j\right)^T\phi\left(\zeta_i-\zeta_j\right)\\&\quad=\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\xi_i^T\phi\left(\zeta_i-\zeta_j\right).\end{aligned}\tag{28}$$

This completes the proof. □

4.1. Cooperative Stability Analysis under Fixed Topology

Theorem 10. *Consider the multicharging systems (5) with fixed communication topology. Under the Assumption 7 that the directed graph is symmetric and $\rho_i > 0$ for at least one charger, the cooperative control objective below can be achieved by the control law (17):*

$$\lim_{t\to+\infty}\left|y_i(t)-y_0\right|=0.\tag{29}$$

That is, the output current of the parallel chargers can ultimately be consensus and track the desired reference current. Furthermore, the overall closed-loop system is asymptotically cooperative stable.

Proof. To prove the multicharging system's cooperative stability, we turn to analyze the auxiliary system's (20) stability indirectly.

For the auxiliary closed-loop system (20), the Lyapunov function candidate can be chosen as follows:

$$V=\frac{1}{2}\sum_{i=1}^{n}\delta_i^T\delta_i.\tag{30}$$

Differentiating the Lyapunov function V along the auxiliary subsystem's trajectory with respect to time t and then substituting the state equation of each subsystem into the derivative function $\dot{V}$, we can have

$$\begin{aligned}\dot{V}&=\sum_{i=1}^{n}{\delta_i}^T\dot{\delta}_i\\&=\sum_{i=1}^{n}{\delta_i}^T\left(c_i\left(\sum_{j\in N_i}a_{ij}\phi\left(\delta_j-\delta_i\right)\right)-\rho_i\phi\left(\delta_i\right)\right)\\&=-c_i\sum_{i=1}^{n}{\delta_i}^T\sum_{j\in N_i}a_{ij}\phi\left(\delta_i-\delta_j\right)-c_i\sum_{i=1}^{n}{\delta_i}^T\rho_i\phi\left(\delta_i\right).\end{aligned}\tag{31}$$

According to the communication topology, we know $\sum_{j\in N_i}a_{ij}=\sum_{j=1}^{n}a_{ij}$.

The derivative function $\dot{V}$ (31) can be transformed into

$$\begin{aligned}\dot{V}&=-c_i\sum_{i=1}^{n}{\delta_i}^T\sum_{j=1}^{n}a_{ij}\phi\left(\delta_i-\delta_j\right)-c_i\sum_{i=1}^{n}{\delta_i}^T\rho_i\phi\left(\delta_i\right)\\&=-c_i\sum_{i=1}^{n}\sum_{j=1}^{n}a_{ij}{\delta_i}^T\phi\left(\delta_i-\delta_j\right)-c_i\sum_{i=1}^{n}{\delta_i}^T\rho_i\phi\left(\delta_i\right).\end{aligned}\tag{32}$$

Based on Lemma 9, the above equation (32) is equal to

$$\begin{aligned}\dot{V}=&-\frac{1}{2}c_i\sum_{i=1}^{n}\sum_{j=1}^{n}a_{ij}\left(\delta_i-\delta_j\right)^T\phi\left(\delta_i-\delta_j\right)\\&-c_i\sum_{i=1}^{n}\delta_i^T\rho_i\phi\left(\delta_i\right).\end{aligned}\tag{33}$$

Since $\phi(\delta_i-\delta_j)$ and $(\delta_i-\delta_j)$, $\phi(\delta_i)$ and δ_i have the same sign componentwise, we get $\dot{V}\le 0$. To this end, the overall auxiliary system is stable.

Note that $\dot{V} \equiv 0$ implies that $\delta_i - \delta_j = 0$ and $\delta_i = 0$ when the associated directed graph is symmetric, which, in turn, implies that $\delta_i = \delta_j$, $\forall i \neq j$.

From LaSalle's Invariance principle, it follows that $\delta_i \to \delta_j, \forall i \neq j$ asymptotically as $t \to +\infty$; that is, $\lim_{t\to+\infty}|\delta_i - \delta_j| = 0$ and $\lim_{t\to+\infty}|\delta_i| = 0$. That is to say, the overall auxiliary closed-loop system is asymptotically cooperative stable.

Furthermore, we can obtain $\lim_{t\to+\infty}|y_i(t) - y_0| = 0$, $\forall i \neq j$ since the fact is that $\delta_i = y_i - y_0$.

To this end, the current state of the all chargers can ultimately be consensus and track the desired reference current. Furthermore, the overall closed-loop system is asymptotically stable.

This completes the proof. □

4.2. Cooperative Stability Analysis under Dynamic Topology

Theorem 11. *Consider the multicharging systems* (5) *under dynamic communication topology. Under the Assumption 8 that the directed graph is symmetric at each time t and* $\rho_i > 0$ *for at least one charger, the current-sharing objective can be achieved under the proposed distributed time-varying cooperative control law* (21).

That is to say, the output current of the all chargers can ultimately be consensus and track the desired reference current. Furthermore, the overall closed-loop system is asymptotically cooperative stable.

Proof. Being similar to the fixed topology case, the Lyapunov function candidate for the auxiliary system (23) can be chosen as follows:

$$V = \frac{1}{2}\sum_{i=1}^{n}\delta_i^T\delta_i. \tag{34}$$

Differentiate the Lyapunov function V along the auxiliary subsystem's trajectory with respect to time t; $\dot{V}$ can be derived as follows:

$$\dot{V} = -c_i\sum_{i=1}^{n}\left(\delta_i^T\sum_{j\in N_i}a_{ij}(t)\,\phi\left(\delta_i - \delta_j\right) + \rho_i\phi\left(\delta_i\right)\right). \tag{35}$$

However, the time varying adjacent matrix $A(t)$ of the direct graph is symmetric at each time t; that is, $a_{ij}(t) = a_{ji}(t)$.

By virtue of Lemma 9, $\dot{V}$ can be transformed to be

$$\begin{aligned}\dot{V} = &-\frac{1}{2}c_i\sum_{i=1}^{n}\sum_{j=1}^{n}a_{ij}(t)\left(\delta_i - \delta_j\right)^T\phi\left(\delta_i - \delta_j\right)\\ &- c_i\sum_{i=1}^{n}\delta_i^T\rho_i\phi\left(\delta_i\right).\end{aligned} \tag{36}$$

Similar to Theorem 10, the function $y = \phi(x)$ is monotonically increasing. $\phi(x) \geq 0$ when $x \geq 0$, and $\phi(x) \leq 0$ when $x \leq 0$. That is to say, $\phi(x)$ and x have the same sign componentwise. And then, we can obtain that $\dot{V} \leq 0$. Consequently, the overall auxiliary systems is stable.

Table 1: The main parameters for the charging system model.

Charger i	L_i (mH)	r_i (Ω)	U_{d_i} (V)
1	3.05	0.0035	1335
2	3.12	0.0031	1272
3	2.95	0.0029	1295
4	3.01	0.0040	1371

Table 2: The simulation parameters for the super capacitor.

Parameter	R_1	C_0	C_v
Value	5.6 mΩ	92.3 F	$U_c * 0.0747$ F/V

Similarly to the proof of Theorem 10 with the help of LaSalle's Invariance principle, the overall auxiliary close-loop system is asymptotically stable provided that the direct dynamical communication topology is symmetric at each time interval. Furthermore, the cooperative objective can be achieved while the overall closed-loop system is asymptotically cooperative stable under the dynamic topology.

This completes the proof. □

5. Case Studies

In this section, we use the charging test system shown in Figure 2 to validate the feasibility of the proposed cooperative control scheme. Three different cases below are considered and compared in this section.

Case A. The charging current is chosen only to be 450 A in the whole charging process.

Case B. The whole charging process includes two sequential phases, namely, a fast charging phase and a trickle charging phase. At the fast charging phase, the charging current is chosen to be 450 A. At the trickle charging phase, the charging current is chosen to be 100 A.

Case C. On the basis of Case B, the coupling strength in the cooperative current-sharing controller can be tuned to be much larger in order to improve the response speed.

The main parameters for the charging systems model are given in Table 1. The simulation parameters for the supercapacitor are shown in Table 2. The initial current of the chargers is 0 A. In Case B, the supercapacitor's initial voltage U_c is 500 V, nominal voltage is 900 V, and nominal charging current is 450 A and 100 A, which means that the supercapacitor's residual voltage is 500 V when the energy storage type light rail vehicle reaches the platform and the supercapacitor should be charged to be about 870 V by 4 chargers in parallel with each other by choosing a 450 A charging current in a very short period of time, and then the charging current is turned to be 100 A, which charges the supercapacitor to 900 V.

