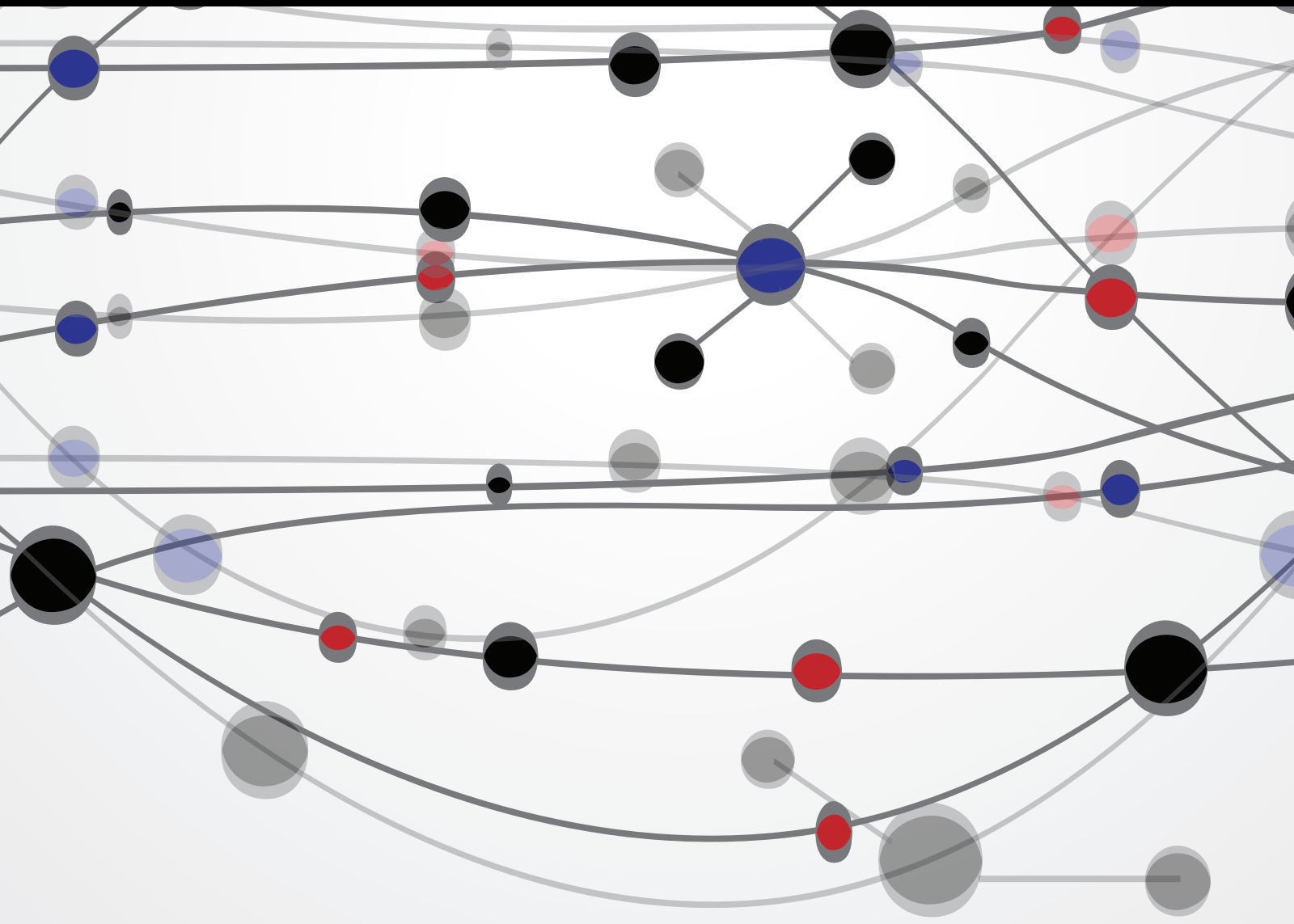
The Scientific World Journal

Recent Advances in Complex Networks Theories with Applications

Guest Editors: Hamid Reza Karimi, Wei Zhang, Xuebo Yang, and Zhandong Yu



Recent Advances in Complex Networks Theories with Applications

The Scientific World Journal

Recent Advances in Complex Networks Theories with Applications

Guest Editors: Hamid Reza Karimi, Wei Zhang, Xuebo Yang, and Zhandong Yu

Contents

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 504923, 4 pages
http://dx.doi.org/10.1155/2014/504923

Editorial

Recent Advances in Complex Networks Theories with Applications

Hamid Reza Karimi,[1] Wei Zhang,[2] Xuebo Yang,[3] and Zhandong Yu[4]

[1] *Department of Engineering, Faculty of Engineering and Science, University of Agder, 4898 Grimstad, Norway*
[2] *Department of Management Science and Engineering, School of Management, Harbin Institute of Technology, Harbin, China*
[3] *Research Institute of Mechatronics and Automation, Bohai University, Jinzhou, Liaoning, China*
[4] *College of Automation, Harbin Engineering University, Harbin, China*

Correspondence should be addressed to Hamid Reza Karimi; hamid.r.karimi@uia.no

Received 4 May 2014; Accepted 4 May 2014; Published 23 July 2014

Since the 1980s, the rapid development of computer and information engineering technology represented by the Internet makes the human society stride into the "network era." From the Internet to the World Wide Web, from the power grid to the transportation network, from the organism's brain neural networks to the metabolic network, and from research cooperation network to a variety of political, economic, and social networks, people have actually lived in a world filled with a wide variety of complex networks. Hence, in recent years, complex networks have become the research focus in different disciplines including mechanical, physics, biology, system control, communication technology, social, economic, and military disciplines. Previous studies have mainly focused on the following: (1) static statistical analysis of complex network topology, such as the matching pattern based on a given degree distribution, a variety of correlation, and the description of weighted networks; (2) the evolution and mechanisms model of complex networks, such as network evolution statistical regularity and geometric properties of network; (3) kinetics of complex networks, including network fault tolerance and attack robustness, and the dynamic processes of spread, synchronization, and resonance on the network.

The overall aim of this special issue is to bring together the latest/innovative knowledge and advances in theoretical and technological developments and their applications to complex networks, which may depend largely on methods from artificial intelligence, statistics, operational research, management science, and engineering.

This special issue contains twenty-eight papers, the contents of which are summarized as follows.

"*Extraction of multilayered social networks from activity data*" by K. Musial et al. proposes a new method for creation of multilayered social network based on the data about users activities towards different types of objects between which the hierarchy exists. Due to the flattening, preprocessing procedure new layers and new relationships in the multilayered social network can be identified and analysed.

"*A social diffusion model with an application on election simulation*" by J.-K. Lou et al. designs a diffusion model which is capable of managing the aforementioned scenario. To demonstrate the usefulness of our model, the authors simulate the diffusion on the network built based on a publicly available bibliography dataset.

"*An energy efficient simultaneous-node repositioning algorithm for mobile sensor networks*" by M. A. Khan et al. addresses the concerns of both connectivity and coverage in an integrated way so that this gap can be filled. A novel algorithm for simultaneous-node repositioning is introduced. In this approach, each neighbor of the failed node, one by one, moves in for a certain amount of time to take the place of the failed node, after which it returns to its original location in the network. The effectiveness of this algorithm has been verified by the simulation results.

"*Joint estimation of 2D-DOA and frequency based on space-time matrix and conformal array*" by L.-T. Wan et al. studies a joint frequency and two-dimension DOA (2D-DOA) estimation algorithm for conformal array. The delay correlation function is used to suppress noise. Both spatial and time samplings are utilized to construct the spatial-time matrix. The frequency and 2D-DOA estimation are accomplished based on parallel factor (PARAFAC) analysis without spectral peak searching and parameter pairing. The proposed algorithm needs only four guiding elements with precise positions to estimate frequency and 2D-DOA. Other instrumental elements can be arranged flexibly on the surface of the carrier. Simulation results demonstrate the effectiveness of the proposed algorithm.