The given communication topology between each charging subsystem is shown in Figure 6.

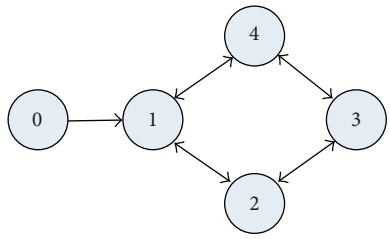


Figure 6: The given communication topology between each charging subsystem.

The specific final controller in the simulation is formulated below:

$$u_i = \left(c_i \left(\sum_{j \in N_i} a_{ij} \phi\left(y_j - y_i\right) + \rho_i \phi\left(y_0 - y_i\right) \right) - \left(-\frac{r_i}{L_i} y_i - \frac{U_c}{L_i} \right) \right) L_i = c_i \left(\sum_{j \in N_i} a_{ij} \phi\left(y_j - y_i\right) + \rho_i \phi\left(y_0 - y_i\right) \right) L_i + r_i y_i + U_c, \tag{37}$$

where r_i, L_i are given in Table 1, $\rho_1 = 1$, and $\rho_2 = \rho_3 = \rho_4 = 0$. In Case A and Case B, the coupling strengths are $c_1 = 2.25$, $c_2 = 2.5$, $c_3 = 2.75$, and $c_4 = 3$, while $c_1 = 4.25$, $c_2 = 4.5$, $c_3 = 4.75$, and $c_4 = 5$ in Case C. In Case A, the current-sharing object $y_0 = 450$ A. In Case B and Case C, the current-sharing object $y_0 = 450$ A in the fast charging phase and $y_0 = 100$ A in the trickle charging phase. The initial current $y_i(0) = 0$ A, $i = 1, 2, 3, 4$, the supercapacitor's initial voltage $U_c(0) = 500$ V in all Cases. The entry a_{ij} of the adjacency matrix A for the given communication topology in Figure 6 is

$$A = \begin{bmatrix} 0 & 1.25 & 0 & 1.25 \\ 1.25 & 0 & 1.5 & 0 \\ 0 & 1.5 & 0 & 1.75 \\ 1.25 & 0 & 1.75 & 0 \end{bmatrix}. \tag{38}$$

The specific function expression for the saturation function in Figures 7–11 is given as follows:

$$\phi(x) = \begin{cases} a, & x \geq a \\ x, & x \in [-a, a] \\ -a, & x \leq -a, \end{cases} \tag{39}$$

where a is parameter which can character the upper and lower condition of the saturation function. Here, $a = 300$ in the simulation of Figures 7–11.

When the coupling strength c_i is chosen to be $c_1 = 2.25$, $c_2 = 2.5$, $c_3 = 2.75$, and $c_4 = 3$, the current-sharing curve for Case A under the communication topology in Figure 6 is plotted in Figure 7, which illustrates that the current has been balanced and the consensus and cooperative objective have been accomplished.

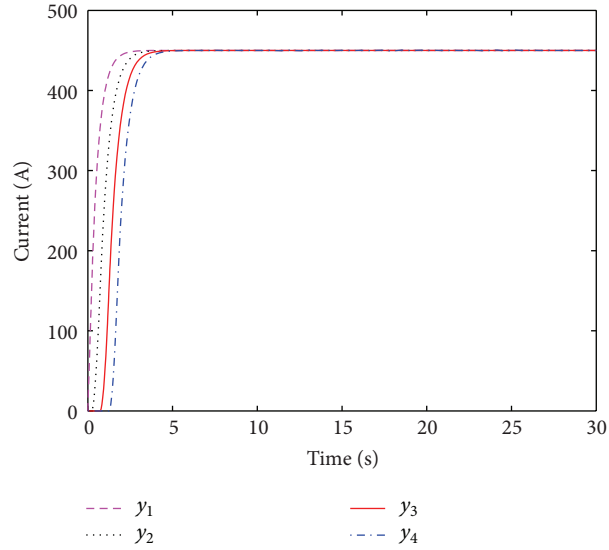


Figure 7: The current-sharing curve for Case A under the given communication topology.

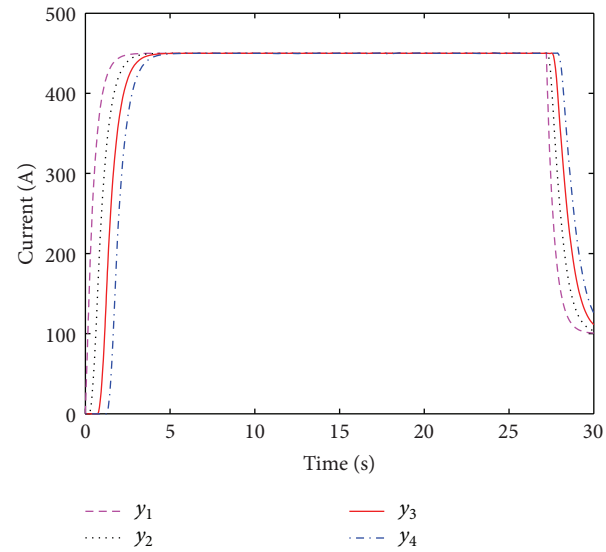


Figure 8: The current-sharing curve for Case B under the given communication topology.

As shown in Figure 8 for Case B, the charging current ascends to be 450 A from 0 A within 4 s, and then it goes to be constant current charging stage until the supercapacitor is charged to be about 870 V. Since the supercapacitor's voltage will decline at the end of charging stage, as shown in Figure 9(a), if we always choose the large charging current in the whole charging process as Case A, the charging current in Case B is then reduced to 100 A at the end of charging stage. Such a stage is called the trickle charging phase. As Figure 9(b) shows, the voltage is charged fully from 500 V to about 900 V in less than 30 second, illustrating that voltage objective has been achieved by adopting the proposed control strategy in Case B.

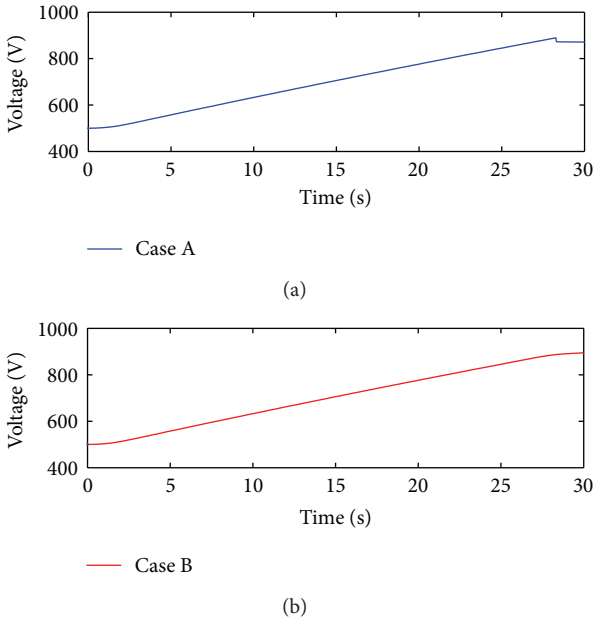


FIGURE 9: The charging voltage curve of the whole charging system under the given communication topology, (a) Case A and (b) Case B.

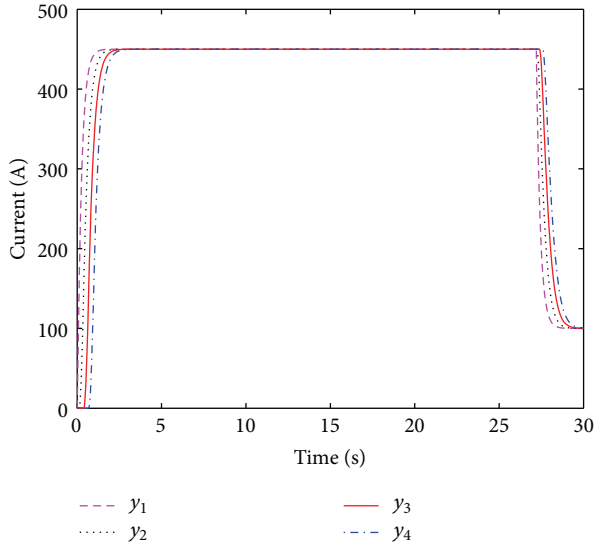


FIGURE 10: The current-sharing curve for Case C under the given communication topology with coupling strengths $c_1 = 4.25$, $c_2 = 4.5$, $c_3 = 4.75$, and $c_4 = 5$.

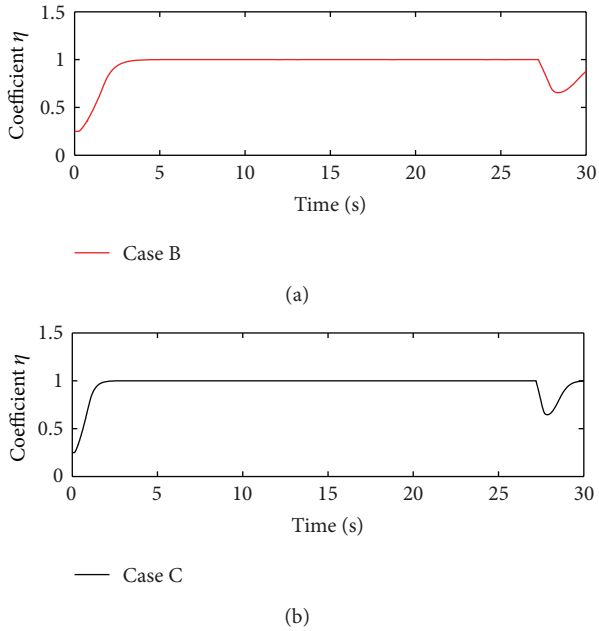


FIGURE 11: The current-sharing coefficient curve under the proposed cooperative control scheme with different coupling strengths in (a) Case B, $c_1 = 2.25$, $c_2 = 2.5$, $c_3 = 2.75$, and $c_4 = 3$ and (b) Case C, $c_1 = 4.25$, $c_2 = 4.5$, $c_3 = 4.75$, and $c_4 = 5$.

Comparing Figure 9(a) with Figure 9(b), the supercapacitor's voltage will decline about 18 V in Case A as shown in Figure 9(a) if the charging current is chosen only to be 450 A. On the contrary, if we choose 450 A as the charging current at the first stage and 100 A at the last stage in Case B, the supercapacitor can be charged fully to be 900 V as shown in Figure 9(b).

The current-sharing coefficient curves in Case B and Case C under the proposed cooperative control scheme are plotted in Figure 11, which is used to measure the balance of the current. When the coupling strengths are chosen to be larger as in Case C, the current-sharing coefficient converges to be 1 as shown in Figure 11(b), which is much better than that of Case B as shown in Figure 11(a).

Comparing Case B with Case C, which are shown in Figures 8 and 10, the convergence performance will be better if the coupling strengths are chosen to be larger. In a word, we can conclude that if we choose a larger coupling strength, the convergence performance will be improved, which will unfortunately result in a larger ripple peak value at the same time.

Next, simulation comparison between the proposed cooperative current-sharing control and the proportional integral (PI) control is given below. Without loss of generality, we consider the most basic charging scene, Case A. The initial currents are $y_1(0) = 50$ A, $y_2(0) = 45$ A, $y_3(0) = 60$ A, and $y_4(0) = 55$ A and the supercapacitor's initial voltage is $U_c(0) = 500$ V in such two schemes.

The charging curve for Case A is given in Figure 12 by choosing PI control law. The proportional control gains $k_p = 0.2$ and the integral control gains $k_i = 0.13$, which are the same for all chargers. Although the control objects are known to each charger, they do not interact with each other. The current-sharing result is not well sometimes as shown in Figure 12. In addition, there exist overshoot and static error by using the PI current-sharing control approach. Too large overshoot and static error are not allowed in the charging system. The initial charging current will be very large if the existing overshoot is too large. Not only the supercapacitor

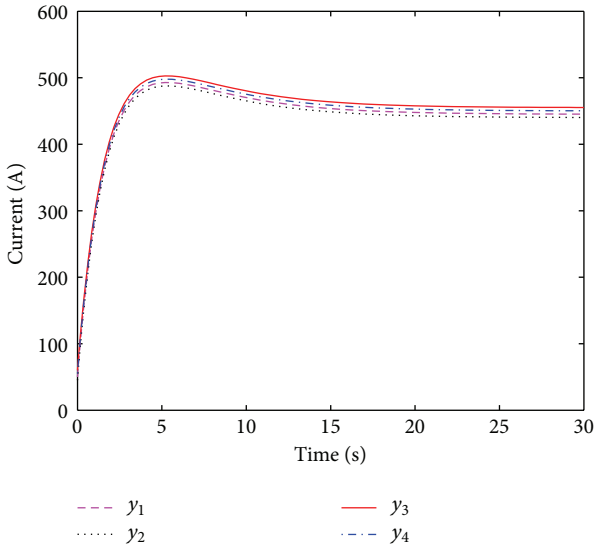


FIGURE 12: The current-sharing curve for Case A under the PI control law.

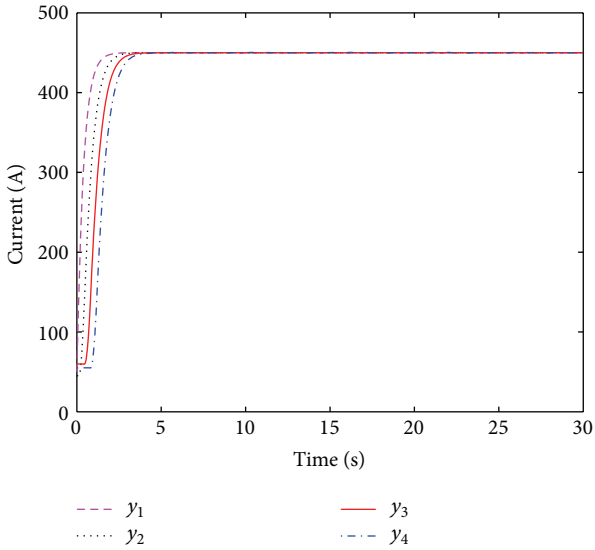


FIGURE 13: The current-sharing curve for Case A under the cooperative control law.

and charger will be damaged, but also the operator may be endangered.

But in the proposed cooperative control scheme, each charger can communicate with each other through the network and the control objective is just needed to be preconfigured in one charger. The current-sharing result is shown in Figure 13 when the initial current is not zero, where $c_1 = 2.65$, $c_2 = 2.9$, $c_3 = 3.15$, and $c_4 = 3.4$. There is not any overshoot and static error by using the proposed cooperative current-sharing approach compared with PI control method within the same rising time t_r.

6. Conclusions

In this paper, the distributed cooperative control of the nonlinear and nonidentical multiagent systems is adopted to carry out current-sharing strategy for charging system of energy storage type light rail vehicle, that addresses the current balance problem. Input-output feedback linearization is introduced to convert the distributed cooperative current-sharing control of multicharging with nonidentical subsystem to a first-order integrator consensus problem. The cooperative current-sharing control law is put forward by introducing a general saturation function. The proposed cooperative current-sharing strategy is distributed; each distributed charging subsystem only needs its own information and the information of some neighbors. The whole closed-loop charging system is proved to be cooperative stable when the directed graph is symmetric. The nonidentical feature of the charging system can be removed by the proposed control law, thus ensuring that the overall system is stable and the cooperative current-sharing objective can be achieved. The response speed can be improved by tuning current-sharing control parameters. Case studies show that our current-sharing approach is effective and feasible.

Conflict of Interests

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

Acknowledgment

This work was supported by the National Natural Science Foundation of China (nos. 61071096, 61003233 and 61073103) and Specialized Research Fund for the Doctoral Program of Higher Education (nos. 20100162110012 and 20110162110042).

References

[1] W. Chen, X. Ruan, H. Yan, and C. K. Tse, "DC/DC conversion systems consisting of multiple converter modules: stability, control, and experimental verifications," *IEEE Transactions on Power Electronics*, vol. 24, no. 6, pp. 1463–1474, 2009.

[2] B. K. De, B. Bolsens, J. van den Keybus, A. Woyte, J. Driesen, and R. Belmans, "A voltage and frequency droop control method for parallel inverters," *IEEE Transactions on Power Electronics*, vol. 22, no. 4, pp. 1107–1115, 2007.

[3] J. B. Wang, "Study of cable resistance and remote-sensing scheme in parallel DC/DC converter system via primary droop current-sharing control," *IET on Power Electronics*, vol. 5, no. 6, pp. 885–898, 2012.

[4] S. K. Mazumder, M. Tahir, and K. Acharya, "Master-slave current-sharing control of a parallel DC-DC converter system over an RF communication interface," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 1, pp. 59–66, 2008.