"*Beamforming of joint polarization-space matched filtering for conformal array*" by L. Liu et al. investigates a novel joint polarization-space matched filtering algorithm for cylindrical conformal array. First, the snapshot data model of the conformal polarization sensitive array is analyzed. Second, the analytical expression of polarization sensitive array beamforming is derived. Linearly constrained minimum variance (LCMV) beamforming technique is facilitated for the cylindrical conformal array. Third, the idea of joint polarization-space matched filtering is presented, and the principle of joint polarization-space matched filtering is discussed in detail.

"*The robustness analysis of wireless sensor networks under uncertain interference*" by C. Deng presents robustness analysis of condition monitoring wireless sensor network under uncertain interference based on the complex network theory. In the evolution of the topology of sensor networks, the density weighted algebraic connectivity is taken into account, and the phenomenon of removing and repairing the link and node in the network is discussed. Numerical simulation is conducted to explore algebraic connectivity characteristics and network robustness performance. It is found that nodes density has an effect on algebraic connectivity distribution in the random graph model; high density nodes carry more connections, use more throughputs, and may be more unreliable.

"*Reconfiguration and search of social networks*" by L. Zhang et al. presents a new network model which suits to portray the structure of social networks based on the topological characteristics of the real social networks, and the characteristic parameters of the model were calculated. To find out the relationship between two people in the social network, and using the local information of the social network and the parallel mechanism, a hybrid search strategy based on walker random and a high degree was proposed. Simulation results show that the strategy can significantly reduce the average number of search steps, so as to effectively improve the search speed and efficiency.

"*A new cooperative MIMO scheme based on SM for energy-efficiency improvement in wireless sensor network*" by Y. Peng and J. Choi studies a new cooperative multiple-input multiple-output (CMIMO) scheme based on the spatial modulation (SM) technique named CMIMO-SM for energy-efficiency improvement. They first establish the system model of CMIMO-SM. Based on this model; the transmission approach is introduced graphically. In order to evaluate the performance of the proposed scheme, a detailed analysis in terms of energy consumption per bit of the proposed scheme compared with the conventional CMIMO is presented. Later, under the guide of this new scheme they extend their proposed CMIMO-SM to a multihop clustered WSN for further achieving energy efficiency by finding an optimal hop length.

"*Augmented Lagrange based on modified covariance matching criterion method for DOA estimation in compressed sensing*" by W. Si et al. addresses a novel direction of arrival (DOA) estimation method in compressed sensing (CS), in which DOA estimation is considered as the joint sparse recovery from multiple measurement vectors (MMV). The proposed method is obtained by minimizing the modified-based covariance matching criterion, which is acquired by adding penalties according to the regularization method. This minimization problem is shown to be a semidefinite program (SDP) and transformed into a constrained quadratic programming problem for reducing computational complexity which can be solved by the augmented Lagrange method. The proposed method can significantly improve the performance especially in the scenarios with low signal to noise ratio (SNR), small number of snapshots, and closely spaced correlated sources.

"*Channel selection based on trust and multiarmed bandit in multiuser, multichannel cognitive radio networks*" by F. Zeng and X. Shen presents a channel selection scheme for the multiuser, multichannel cognitive radio networks. This scheme formulates the channel selection as the multiarmed bandit problem, where cognitive radio users are compared to the players and channels to the arms. By simulation negotiation we can achieve the potential reward on each channel after it is selected for transmission; then the channel with the maximum accumulated rewards is formally chosen.

"*Energy-efficient routing control algorithm in large-scale WSN for water environment monitoring with application to Three Gorges Reservoir area*" by Y. Zhong et al. studies a new energy-saving routing algorithm to maximally prolong lifetime of large-scale WSN, using the method of maximum energy-welfare optimization clustering. Firstly, temporary clusters are formed based on two main parameters, the remaining energy of nodes and the distance between a node and the base station. Secondly, the algorithm adjusts cluster heads and optimizes the clustering according to the maximum energy welfare of the cluster by the cluster head shifting mechanism. Finally, in order to save node energy efficiently, cluster heads transmit data to the base station in single-hop and multihop way.

"*Distributed leader-following finite-time consensus control for linear multiagent systems under switching topology*" by X. Xu et al. investigates the finite-time consensus problem of leader-following multiagent systems. The authors propose a sufficient condition to design the observer-based consensus protocol, which makes the multiagent systems achieve finite-time consensus under switching topologies.

"*Content patterns in topic-based overlapping communities*" by S. A. Ríos and R. Muñoz studies the problem of understanding the underlying community structure for social network analysis. The authors present a hybrid algorithm

which combines two different overlapping subcommunity detection approaches and a method to analyze and compare the content generated.

"*In-band asymmetry compensation for accurate time/phase transport over optical transport network*" by S. Siu et al. studies the issues that are relevant to distributing accurate time/phase over optical transport network (OTN). With the proposed scheme in this paper, the fiber link delay asymmetry can be compensated relying on the estimated mean fiber link delay by the Telecom-Boundary clock, while the OTN control plane is responsible for processing the fiber link delay asymmetry to determine the asymmetry compensation in the timing chain.

"*Modeling of information diffusion in Twitter-like social networks under information overload*" by P. Li et al. takes Twitter-like social networks into account and proposes models to characterize the process of information diffusion under information overload. The results are of importance to understand the diffusion dynamics in social networks, and this analysis framework can be extended to consider more realistic situations.

"*Secure and fair cluster head selection protocol for enhancing security in mobile ad hoc networks*" by B. Paramasivan and M. Kaliappan studies the problem of secure and fair cluster head selection for mobile ad hoc networks (MANETs). The authors present a secure and fair cluster head selection protocol (SFCP) which integrates security factors into the clustering approach for achieving attacker identification and classification.

"*SVM-based spectrum mobility prediction scheme in mobile cognitive radio networks*" by Y. Wang et al. studies the problem of spectrum mobility which has not been fully investigated in mobile cognitive radio networks (CRNs). The authors propose a novel support vector machine based spectrum mobility prediction (SVM-SMP) scheme by considering time-varying and space-varying characteristics simultaneously in mobile CRNs.

"*A fast overlapping community detection algorithm with self-correcting ability*" by L. Cui et al. defines a weighted modularity based on the density and cohesion as the new evaluation measurement due to the defects of all kinds of modularity. The authors propose three test conditions for overlapping nodes and present a fast overlapping community detection algorithm with self-correcting ability, which is decomposed into two processes. And they also give a new understanding on membership vector.

"*Motion adaptive vertical handoff in cellular/WLAN heterogeneous wireless network*" by L. Li et al. studies the problem of vertical handoff for guaranteeing quality of service and overall performance of network. The authors propose a vertical handoff algorithm based on Q-learning. Meanwhile, neural fuzzy inference system (NFIS) is embedded to retain a continuous perception of the state space.

"*An improved proportionate normalized least-mean-square algorithm for broadband multipath channel estimation*" by Y. Li and M. Hamamura makes use of the sparsity property of broadband multipath wireless communication channels. The authors mathematically propose an lp-norm-constrained proportionate normalized least-mean-square (LP-PNLMS) sparse channel estimation algorithm which can effectively improve the estimation performance of the PNLMS-based algorithm for sparse channel estimation applications.

"*A novel complex networks clustering algorithm based on the core influence of nodes*" by C. Tong et al. investigates the network clustering problem which is significant for the study of complex networks. The authors propose a clustering algorithm by calculating the core influence of nodes. The clustering accuracy of this algorithm is superior to the classical clustering algorithm (fast Newman algorithm). It clusters faster and plays a positive role in revealing the real cluster structure of complex networks precisely.