[5] P. Li and B. Lehman, "A design method for paralleling current mode controlled DC-DC converters," *IEEE Transactions on Power Electronics*, vol. 19, no. 3, pp. 748–756, 2004.

[6] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination,"

IEEE Transactions on Industrial Informatics, vol. 9, no. 1, pp. 427–438, 2013.

[7] 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.

[8] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control Systems Magazine*, vol. 27, no. 2, pp. 71–82, 2007.

[9] G. Antonelli, "Interconnected dynamic systems: an overview on distributed control," *IEEE Control Systems Magazine*, vol. 33, no. 1, pp. 76–88, 2013.

[10] Z. Qu, J. Wang, and R. A. Hull, "Cooperative control of dynamical systems with application to autonomous vehicles," *IEEE Transactions on Automatic Control*, vol. 53, no. 4, pp. 894–911, 2008.

[11] H. Xin, Z. Qu, J. Seuss, and A. Maknouninejad, "A self-organizing strategy for power flow control of photovoltaic generators in a distribution network," *IEEE Transactions on Power Systems*, vol. 26, no. 3, pp. 1462–1473, 2011.

[12] A. Bidram, A. Davoudi, F. L. Lewis, and J. M. Guerrero, "Distributed cooperative secondary control of microgrids using feedback linearization," *IEEE Transactions on Power System*, vol. 28, no. 3, pp. 3462–3470, 2013.

[13] R. Diestel, *Graph Theory*, vol. 173 of *Graduate Texts in Mathematics*, Springer, New York, NY, USA, 1997.

[14] W. Ren, "Consensus tracking under directed interaction topologies: algorithms and experiments," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 1, pp. 230–237, 2010.

[15] H. Zhang, F. L. Lewis, and Z. Qu, "Lyapunov, adaptive, and optimal design techniques for cooperative systems on directed communication graphs," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 7, pp. 3026–3041, 2012.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 730652, 8 pages
http://dx.doi.org/10.1155/2014/730652

Research Article

Matthew Effect of the Random Drift on the Evolution of Cooperation

Chao Liu[1] and Rong Li[2]

[1] *Department of Applied Mathematics, Shanghai University of Finance and Economics, Shanghai 200433, China*
[2] *Department of Applied Mathematics, Shanghai Finance University, Shanghai 201209, China*

Correspondence should be addressed to Chao Liu; liu.chao@shufe.edu.cn

Received 22 November 2013; Accepted 7 January 2014; Published 16 February 2014

Academic Editor: Wenwu Yu

The effect of the random drift on the evolutionary prisoner's dilemma game is studied on regular lattices. A new evolutionary rule is proposed, which stochastically combines the deterministic rule with the random drift rule. It is found that the random drift has an effect on the evolutionary dynamics depending on the values of the temptation-to-defect b and the probability p of the random drift. When the random drift occurs with low probabilities, which interests us more, a phenomenon of the Matthew effect on the evolution of cooperation is found. Explanations of this phenomenon are deduced through the analysis on the dynamics and pattern formations of the PDG system.

1. Introduction

Since the introduction of evolutionary game theory by Smith and Price [1, 2], one of the important issues is to understand the emergence and evolution of cooperation [3–6]. Among the many games, the most studied example is the prisoner's dilemma game (PDG), as it provides a simple example of the difficulties of cooperation [7–10]. The standard PDG is described by the following set of rules. When two players play a PDG, each of them can choose to cooperate (C) or defect (D). Each player will gain a payoff depending jointly on his choice and the opponent's choice. A cooperator receives R when playing with a cooperator and S when playing with a defector while a defector earns P when playing with a defector, and T against a cooperator, with $T > R > P > S$. Given this payoff ordering, in a well-mixed (unstructured) population where each agent interacts with all other agents (or a representative sample of the population composition), defectors are fitter and thus the fraction of cooperators asymptotically vanishes.

However, cooperation is widespread in many biological, social, and economic systems [11, 12]. One of the proposed mechanisms to explain this phenomenon is network reciprocity [13, 14]. For example, early pioneering work on the PDG in two-dimensional square lattices made the observation that, unlike in unstructured populations, cooperators and defectors can coexist in the lattice indefinitely [15, 16]. Simply said, the clustering of cooperators in the lattice could provide high enough fitness to the cooperator nodes exposed to invasion, to the extent of preserving cooperators from evolutionary extinction, even when defection is favoured by the one-shot (two-players) game analysis [17]. A recent and powerful approach to studying such questions is provided by evolutionary graph theory [18].

All the game models incorporate some kind of evolutionary dynamics, which also play a crucial role in the results. As for [15, 16], the well-known promotion of cooperation enforced by spatial lattices is linked to a particular best-takes-over update rule, in which each individual node plays with its immediate neighbours each time step accumulating a payoff; then it updates its strategy by imitating the one of the highest payoff in its neighbourhood, including itself.

The best-takes-over update rule is a nonstochastic imitation strategy. However, for real dynamical systems, external disturbances or system errors are generally inevitable. Therefore, it is worth considering the dependence of the promotion of cooperation on the evolutionary rules and its robustness against perturbations. Indeed, previous work has pursued this

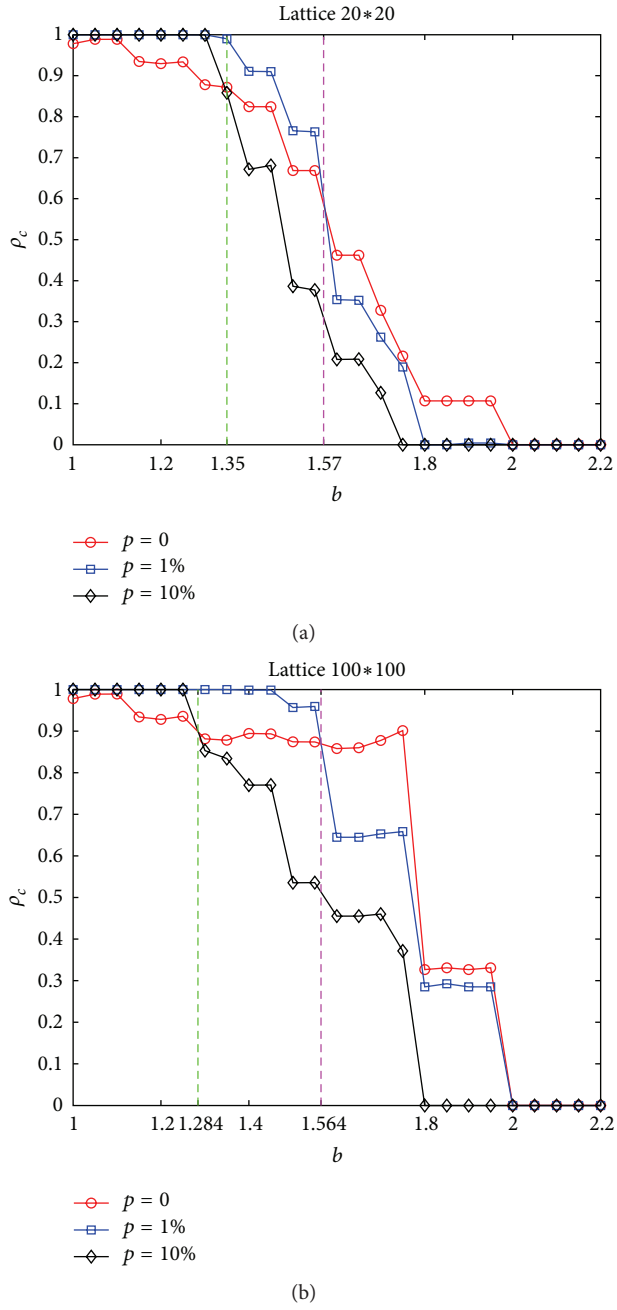


Figure 1: The frequency of cooperators, ρ_c, as a function of the temptation-to-defect b in an evolutionary PDG on both 20 ∗ 20 and 100 ∗ 100 regular 8-neighbored lattices with periodic boundary conditions.