"*Robust H_∞ filtering for a class of complex networks with stochastic packet dropouts and time delays*" by J. Zhang et al. studies the problem of robust H_∞ filtering for a class of complex network systems which has stochastic packet dropouts and time delays, combined with disturbance inputs. The authors aim to design a filter such that the estimation error converges to zero exponentially in the mean square, while the disturbance rejection attenuation is constrained to a given level by means of the performance index.

"*Discovering the influences of complex network effects on recovering large scale multiagent systems*" by Y. Xu et al. presents an initial effort to find how a standard network recovery policy, MPLS algorithm, may change the network topology of the multiagent system in terms of network congestion. The authors have established that when the multiagent system is organized as different network topologies according to different complex network attributes, the network shifts in different ways.

"*WDM network and multicasting protocol atrategies*" by P. Kirci and A. H. Zaim studies a new multicasting protocol with optical burst switching (OBS). The authors examine the performance of the protocol with just enough time (JET) and just in time (JIT) reservation protocols. Also, the paper involves most of the recent advances about wavelength division multiplexing (WDM) multicasting in optical networks.

"*Assessment on knowledge network sharing capability of industrial cluster Based on Dempster-Shafer theory of evidence*" by S. Dai and H. Zhang presents a concept model of knowledge sharing network based on theory of evidence. Next, the authors create a set of assessment index systems. The research result shows relatively high knowledge network sharing capacity among the certain industrial cluster firms.

"*A novel joint problem of routing, scheduling, and variable-width channel allocation in WMNs*" by C.-C. Lin et al. studies a novel joint problem of routing, scheduling, and channel allocation for single-radio multichannel wireless mesh networks. As a result, this paper first constructs a linear programming model with more practical concerns and then proposes a simulated annealing approach with a novel encoding mechanism.

"*Modeling and simulation of complex network attributes on coordinating large multiagent system*" by Y. Xu et al. presents simulation testbed CoordSim built to model the coordination of network centric multiagent systems. The authors have theoretically analyzed that the characters of

complex network make a significant difference with both random and intelligent coordination strategies.

"*Towards an optimal energy consumption for unattended mobile sensor networks through autonomous sensor redeployment*" by J. Chen et al. studies an autonomous sensor redeployment algorithm to balance energy consumption and mitigate energy hole for unattended mobile sensor networks.

Of course, the selected topics and papers are not a comprehensive representation of the area of this special issue. Nonetheless, they represent the rich and many-faceted knowledge that we have the pleasure of sharing with the readers.

Acknowledgments

We would like to express appreciation to the authors for their excellent contributions and patience in assisting us. The hard work of all reviewers on these papers is also very greatly acknowledged.

Hamid Reza Karimi
Wei Zhang
Xuebo Yang
Zhandong Yu

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 785305, 14 pages
http://dx.doi.org/10.1155/2014/785305

Research Article

An Energy Efficient Simultaneous-Node Repositioning Algorithm for Mobile Sensor Networks

Muhammad Amir Khan,[1,2] Halabi Hasbullah,[1] Babar Nazir,[3] and Imran Ali Khan[3]

[1] *Department of Computer and Information Sciences, Universiti Teknologi Petronas, 31750 Tronoh, Perak, Malaysia*
[2] *Department of Electrical Engineering, COMSATS Institute of Information Technology, University Road, Tobe Campus, Abbottabad 22060, Pakistan*
[3] *Computer Science Department, COMSATS Institute of Information Technology, University Road, Tobe Campus, Abbottabad 22060, Pakistan*

Correspondence should be addressed to Muhammad Amir Khan; amirkhancomsats@gmail.com

Received 27 December 2013; Accepted 19 February 2014; Published 23 July 2014

Academic Editors: H. R. Karimi, X. Yang, Z. Yu, and W. Zhang

Recently, wireless sensor network (WSN) applications have seen an increase in interest. In search and rescue, battlefield reconnaissance, and some other such applications, so that a survey of the area of interest can be made collectively, a set of mobile nodes is deployed. Keeping the network nodes connected is vital for WSNs to be effective. The provision of connectivity can be made at the time of startup and can be maintained by carefully coordinating the nodes when they move. However, if a node suddenly fails, the network could be partitioned to cause communication problems. Recently, several methods that use the relocation of nodes for connectivity restoration have been proposed. However, these methods have the tendency to not consider the potential coverage loss in some locations. This paper addresses the concerns of both connectivity and coverage in an integrated way so that this gap can be filled. A novel algorithm for simultaneous-node repositioning is introduced. In this approach, each neighbour of the failed node, one by one, moves in for a certain amount of time to take the place of the failed node, after which it returns to its original location in the network. The effectiveness of this algorithm has been verified by the simulation results.

1. Introduction

Wireless sensor networks (WSNs) are one of the most significant technologies today as they have become widely utilized and have come to be a research field that is well-developed. Typically, WSNs are made up of a large set of compactly installed small sensor nodes that are mobile. These nodes are inexpensive and use little power. They are most often effectively applied to monitor various environments, process information and for communicates each other by radio [1–5]. WSNs can lower delay and cost during the time of development. Moreover, they are applicable for use in many types of environment. They are used in situations where deployment of normal wired sensor networks would be impossible. These environments are such as those found in outer space, deep oceans, or battle fields [6, 7]. For the most part, sensor nodes are utilized for monitoring in areas of home, health, and military. Because they act promptly, are self-organised, and have fault tolerance individuality, sensor networks used in military applications, for example, can be extremely applicable to many systems in the armed forces. Such military uses are control, command, surveillance, communication, and targeting. However, when used in the health industry, they are implemented to help disabled patients as well as in monitoring patients. In addition, the management of inventory and monitoring of disaster areas and product quality are examples of other commercial matters that can also use these sensor nodes [8–10].

When these nodes are deployed in unattended environments that are very harsh, the depletion of the onboard energy or some type of physical damage can result in the network being separated into multidisjoint blocks. This can cause it to stop working resulting in the loss of the node. Such areas are commonly known as sensing holes. So that these holes can be filled, extra sensors or sensors that were deployed earlier have to be moved so that these failures can be resolved,

or as a response to an event that necessitates moving a sensor to a certain area [11–14]. The internode connectivity is vital for the effectiveness of an application; moreover, there are nodes that have a role to play in keeping the flow of the data from the sensors, which are in place, and from remote users [15].

When a substitution for the dead node needs to be made, it could use up a lot of time and often may not be possible in dangerous environments such as on the battle field. There have been studies that have proposed node repositioning as an efficient tool in the restoration of partitioned networks [16]. During the time that the network is operational, dynamical node repositioning is needed to increase the network's performance. An example of such a situation would be when several sensors surrounding a base-station cease to work because their batteries have been used up; some extra sensors from different areas of the region being monitored can be identified and moved to the affected area for replacement of the nonfunctioning sensors. This will increase the lifetime of the network [17–19]. Moreover, this type of dynamic relocation is quite advantageous in an application that is used to track a mobile target. Some of the sensors, for example, may be moved to a position that is near the target to improve the accuracy of the sensor's data. Furthermore, for safety, some applications could require keeping the base-station a suitable distance away from targets that are dangerous, for example, an enemy tank. This is accomplished by moving it to a less dangerous location so that its availability is guaranteed [20–22].

It is quite difficult to relocate nodes whilst the network is under its normal operating condition. This movement to a new location is made as a reaction to an environment- or network-based event. This is unlike the initial deployment. Further, because the relocation is based on an occurring event, continual monitoring of the state and performance of the network is needed. It also requires an analysis of events taking place in the area of the node [23–25]. Furthermore, this process of relocating the node must be handled with care because of the potential disruption to the delivery of the data. These earlier studies were more focused on restoring broken connectivity with no consideration taken in regard to the unconstructive impact that relocating the nodes could have on the coverage, such as an extra node being relocated in place of the failed node.

Suggesting that the QoS of the sensor network can be measured by coverage with good connectivity [26, 27], therefore, it would appear that connectivity and coverage must both be taken into consideration. It has been shown in an earlier study that failure of an individual node with no redundant node available in the network is similar to baseline methods. This paper proposes an energy efficient simultaneous node repositioning algorithm; moreover, it contributes to filling this hole in the research. Not like other methods that reposition nodes in order to readjust the topology of the network, the algorithm that has been proposed makes an attempt to maintain the network topology as it is and localize the recovery's scope. In general, a node's failure is handled by the replacement of the node, temporarily, with one of its neighbours. These neighbours move to the failed node's location one after another. When a failed node is detected, its neighbours work together to create a schedule so that each of the neighbours knows when to move itself to the location of the failed node. A substitute node will return to its starting position after serving for a certain amount of time. This lets other substitutes for the failed node take its place. This is repeated again and again. As this proposed algorithm is distributed, it creates quite a limited amount of messaging overhead during the process of the recovery.

The main contributions of this paper can be summarized as follows.

(i) Firstly, a method is proposed to dynamically reposition the nodes when simultaneous nodes failure occurred in the network to improve the performance of the network.

(ii) Secondly, an energy efficient simultaneous nodes relocation algorithm for mobile sensor network is introduced to address the simultaneous nodes failure problem. The proposed algorithm strives to keep most of the network topology intact and localize the scope of recovery. The failure of a node is tolerated by temporarily replacing it with one of its neighbours. These neighbours take turns in moving to the position of the failed node.

(iii) Thirdly and lastly, extensive simulations are carried out to evaluate the performance of the proposed protocol, by comparing its performance with baseline approaches. The results demonstrated that our algorithm has successfully minimized the total distance travelled and has improved the other QoS parameters like number of exchanged messages, average number of nodes moved, and percentage of reduction in field coverage.

The following sections of this paper are organised as follows. In Section 2, related work is summarized. In Section 3, simultaneous node relocation algorithm for mobile sensor networks is proposed. Simulation results are presented in Section 4, and conclusions and future work are offered in Section 5.

2. Related Work

Related to this category are three algorithms: the VECtor-based algorithm (VEC), VORonoi-based algorithm (VOR), and Minimax algorithm. They were suggestions from Wang et al. [28]. There is a close relationship to each of these three algorithms with the Voronoi polygon of the sensor node or point. This is where a node or point is located closer than another node or point to the sensory boundary. In a situation such as this, the VECtor-based algorithm is prepared following Coulomb's law as a process of equating that displays the prevention amongst the electrostatic particles. Consequently, the node that is used for the dispossession of an element of a node's Voronoi polygon is moved away from the other nodes that are close with a force that is comparative to the distance it is from the angular points of the polygon or from the nodes themselves. Heo and Varshney [29] proposed a corresponding procedure as an indication of Coulomb's

law but in this case no consideration is made to the node's Voronoi polygon. Each node functions, in turn, as a substitute that is going to move far away, proportionally, to the nodes' compactness in its locale usually nearby. It is able to cause the nodes to relocate to the closest neighbouring point of its Voronoi polygon, at the same time, putting forth a polygon that is more regular. In fact, the VORonoi-based algorithm, the nodes in a number of locations to fluctuate. Meanwhile, in the Minimax algorithm, a rather small and common movement can be found. This is true even when there is not as much fluctuation. However, all in all, these algorithms could, in fact, bring forth movements, intermittently, which afterward cause wastage of both time and energy.

In order to reduce how long the course is, a method that is based on a proxy without any sensor node movement. The exception would be in the case that their destination is calculated beforehand [30]. In their work, they focused more on a system that corroborates with both immobile and mobile sensors. In this situation, the mobile nodes are responsible for loading the extent of the positions from which the nodes are absent. This is accomplished in manner that is distributive and can be predicted by immobile nodes. This implies that the movement of the mobile nodes is logical and that the immobile nodes as the agents are in a logical position. Using this method will result in the distance being reduced by quite a bit. The reduction could be from its mean or from the totality in which the mobile nodes are directed along in order to maintain an equal rate of coverage [31]. On the other hand, the result might only be an increase in the message density. In general, the assumption can be made here that methods such as these have a tendency to avoid holes in the coverage rather than actually concentrating on the connectivity. Another solution was proposed by Wu and Yang. Their proposal known as SMART made use of 2D scrutinisation of the selected networks. The aim was to cut down on the overall deployment time [32]. A method adopted from a well-accepted design for a balance of the load amongst nodes in a number of conformations on the parallel processing was used. The process is divided into separate components, implemented on various processors. Subsequently, this design is utilized in multicluster WSNs. In these networks, each single cluster is constituted by a 2D mesh that is formed by square cells. Moreover, several sensors will be added to a single cell to represent the load on a cluster. The location in an interconnection in vertical or horizontal indices as well as the number of sensors in its cluster indicates that a cluster-head's only communication can be with sensors in similar locations in other cells close at hand. In due course, the corresponding coverage that is achieved is related to the issue of balancing the amount of the energy needed to level the distribution of the sensors amongst the clusters. All in all, these methods are more focused on avoiding gaps in the coverage than they are in sustaining connectivity.

Situating nodes into a structure that is very efficient is the aim of the algorithm mentioned above. When the network application begins, a node failing could cause the efficiency to be reduced quite a bit. Moreover, alterations in the needs of the application could also have an influence on the meaning of efficiency itself. In both situations, the nodes have to reposition themselves to maintain a network layout's efficiency. Wang et al. [28] proposed the cascaded movement to replace failed sensor nodes. It achieves this using nearby redundant nodes to repeatedly take the place of the failed node. Some other studies have also taken connectivity into consideration. This is explained in [28] where, for example, one method makes the decision to maintain connectivity of two degrees. It does this, even if a link or node fails, by relocating a subset of the nodes. Whilst the movement of the nodes is comparable to our approach, if the need for 2-connectivity is stressed upon, the application-level functionality might be constrained. However, in large-scale networks, where the nodes' resources are constrained, this may not be practical. In our study, the approach with the closest relation to RIM (recovery through inward motion) found in the literature is DARA (distributed actor recovery algorithm) [33]. In the approach, each of the nodes is required to keep a list of their 2-hop neighbours. They must choose one of the failed node's neighbours to move on the basis of how many communication links there are. However, a set of rules for a scattered actor improvement for fixing the actor networks that have been partitioned was suggested by Akkaya et al. [34]. Real-time restoration is raised by DARA with no hypothesis being made as to how the network is interconnected before an actor malfunctions. Moreover, there is no interrelation to each other. Instead of relocating a block [35], DARA may track a flown reposition of several actors. In this way, a small number of movements are assumed rather than a whole block movement. This leads to the entire movement of the actors being required in the movement or flown reposition. Moreover, the block movement involves being aware of all of the actors in each separated network moving towards the location and how far the movement goes. This introduces extra messaging overhead. In this situation, it is performed so that the two-hop neighbour lists can be created and easily maintained.

Meanwhile, postdeployment connectivity and coverage were considered by Akkaya and Younis [36, 37]. Coverage is increased without breaking any existing internode links when C^2AP (coverage-aware connectivity-constrained actor) spreads out the connected nodes whilst with COCOLA (connected coverage and latency aware actor placement), a network architecture that is hierarchical is considered. In this approach, nodes on a higher tier are repositioned, incrementally, so that the coverage can be maximised but with no extension to the data route to the node on the first tier. This is so that a desired bound on data latency is maintained. However, the impact that a failed node has is not handled by either C^2AP or COCOLA.

The idea of assigning a mobility readiness index (MRI) to each actor according to the presently performed task's impact was proposed by Abbasi et al. [38]. It is at this time the value of the MRI that allows an actor to be relocated or not. The network topology is such that if noncritical actors fail, the interactor connectivity providing alternate routes will not be damaged. However, three disadvantages have been noted with their proposed method. The first disadvantage is that C2AM (interactor connectivity with application level

constraints on actors' mobility) is a reactive method and, therefore, it might not be suitable for applications that are mission-critical and time-sensitive. The second disadvantage is that C2AM does not take into consideration the actor's ability when relocating the actor during the process of recovering a malfunction. Here, performing the relocation of a noneffective actor might cause the opposite of the effect wanted in the application. The third disadvantage is that C2AM needs 2-hop information to be maintained and does not care about the coverage of the actor.