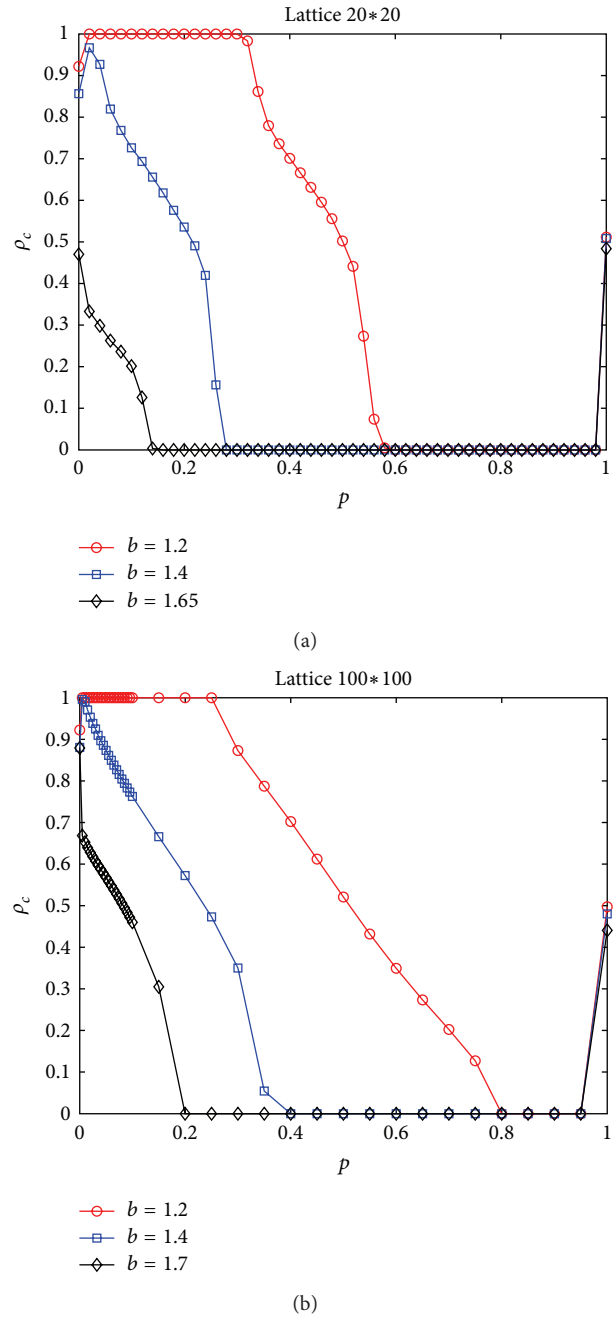


Figure 2: The frequency of cooperators as a function of the probability p in an evolutionary PDG on 20 ∗ 20 and 100 ∗ 100 regular 8-neighbored lattices with periodic boundary conditions.

enquiry [19–21], replacing the deterministic update strategy with some stochastic imitative rules, finding greatly reduced cooperation levels.

In these game models, players are viewed as rational, who update their strategy by copying, within certain constraints, the strategy of those others that are doing better, or in game theoretical terms that are obtaining higher payoffs from the game. Then, what difference will be there in the evolution of cooperation if players occasionally behave irrationally, updating strategy with no concern of their payoffs. In this paper, we will explore this problem by combining the best-takes-over rule with the random drift reproduction rule in the prisoner's dilemma game.

2. The Model

We study the PDG with pure strategies: the players can either defect (D) or cooperate (C). The same as in [15, 16], players are disposed on two-dimensional square lattices, and each player interacts only with its nearest neighbors and collects profits

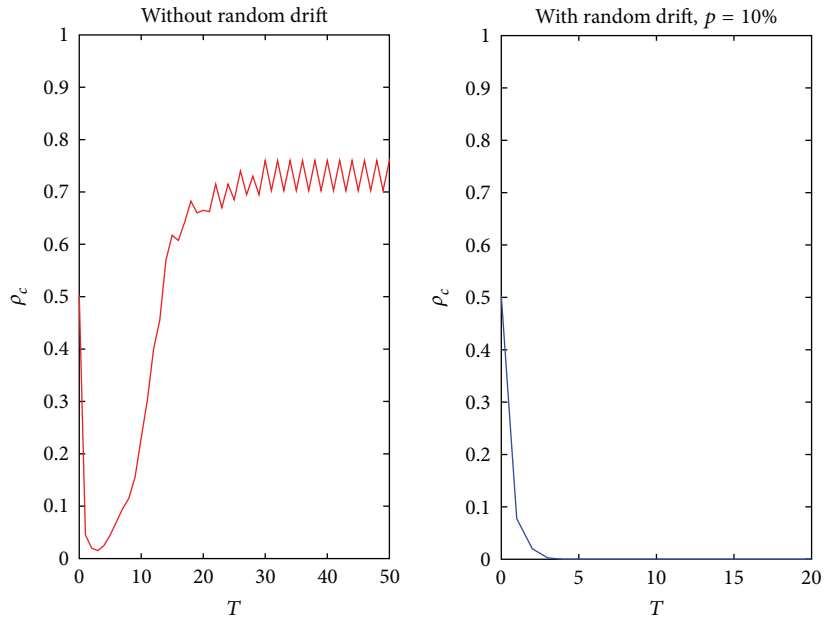


Figure 3: The dynamics of the PDG system with or without the influence of the random drift on a 20 ∗ 20 regular lattice. The temptation-to-defect is $b = 1.65$. All the simulations start from the same random initial configuration.

depending on the payoff parameters. Following common studies, the PDG is rescaled such that it depends on a single parameter; that is, the parameters are chosen to be $R = 1$, $P = S = 0$, and $T = b$, representing the advantage of defectors over cooperators (or the temptation-to-defect). After each round, the players are allowed to inspect their neighbors' payoffs and, according to the comparison, determine their strategies to adopt in the next round. Two alternative update rules determining the transformation of each player's strategy are described below.

(I) Best-takes-over: in each generation, an individual node imitates the strategy of one of its neighbors (including the node itself) that received the highest payoff in the last round. The best-takes-over rule is a deterministic rule according to which the individual with the highest gain in a given neighborhood reproduces with certainty.

(II) Random drift: whenever an individual node i is updated, one neighbor j is drawn at random among all its neighbors (including node i itself), and node i imitates the strategy of node j with certainty regardless of the payoff of node i or j in the last round. Thus, for the random drift update rule, reproduction ability or fitness is not correlated with payoff but is correlated with the proportion of one strategy in the neighborhood.

To investigate how the random drift affects evolutionary games, we propose a new evolutionary rule which stochastically combines the deterministic rule with the random drift rule. A parameter of probability $p \in [0, 1]$ is used in the choosing of the two alternative rules. That is, in each generation, an individual node updates its strategy by the best-takes-over rule with probability $1 - p$ and by the random drift rule with probability p.

Our simulations are carried out on the regular 8-neighbored square lattices. Initially, the cooperative and defective strategies are randomly distributed among the players with equal probability 1/2. A synchronous updating scheme is adopted. Equilibrium frequencies of cooperators are obtained by averaging over 1000 generations after a transient time of 50000 generations. Each data point results from an average over 100 realizations of the initial conditions.

3. Results and Discussion

Figure 1 shows the varying frequency of cooperation versus temptation-to-defect b on $20 * 20$ and $100 * 100$ regular lattice, respectively. ρ_c represents the density of cooperators. The random drift update rule is employed with different probabilities p, where $p = 0$ means the pure deterministic imitative dynamics. As shown in Figure 1, the random drift has obvious influence on the evolution of cooperation. Moreover, the influence depends on the value of the parameter b.

Despite the huge gap of the two lattices in size, the simulation results are qualitatively similar. In the following, we mainly analyze the results on the $20 * 20$ lattice as an example. Based on the different influence of the random drift, the range of parameter b can be roughly divided into three regions. The first region is the small value region $b \in (1, 1.35)$. In this parameter region, the phenomenon of stochastic resonance is observed. For small temptation-to-defect b, the frequency

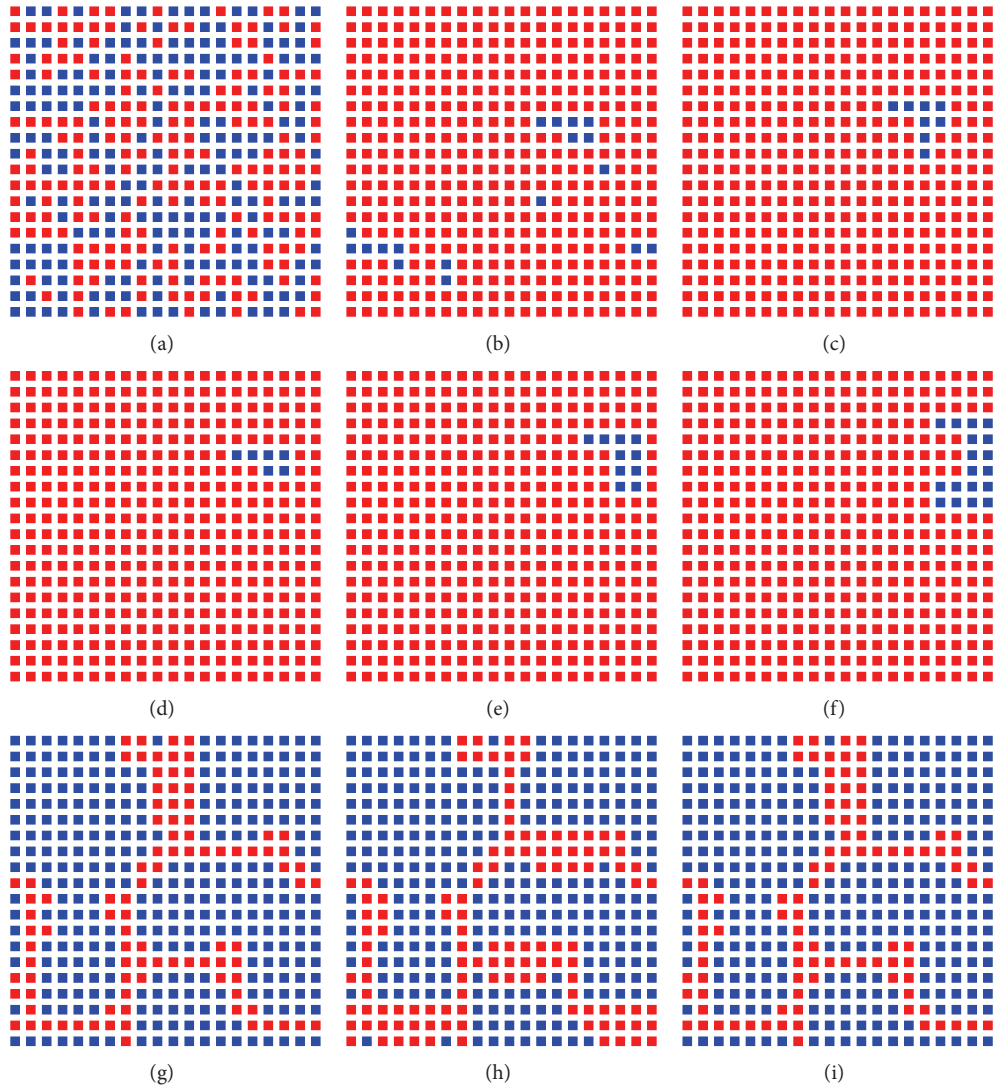


Figure 4: The pattern formation of the PDG system without the influence of the random drift on a $20 * 20$ regular lattice. The temptation-to-defect is $b = 1.65$. Color code: cooperators blue, defectors red.

of cooperation is high under the pure best-takes-over mechanism. However, it is still raised when the random drift is applied with small probabilities, even to an all-C state. The second parameter region is the moderate value region $b \in (1.35, 1.57)$. In this region, whether the random drift favours cooperation depends on the value of the probability p. When the random drift update rule is applied with probability $p = 1\%$, the density of cooperators is higher than the one with the pure deterministic rule. However, when the probability rises to $p = 10\%$, the frequency of cooperation decreases lower than the deterministic one. $b \in (1.57, 2.2)$ is the large parameter region. In this region, the evolutionary survival of cooperators is hindered by the random drift update rule. In particular, the parameter region $b \in (1.8, 2)$ is worth of more attention. It is the chaotic parameter region under the deterministic mechanism, where the cooperators and defectors can coexist indefinitely; see [16] for details. As the random drift is applied, the chaotic evolution of the dynamic system is broken down, leading to an all-D state.

To have a more clear knowledge of the influence of the random drift on the evolution of cooperation, the variation of ρ_c as a function of the probability p is given in Figure 2. Because of the qualitatively similarity, we still focus on the results of the $20 * 20$ lattice. The three values of the temptation-to-defect b are from the three parameter regions mentioned above. As shown in Figure 2, for all the three values of b, the random drift is not favourable to the maintenance of cooperation when the probability p is large enough. In particular, an all-D state prevails to form a flat valley in each of the three curves. When p reaches 100%, the equilibrium frequency of cooperation fluctuates around the original frequency (50%). It is because, in this situation, the PDG evolves entirely under the random drift mechanism, for which payoffs of nodes do not count and the chance of survival for cooperators and defectors are equal. In the case $b = 1.65$, the frequency of cooperation decreases monotonously with the increasing probability p of the random drift. In the case $b = 1.2$ or $b = 1.4$, where b is relatively small, an earlier raised and

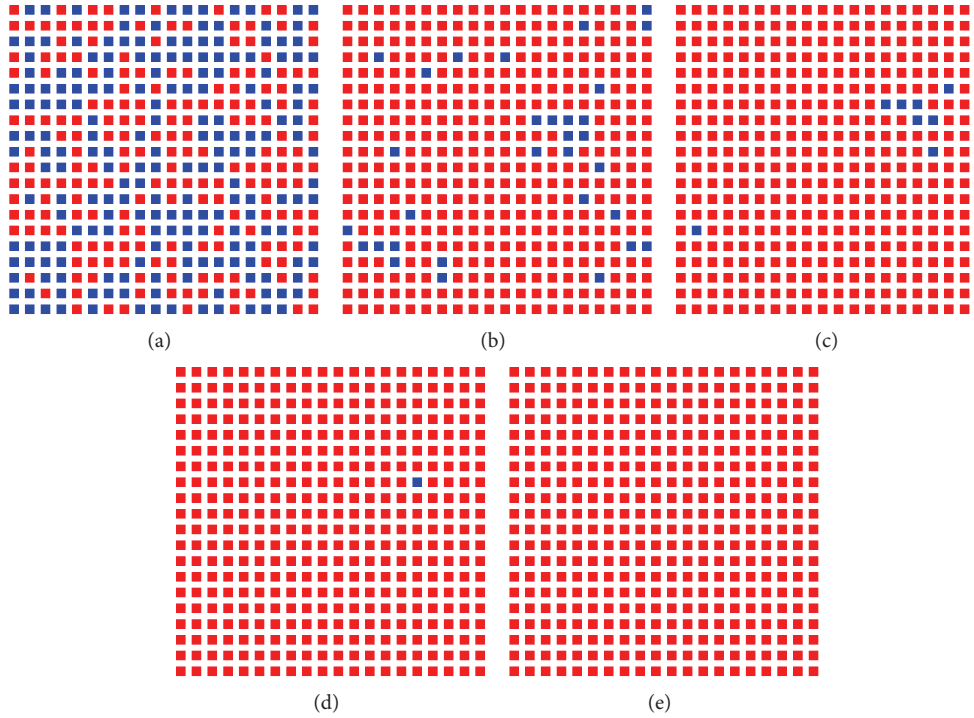


FIGURE 5: The pattern formation of the PDG system with the influence of the random drift on a $20 * 20$ regular lattice. $b = 1.65$ and $p = 10\%$. Color code: cooperators blue, defectors red.

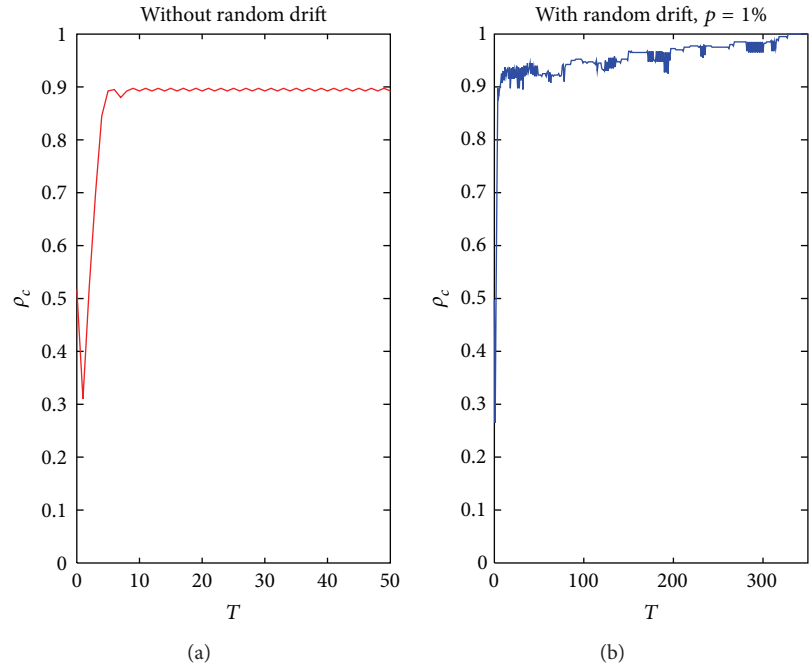


FIGURE 6: The dynamics of the PDG system with or without the influence of the random drift on a $20 * 20$ regular lattice. The temptation-to-defect is $b = 1.2$. All the simulations start from the same random initial configuration.

later decreased frequency of cooperation is found. Similar results are found on the $100 * 100$ lattice.