The NN (nearest neighbour) and RIM (recovery through inward motion) algorithms were in turn proposed by Younis et al. [39]. RIM is a distributed approach for the restoration of connectivity by way of an inward motion. The main concept behind the approach is that upon the failure of node F, its nearest neighbouring nodes will reposition themselves inwards to the location of the failure so that they will have the ability to link with each other. This is a result of the neighbouring nodes referring to the nodes that have been directly impacted on by the failure of node F. Thus, when they can reach each other again, the connectivity of the network is restored to its level before the failure took place. The relocation process is performed in a repeated manner for any node that has failed in order to move one of its neighbour nodes, for example, one of the nodes that has already relocated to the failed node's position. The NN approach, like RIM, follows voracious heuristics. Upon the failure of a node, the NN will travel to its nearest neighbour, FNN, where F is located. It takes this action in order to repair the disrupted connectivity around node F. In response to the relocation of FNN, its closest neighbour from among those nodes near its location will travel to the original position of FNN and settle there. This process will be repeated. When no neighbour can be found for a relocated node as far as the edge of the network or all of the network nodes have already been repositioned, the NN will stop. Unlike RIM which uses a 1-hop neighbour list, NN requires each node to have knowledge of its 2-hop neighbours. Because of this, the closest neighbouring node will be determined ahead of the F node failing. This is where, neither RIM nor NN are concerned with the impact that the restoration of the connectivity has on the coverage of the network. Repositioning permanently like this is avoided by using the algorithm known as coverage conscious connectivity restoration (C^3R) [22]. Whilst exchanging another node for the neighbour node restores the connectivity, the fact of the matter is that it only changes the gap in coverage to another area of the field. This could be in the inner area of the network or at its outer edge. This could be handled with the temporary replacement of the node that has failed with one or more of its neighbouring nodes. Table 1 shows comparative summary of discussed state of the art protocols, in terms of some basic parameter.

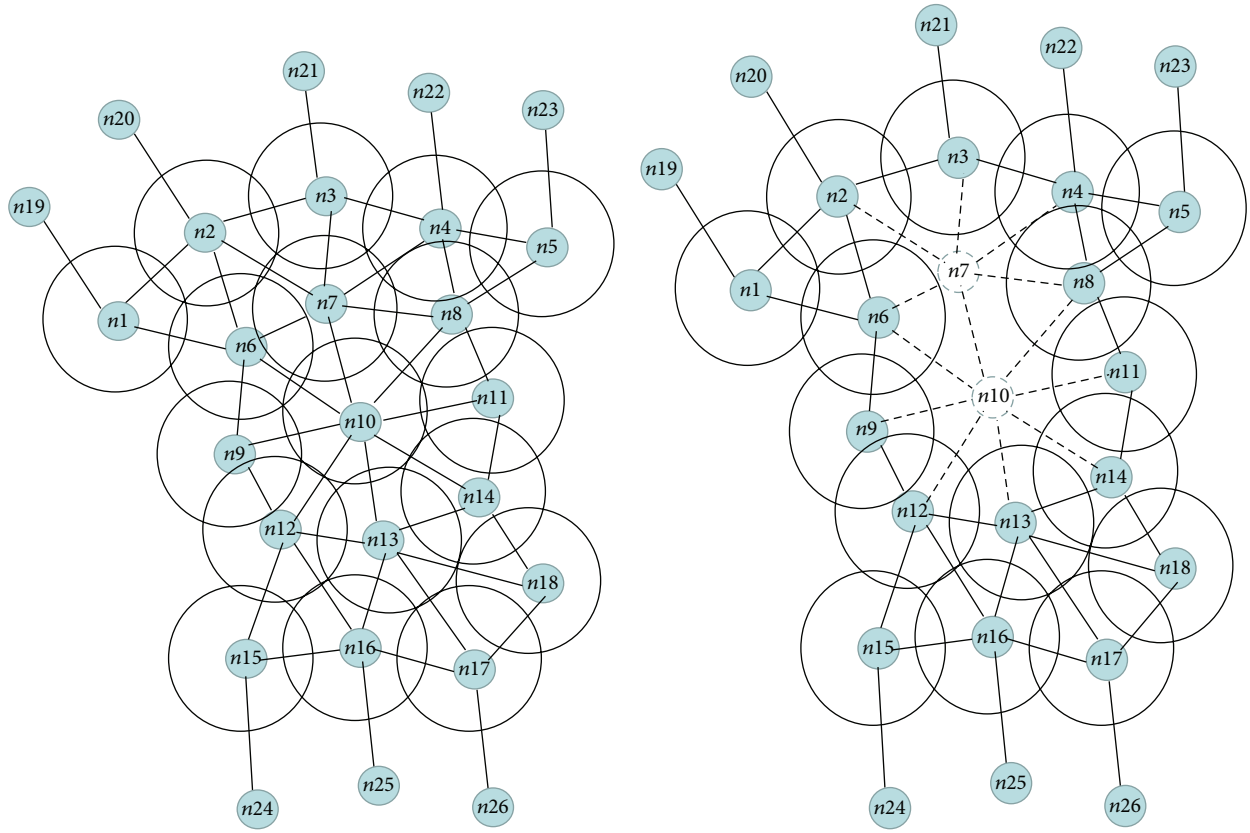
3. Proposed Protocol

3.1. Problem Description. Not only could the coverage of the network be affected by losing a node as a result of node failure but also it could have an effect on the connectivity of the network. The proposed method's procedure for restoration of connectivity starts off in the same manner as C^3R. The focus of this work is on keeping the network connected while maintaining the prefailure coverage level when there is a failure of simultaneous nodes. The proposed method can be referred to in Figure 1. Figure 1(a) shows the topology of a network that has well connected nodes. As shown in Figure 1(b) nodes 7 and 10 failed simultaneously and all the neighbouring nodes relocate towards nodes $n7$ and $n10$ in order to start recovery process. It is possible that relocating other nodes to replace nodes that have failed can restore connectivity. However, it does not solve the problem as it only moves the gap in the coverage area to another position of the field. This could be in the inner area or at the outer edge of the network. If the failed node is replaced temporarily by one or more of its neighbours, this issue could be dealt with. In other words, when there is a failure of the participating nodes' neighbour, each participating node will make a decision related to some particular criteria as to which neighbour node it will join. The nodes involved will take turns moving back and forth which will result in the topology of the network as well as its coverage being nearly the same as they were before the failure. An energy efficient simultaneous node repositioning method for network connectivity and coverage recovery has been proposed in this paper.

3.2. Proposed Solution. As discussed earlier, the network could be partitioned into disjoint segments and/or there could be a gap in the network coverage if node F failed. If the node, A, is repositioned to the failed node's location to facilitate connectivity restoration and node A is not an extra node, then the gap in the coverage is simply moved to another location. As a result, connectivity is still lost because of the relocation of node A. However, if node A could return to its home position after spending only a certain amount of time at node F's spot, a major change in the topology of the network could be avoided. This to and from movement of A restores the network connectivity and coverage from being lost permanently but does not result in a cascading motion. Other close neighbours of node F could do likewise. In this way, the process of recovering connectivity could then be rotated amongst the other nearest neighbours of the failed node. This would continue until there is no further introduction of any new, permanent loss of connectivity and/or coverage. Furthermore, the process used for the recovery of the network should be not only fast but also lightweight. As stated previously, the failure of a node which causes a network to be partitioned is the most challenging problem and it is very serious. In such a situation, when trying to restore connectivity, the main problem is that some nodes could be unreachable by others. As such, it becomes difficult to achieve a well-coordinated noncentralised process of recovery. Further, the sensor nodes, which are resource-constrained, require that the overhead be minimised. A solution would be for all of node F's neighbours to simultaneously begin moving towards the site of node F. Eventually, the nodes would arrive at a position where they would be in range to communicate with each other. At that

Table 1: Protocol summary.

Reference	Optimization objective	Migration technique	Node type	Area partition	Network architecture	Sensor mobility	hops	Limitations/constraint
[28]	Connectivity restoration	Direct	Sensor	Voronoi diagram	Centralized	Robot	2	Convergence time: do not consider coverage
[29]	Connectivity restoration	Shifted	Data collector	Zone	Centralized	Robot	2	Convergence time: do not consider coverage
[31]	Coverage	Shifted	Sensor	Plane	Centralized	Actuator	2	Do not consider coverage
[32]	Connectivity	Shifted	Sensor	Square or hexagon	Centralized	Mobile sensor	2	Do not consider connectivity
[33]	Connectivity	Shifted	Data collector	Mesh	Distributed	Robot	1	Do not handle actor failure
[34]	Connectivity	Shifted	Data collector	Plane	Centralized	Actuator	2	Do not consider coverage
[35]	Connectivity	Shifted	Sensor	Plane	Centralized	Mobile sensor	2	Do not consider coverage
[36]	Coverage	Direct	Data collector	Plane	Distributed	Actuator	1	Consider neither coverage nor application level interests
[37]	Coverage	Shifted	Data collector	Plane	Distributed	Actuator	1	Do not handle actor failure
[38]	Connectivity	Shifted	Sensor	Grid	Distributed	Mobile sensor	2	Do not consider connectivity
[39]	Connectivity	Direct	Sensor	Grid	Distributed	Mobile sensor	2	Do not consider coverage
[22]	Connectivity and coverage	Direct	Sensors	Grid	Distributed	Mobile sensor	1	Do not consider multinode and simultaneous node failure



(a) Connected mobile nodes network

(b) After simultaneous nodes failure

Figure 1: A connected network of mobile node. Individual node denoted the communication range.

time, they would synchronise their clocks and decide on a schedule for the recovery plan. A time slot could be assigned to each neighbour as to when each would move to the failed node's area during the recovery process. The schedule would be agreed upon and each node would return to its home location to begin to follow the new schedule and initiate the actual process of recovery.

3.3. Explanation of Proposed Protocol. It is obvious that when a network has extra nodes, replacing the failed node F with one of the extra nodes is the best, most effective way to solve the failure problem in terms of the coverage and connectivity of a network. On the other hand, without extra nodes it is impossible to keep degradation from taking place. The philosophy behind the design of the proposed method, as indicated above, is to keep from having to permanently replace a node that has failed. To achieve this end, the role that used to be played by F for sensing and for routing data in the network is carried out by its nearest neighbours instead. However, the neighbours involved in the recovery must still take care of their own tasks. As a result, the jobs that are performed for failure tolerance are functions which add load to these neighbouring nodes.

3.3.1. Process before Node Failure. In our approach, a prefailure list of 1-hop neighbours is the only knowledge required of each node. This list is created before deployment and each node broadcasts a HELLO message as an introduction to its neighbours. Moreover, the nodes must each determine the ID and position of each of its neighbouring nodes. With the proposed approach, only approximate GPS-based coordinates are needed for the nodes' locations. This data is needed for use only in the case of a failing neighbouring node. Moreover, in order to verify that they are available, nodes will transmit heartbeat messages to their neighbours from time to time. Thus, if a node, A, has not received its predetermined count of heartbeat messages from a neighbouring node, F, it will take it to mean that the node, F, has failed. Upon beginning to relocate, a node will transmit a message of its move to each of its neighbours. In this way, it will not be mistakenly believed to have failed. Moreover, this means that the list of the neighbouring nodes could be updated every time the location of one of them is altered.

Node A initiates the process of the recovery as soon as it detects that a neighbour node, F, has failed. It is noteworthy that there are two general strategies available. The first strategy involves determining if the failure of node F will cause the network to be partitioned, and a response will be made only if F is a cut-vertex. This solution, as adopted by DARA, prevents an overreaction from being caused when a failure not preventing the other nodes from communicating

with each other takes place. Although, this option requires that there be a procedure for nontrivial cut-vertex detection. This necessitates that 2-hop information be available which causes increased messaging overhead. Furthermore, there is no consideration of the effect that it will have on coverage. The second strategy only implements a process for the recovery when coverage is lost irregardless of node F being a cut-vertex or not. This proposed approach uses this strategy as it does not cause major alterations in the topology of the network when restoring connectivity. Moreover, it deals with the gap caused in the coverage of the network.

3.3.2. Neighbours' Node Management. Node A starts the process of recovery as soon as it detects that its neighbour node F has failed. The first thing that node A must determine is whether or not F has neighbouring nodes which could take part in the recovery process. From here on, node F's neighbours will be known as concerned nodes. Node A will not be aware of all of the concerned nodes since the individual nodes keep a list of only their 1-hop neighbours. These other concerned nodes could also be making plans at the same time for their part in recovering from node F failing. The only way that all of the concerned nodes can coordinate with each other is if each of them were to travel towards the location of node F until they could be positive that they were in the communication range of all of the others. It must be understood that the failure of node F could cause all of the routes of communication among these concerned nodes to be disrupted. However, a node needs only to travel for a distance of rc/2 from node F, where a node's communication range is rc, to be able to connect with each and every one of the concerned nodes. As the concerned nodes might not detect and react simultaneously to node F's failure, some synchronisation is needed for this scenario. In lieu of that, the concerned nodes could all travel to the position of the failed node. Then, the first to get there, say node J, would communicate with the other nodes. At that time, they would synchronise with each other. This would negate the requirement for the synchronisation of all of the nodes with each other to be distributed. However, it could cause the travelling distance to be increased which would lead to an increase in the overhead. Even so, our approach favours this method because of the high degree of coordination involved in the proposed approach. Another reason for this choice is that it is difficult, in practice, to come up with a waiting time that is suitable in order to ensure a meeting time for all of the concerned nodes. From now on, the recovery coordinator or the coordinator of the recovery will refer to node J.

3.3.3. Input Parameter Calculation Values. The recovery coordinator performs the role of synchronising the participating nodes, developing a plan for the recovery and distributing the plan to the nodes involved. Later it will be seen that the nodes that are involved in the recovery process are selected based on the coverage overlap of the nodes, how close they are to the failed node, and how much residual energy they possess. The nodes must each calculate these parameter values as they apply to themselves. Then they share the values with the node coordinating the recovery. The coverage overlap of a particular node is the ratio of the entire area that the node covers and which is situated inside the sensing range of one or several of its neighbours. Determining the overlapped coverage is made simple because the nodes know their location in relation to their neighbours. This is accomplished by basing it on the sensing range, r_s, of a node. When a disk coverage model is assumed, the area of the overlap can be calculated by taking into consideration the relationship between the node's closeness to its neighbours and the r_s. When the neighbour is closer, there is a larger intersection between the two circles of the radii r_s around the nodes. In addition, the overlapped coverage of a node will be greater when there are more neighbours within the distance r_s. Two neighbouring nodes have overlapped coverage if the distance, d, between them satisfies

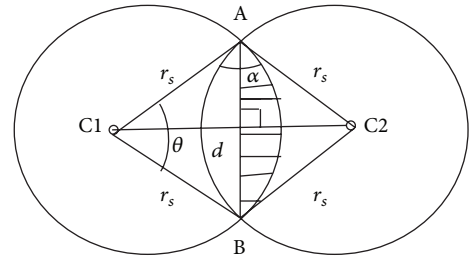


Figure 2: Illustration of the calculation of the overlapped coverage of nodes C1 and C2.

$$d \leq 2 \times r_s. \tag{1}$$

Figure 2 shows that the coverage of the overlap from two nodes is able to be estimated if the area of the chord (θ), hereafter referred to as area (θ), is found. The distance, d, between two nodes, if they are neighbours, can be calculated. When the law of cosines in fundamental Euclidean geometry is used, a node can compute

$$\angle\alpha = 2\sin^{-1}\frac{d}{2r_s}. \tag{2}$$

Then, a triangle with the value of y can be calculated by making use of the property of the sum of angles. In Figure 2, the area (θ) was the difference between the area of sector AB and triangle ACB. It was calculated as presented as follows:

$$\text{area}\,(\theta) = \frac{\theta}{\pi}r_s^2 - \frac{d}{2} \times r_s \sin\left(\frac{\theta}{2}\right). \tag{3}$$

The formula for the area of a chord, presented below, was utilized to find the overlapping ratio:

$$\text{Overlap} = \frac{2 \times \text{area}\,(\theta)}{\pi r_s^2}. \tag{4}$$

3.3.4. Recovery Plan Implementation. The basic main factors for designing and implementing the plan for the recovery process are given as follows.