Thus, in fact, the random drift has an influence on the evolutionary PDG depending on the values of the temptation-to-defect b and the probability p. When p is high, which means that the random drift occurs frequently, it tends to depress cooperation for all values of b. When the random drift occurs only with low probabilities, which interests us more, a conclusion can be drawn as follows: when the temptation for players to defect is relatively small, with cooperators prevailing in the original PDG, the applied random drift tends to further enhance cooperation; under the opposite

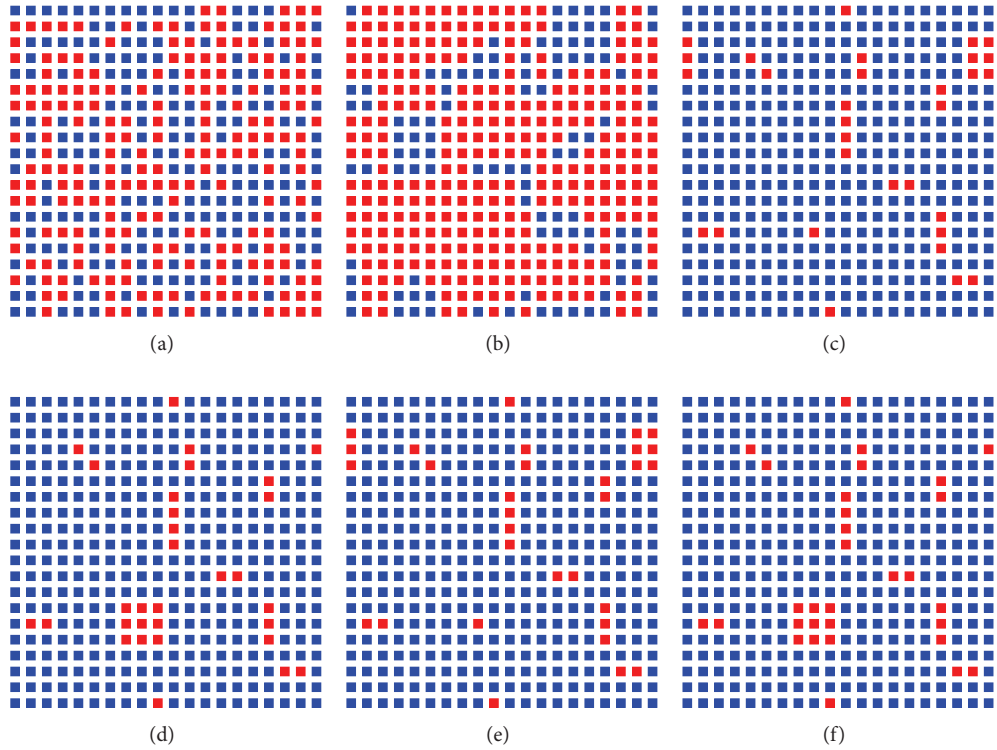


Figure 7: The pattern formation of the PDG system without the influence of the random drift on a 20 ∗ 20 regular lattice. The temptation-to-defect is $b = 1.2$. Color code: cooperators blue, defectors red.

conditions which is cooperation unfavourable, the random drift tends to further depress cooperation. In other words, for the cases with small p, the effect of the random drift on the evolution of cooperation is the so-called Matthew effect, which makes the rich richer.

How come this kind of Matthew effect arises when the random drift is applied to the deterministic imitative mechanism with small probability? The dynamics and the pattern formation of the PDG system may shed light on the explanation.

Setting $b = 1.65$ and starting from the same initial distribution of cooperators, Figure 3 displays the dynamics of the PDG system with and without the influence of the random drift on the 20 ∗ 20 lattice. Under the pure deterministic imitation mechanism, the system settles at a period 2 oscillator after a short-term fluctuation as shown in Figure 3(a). In Figure 3(b), when the random drift update rule is applied with a probability $p = 10\%$, the cooperation is obviously suppressed and the system soon converges to an all-D state.

The corresponding pattern formations of the above two evolutionary processes are displayed in Figures 4 and 5, respectively. Figure 4(a) shows the initial pattern formation of the PDG system, in which cooperators and defectors randomly distributed in the lattice with equal probability. At the second time generation, the density of cooperators drops dramatically due to the high temptation to defectors. After that, the frequency of cooperators grows up gradually for the reason that the square shaped C-clusters can survive and even grow in this region of parameter b, as displayed in Figures 4(b)–4(f). At last, the system regularly switches between two different pattern formations as shown in Figures 4(g)–4(i), corresponding to the period 2 oscillator in Figure 3(a). As for Figure 5, although starting from the same initial configuration, cooperators vanish soon. From Figure 5(b) to Figure 5(c), one node (node (7, 17)) of a 2×2 square shaped C-cluster turns to a defector due to the effect of the random drift, which leads to the breakdown of the entire square shaped C-cluster and eventually to the extermination of all cooperators.

The dynamics of the PDG system with temptation-to-defect $b = 1.2$ is displayed in Figure 6. Under the deterministic imitative mechanism, the dynamical system quickly settles at a period 2 oscillator with a high average equilibrium frequency of cooperators. When the random drift update rule is employed with a probability $p = 1\%$, the phenomenon of intermittent period oscillations appears in the dynamics, where the period oscillators become unstable due to the influence of the randomness. Finally, the system reaches an all-C state.

Figures 7 and 8 illustrate the dynamics of Figure 6 with pattern formations. As shown in Figures 7(c)–7(f), for the small value of b, defectors can only survive and remain stable in straight or oblique lines. At the same time, as shown at the bottom-left parts of Figures 7(c)–7(f), a single defector ($1D$) will grow to form a 3×3 square shaped D-cluster ($9D$) and then return to a single defector ($1D$) in the next time generation, which explains the period 2 oscillation of the dynamical system in Figure 7. However, when the random drift is applied with probability $p = 1\%$, both the stable lines

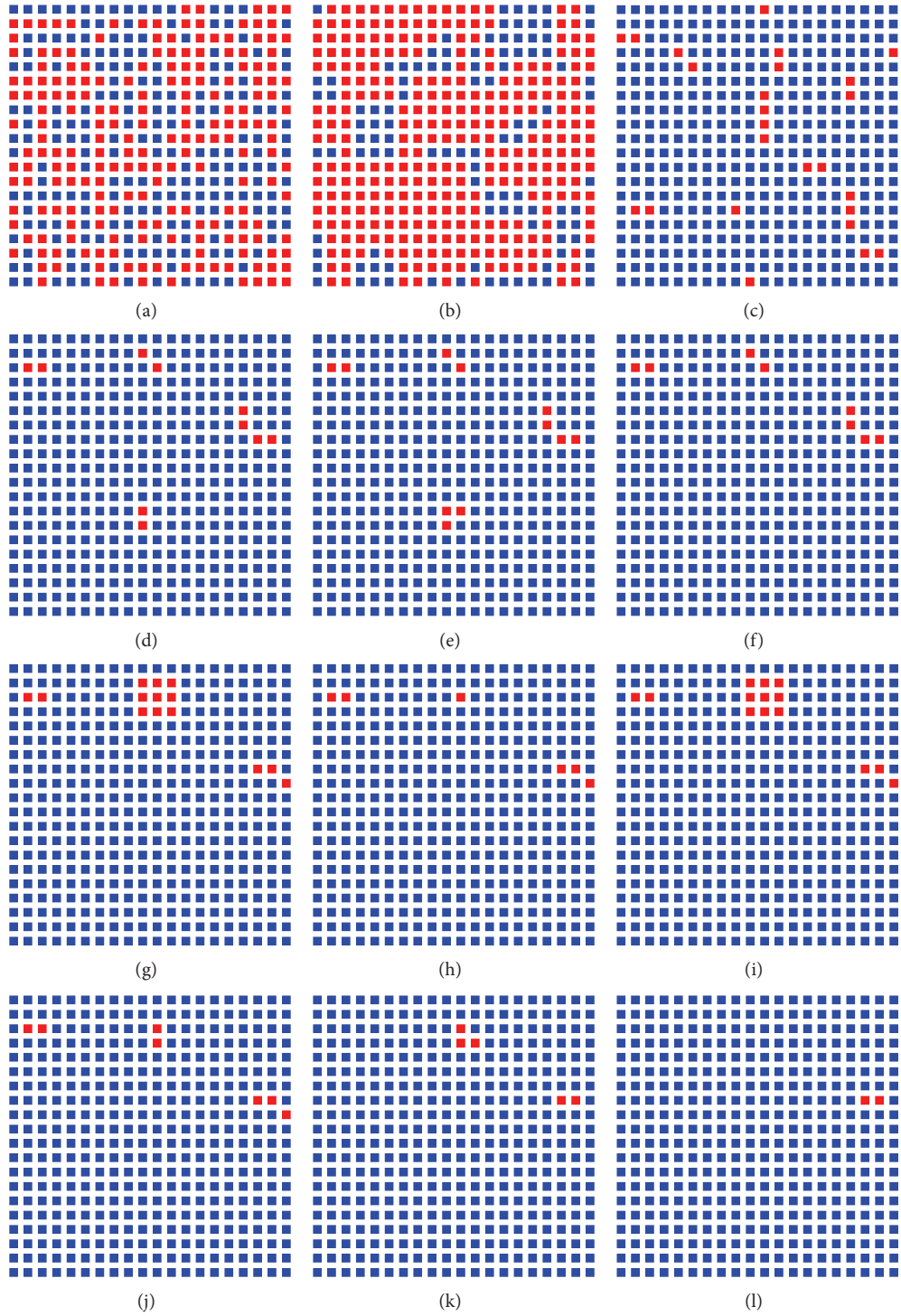


Figure 8: The pattern formation of the PDG system with the influence of the random drift on a $20 * 20$ regular lattice. $b = 1.2$ and $p = 1\%$. Color code: cooperators blue, defectors red.

and the $1D \rightarrow 9D \rightarrow 1D$ oscillators of defectors can be violated by the randomness, resulting in the enhancement of cooperators.

At the beginning, the two systems have similar pattern formations as shown in Figures 7(a)–7(c) and Figures 8(a)–8(c). Figures 8(d)–8(f) show a process of the breakdown of a line shaped D-cluster. In Figure 8(d), node (13, 10) and node (14, 10) form a line shaped D-cluster, which is stable under the deterministic mechanism. Due to the random drift update strategy, node (13, 11), a neighbour of this line shaped D-cluster, turns from a cooperator to a defector. Then an arrow shaped D-cluster is formed in Figure 8(e), which is vulnerable under the deterministic imitative mechanism and eaten up by cooperators as shown in Figure 8(f). Figures 8(g)–8(l) display

the vanishing of a $1D \rightarrow 9D \rightarrow 1D$ oscillator of defectors. In Figures 8(g)–8(i), a $1D \rightarrow 9D \rightarrow 1D$ oscillator of defectors is shown at the upper center. In Figure 8(j), affected by the randomness, the square shaped D-cluster turns to a line shaped D-cluster with two defectors instead of a single defector, which terminates the $1D \rightarrow 9D \rightarrow 1D$ oscillation of defectors. After that, the line shaped D-cluster remains stable for several time generations until it turns to an arrow shaped D-cluster and vanishes in the next time generation as shown in Figures 8(k)–8(l).

Now we are ready to explain the Matthew effect of the random drift on the evolutionary PDG. Despite the population of its opponent, some particular shaped C-clusters and D-clusters can survive and maintain under the deterministic imitative mechanism. However, these equilibrium states strictly depend on the shapes of these clusters and thus are vulnerable to disturbance. The randomness brought by the employed random drift update rule can change the shapes of these clusters, leading to the breakdown of these clusters and the phenomenon of the Matthew effect.

4. Conclusions

Evolutionary dynamics are affected by population structure and update rules. Spatial or network structure facilitates the clustering of strategies, which represents a mechanism for the evolution of cooperation. However, whether the evolutionary dynamics is robust to disturbance deserves further studies. In this paper, the problem is explored by combining the deterministic imitative rule with the random drift reproduction rule in the prisoner's dilemma game on regular lattices. It is found that the employed random drift has an effect on the evolutionary dynamics depending on the values of the temptation-to-defect b and the probability p. We are more interested in the cases where the random drift occurs only with low probabilities. Then, the random drift has an apparent Matthew effect on the evolution of cooperation: when the temptation for players to defect is relatively small, with cooperators prevailing in the original PDG, the applied random drift tends to further enhance the frequency of cooperators; under the opposite conditions which is cooperation unfavourable, the random drift tends to further depress cooperation. Analysis on the dynamics and pattern formation of the PDG system implies that the advantage brought by the clustering of strategies is diluted by the random drift, leading to the phenomenon of the Matthew effect.

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 Dr. Yu-Zhong Chen and Professor Ying-Cheng Lai for helpful discussions. This work was supported by the National Science Foundation of China under Grant no. 11101256 and Research and Innovation Project of Shanghai Municipal Education Commission (no. 14YZ149).

References

[1] J. M. Smith and G. R. Price, "The logic of animal conflict," *Nature*, vol. 246, no. 5427, pp. 15–18, 1973.

[2] J. M. Smith, *Evolution and the Theory of Games*, Cambridge University Press, Cambridge, UK, 1982.

[3] F. C. Santos and J. M. Pacheco, "Scale-free networks provide a unifying framework for the emergence of cooperation," *Physical Review Letters*, vol. 95, no. 9, Article ID 098104, 2005.

[4] R. Boyd, H. Gintis, and S. Bowles, "Coordinated punishment of defectors sustains cooperation and can proliferate when rare," *Science*, vol. 328, no. 5978, pp. 617–620, 2010.

[5] Y. Z. Chen and Y. C. Lai, "Optimizing cooperation on complex networks in the presence of failure," *Physical Review E*, vol. 86, Article ID 045101, 5 pages, 2012.

[6] S. Coakley, M. A. Nowak, J. Almenberg et al., *Evolution, Games, and God: the Principle of Cooperation*, Harvard University Press, 2013.

[7] R. Axelrod, *The Evolution of Cooperation*, Basic Books, New York, NY, USA, 1984.

[8] H. Gintis, *Game Theory Evolving*, Princeton University, Princeton, NJ, USA, 2000.

[9] S. Meloni, A. Buscarino, L. Fortuna et al., "Effects of mobility in a population of prisoner's dilemma players," *Physical Review E*, vol. 79, no. 6, Article ID 067101, 2009.

[10] M. Perc and Z. Wang, "Heterogeneous aspirations promote cooperation in the prisoner's dilemma game," *PLoS ONE*, vol. 5, no. 12, Article ID e15117, 2010.

[11] J. M. Smith and E. Szathmáry, *The Major Transitions in Evolution*, Oxford University Press, Oxford, UK, 1995.

[12] E. Pennisi, "How did cooperative behavior evolve," *Science*, vol. 309, no. 5731, p. 93, 2005.

[13] M. A. Nowak, "Five rules for the evolution of cooperation," *Science*, vol. 314, no. 5805, pp. 1560–1563, 2006.

[14] M. Perc and A. Szolnoki, "Coevolutionary games-A mini review," *BioSystems*, vol. 99, no. 2, pp. 109–125, 2010.

[15] M. A. Nowak and R. M. May, "Evolutionary games and spatial chaos," *Nature*, vol. 359, no. 6398, pp. 826–829, 1992.

[16] M. A. Nowak and R. M. May, "The spatial dilemmas of evolution," *International Journal of Bifurcation and Chaos*, vol. 3, no. 1, pp. 35–78, 1993.

[17] P. D. Taylor, T. Day, and G. Wild, "Evolution of cooperation in a finite homogeneous graph," *Nature*, vol. 447, no. 7143, pp. 469–472, 2007.

[18] G. Szabó and G. Fáth, "Evolutionary games on graphs," *Physics Reports*, vol. 446, no. 4–6, pp. 97–216, 2007.

[19] C. P. Roca, J. A. Cuesta, and A. Sánchez, "Effect of spatial structure on the evolution of cooperation," *Physical Review E*, vol. 80, no. 4, Article ID 046106, 2009.

[20] C. P. Roca, J. A. Cuesta, and A. Sánchez, "Evolutionary game theory: temporal and spatial effects beyond replicator dynamics," *Physics of Life Reviews*, vol. 6, no. 4, pp. 208–249, 2009.

[21] B. Allen, A. Traulsen, C. E. Tarnita, and M. A. Nowak, "How mutation affects evolutionary games on graphs," *Journal of Theoretical Biology*, vol. 299, pp. 97–105, 2012.