(i) Participating node A is able to calculate its coverage overlap, distance to F, and energy reserve before it begins to relocate to location F.

(ii) Node A verifies with its neighbouring nodes for temporary relocation in avoiding to declare faulty that neighbour find other route or buffer the data until node A returns to its original position.

(iii) If two nodes claim to be the recovery coordinator, the node closest to F with the lowest ID will be given the task. That node will then broadcast the message to all of the participating nodes.

(iv) This recovery coordinator keeps the list of the ranking which is determined by taking into consideration the coverage overlap, distance travelled, and the amount of energy in reserve. It then uses the round robin manner to set the priority of the relocation.

(v) During its return trip, A informs its neighbouring nodes and begins the transmission of the buffered data packets again. The same node will repeat a like preset process after that.

(vi) Once it gets below the threshold, the node will transmit a request. The node presently situated in the location of F will receive the request. This node will take the place as the new recovery coordinator and produce a new schedule.

The main factors for designing and implementing the plan for the recovery of connectivity in the case of simultaneous nodes failing at various times are as described below.

(i) As shown in the Figure 3(a) node 7 and node 9 simultaneously failed and in order to start recovery process the neighbour of the failed nodes will start the recovery process they move towards the failed nodes.

(ii) Node 6 and node 8 both are in the range of failed nodes n7 and n10. Node 6 and node 8 first calculate their overlap coverage and distance with the failed nodes, as shown in the Figure 3(b). Due to high overlapping with the failed node n7, the node n6 and n8 will move towards the n7 to participate in the recovery process.

(iii) If the overlapped distance is equal then neighbour node will relocate to that neighbour failed node having less node ID.

(iv) According to Figure 3(c) Node 6 and node 11 is the first node in the recovery schedule they will relocate to the failed nodes.

(v) Figure 3(d) shows that after relocation node 6 and node 11 are back to their initial position. Node 2 and node 14 will relocate to the failed nodes.

(vi) In Figures 3(e)–3(g) after relocating back to their initial positions according to recovery schedule the rest of the nodes will relocate in round robin fashion.

Algorithm 1, explains the recovery process when simultaneous node failure occurs.

Table 2: Parameters for the communication energy model.

Term	Description
β_{11}, β_{12}	Energy dissipated in transmitter and receiver electronics per bit (take to be 25 nj/bit).
β_2	Energy dissipated in transmitter amplifier (take to be 50 pJ/bit/m^2).
s	Number of bits in the message.
d	Distance that the message traverses.

4. Simulations and Results

The simulations were carried out based on the developed system model using OMNet++ [40] for evaluation of how the proposed approach performed. A comparison is made against two contemporary protocols: RIM and NN [39]. The details of the setup of the simulation, the energy model, and the discussion of the results are presented in this section.

4.1. Simulation Setup. The experimentation involved in the simulation process consisted of WSN topologies that were randomly produced with nodes in different numbers as well as different ranges of communication. The number of nodes was set to 25, 50, 75, 100, and 125 in a field of $1000 \times 1000\,\text{m}^2$. RIM and NN make no accommodation for varying communication and sensing ranges so the values of r_s and r_c were maintained as equal throughout all of the experimentation. On the other hand, the experimentation for the proposed approach was carried out by using different sensing and communication ranges as well as by measuring the alterations in the coverage field. These ranges were set to 25, 50, 75, 100, 125, and 150 m. Each node started with an energy level of 100 J. The energy used for communicating, sensing, and moving was calculated on the basis of the specified model.

4.2. Sensor's Energy Model. For the simulation, the models for energy consumption for different activities of a sensor node are summarised below.

Communication Energy Dissipation. In this model, the main energy parameters for communication were the energy/bit used by the transmitter electronics (β_{11}), energy used up in the transmit op-amp (β_2), and energy/bit used by the receiver electronics (β_{12}). If a 1/dn path loss was assumed, the consumed energy was

$$E_{tx} = \langle \beta_{11} + \beta_2 d^n \rangle \times s,$$
$$E_{rx} = \beta_{12} \times s, \tag{5}$$

where E_{tx} was the energy used to send s bits and E_{rx} was the energy used to receive s bits. Table 2 specifies the energy parameters.

```
Input: Identify simultaneous node failure
Output: Simultaneous node failure recovery
Begin
(1) if (nodes, detects a simultaneous neighbour node, F1,F2, has failed)
(2) update routing table
(3) Check level of onboard energy supply
(4) if (sufficient energy)
(5)       calculate the degree of overlapped coverage of both the failed nodes
(6) if (failed F1 node distance is short)
(7)       send "Temporary Relocation" message to neighbours
(8)       send relocation message to current recovery coordinator
(9)       RelocateTemporarilyTo(F1 failed node)
(10)      end if
(11)   else if (distance is equal then relocate to less node id)
(12)  else if (Continue the running recovery process)
(13)  else if (node, A, receives "Temporary Relocation")
(14)  Find new route
(15)   if (new route not found)
(16)       Buffer data
(17)   end if
(18)       else if (node, A, receives "Relocated Back")
(19)                if (buffering data)
(20)                transmit data through A
(21)                end if
(22)       end if
(23)    end if
(24) end if
END
```

Algorithm 1: Recovery process when simultaneous node failure occurs. During recovery process simultaneous-node failure occurs.

Sensing Energy Dissipation. The energy required to sense one bit was assumed to be a constant (β_3) where the total energy used up to sense s bits was

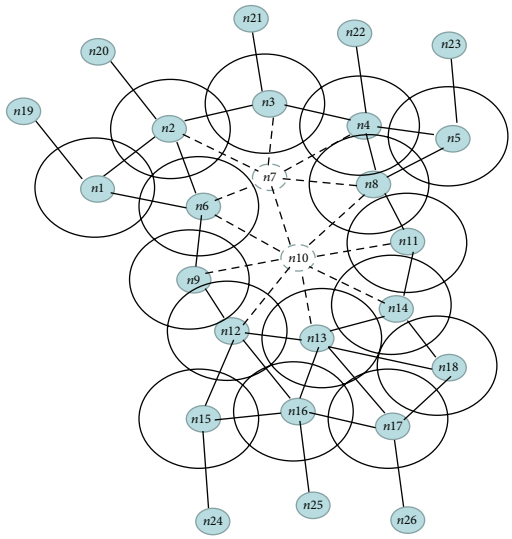
$$E_{\text{sensing}} = \beta_3 \times s. \tag{6}$$

In the simulation, β_3 was equal to 25 nJ/bits. Motion was related to the energy: We made the assumption that a light weight mobile sensor of 0.65 lb could travel at a constant speed of 2.5 cm/s.

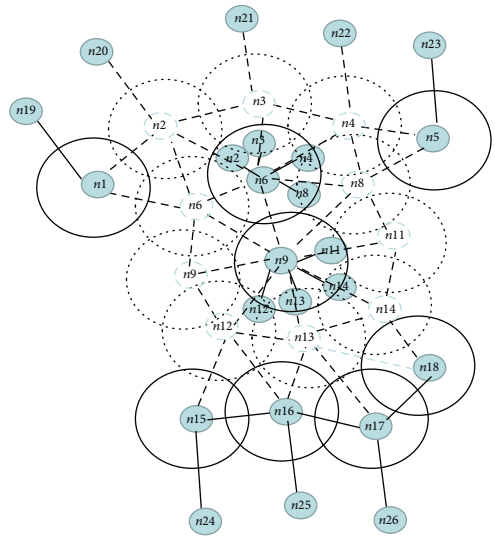
4.3. Results and Discussion. A comparison has been made of the performance of the RIM and NN protocols and the proposed algorithm. For the experimentation, the distance travelled, number of exchanged messages, number of nodes moved, and percentage of the field coverage reduction were some of the parameters employed for measuring how well the proposed algorithm performed.

4.3.1. Distance Moved. The total distance moved is measure of total distance moved by all nodes involved in the recovery which gauges the efficiency in terms of energy efficiency and overhead involved in the recovery. In Figure 4 the total distance moved is plotted on y-axis, with varying communication range (from 25 to 125 m) on x-axis. Figure 4 shows the total distance that nodes collectively had to travel during the recovery as a function of the communication range. Again, the sensing and communication ranges are equal in these sets of experiments. The distance a node would travel depends on the internode proximity, which is at most the communication range r_c. This was easy in the cases of RIM and NN as the distance travelled grew at a high rate. Unlike RIM and NN, with our proposed algorithm, the node involvement was limited in the process of the recovery to only the neighbours of the node that failed. It did not make use of the cascaded relocation found in RIM and NN. It is worthy to mention that when rc was large, the proposed algorithm dealt with the rise in the connectivity of the network very well.

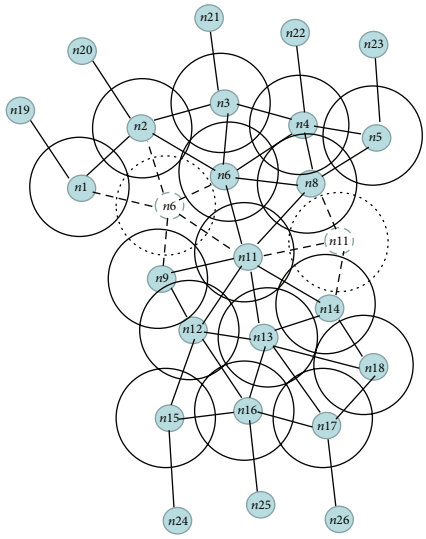
4.3.2. Number of Exchanged Messages. Number of exchanged messages is measured by the number of messages exchanged among nodes. This is a measure of the recovery process overhead. In Figure 5, total packet exchanged is plotted on y-axis, with varying communication range (from 25 to 125 m) on x-axis. Figure 5 shows the total number of packets that were exchanged while restoring connectivity under all three methods. Each broadcast is counted as one message. The messaging overhead with proposed algorithm is minimal, while NN exchanges the most number of packets. This was because, in the proposed algorithm, only the failed node's neighbours were involved. On the other hand, with RIM and NN, the exchange of messages had to synchronise their action with all of the nodes that were repositioned. It is worth mentioning that the number of messages remained almost the same, in our proposed approach, even though



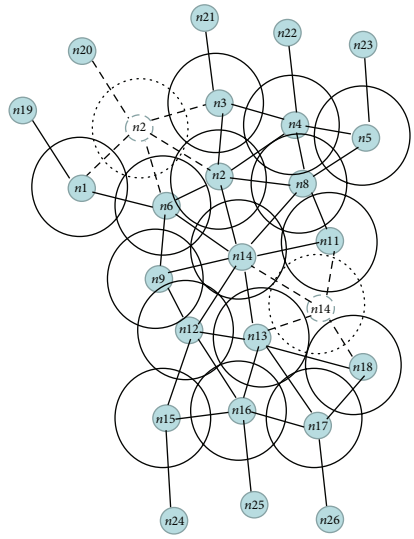
(a) Simultaneous node failure occurred



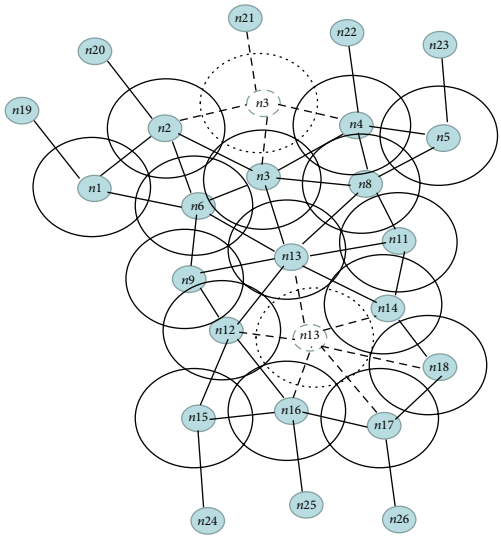
(b) Recovery process after the failure



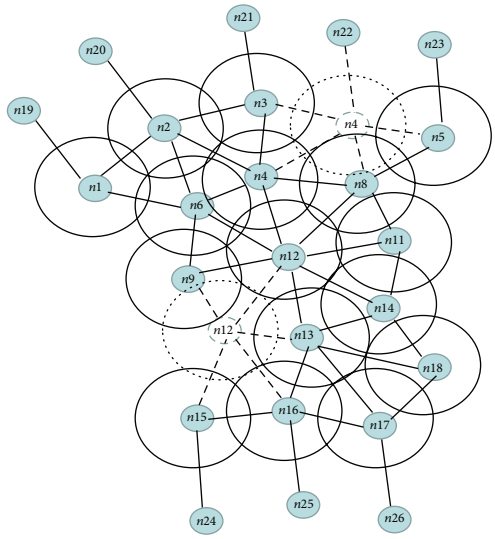
(c) Node 6 and node 11 involved in recovery process



(d) Node 2 and node 14 in during recovery

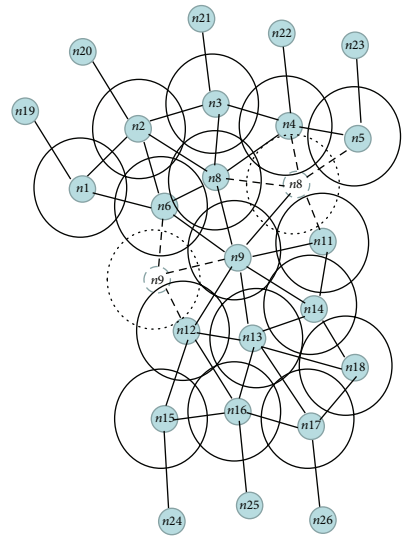


(e) Node 3 and node 13 in during recovery



(f) Recovery process of node $n4$ and node $n12$

Figure 3: Continued.



(g) Node 9 and node 9 involved in recovery process

Figure 3: Example that illustrates the operation of proposed technique.

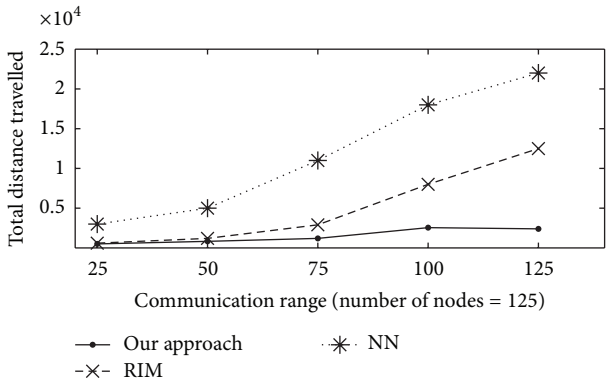


Figure 4: Total distance travelled by all nodes (meters) versus communication range (meters).

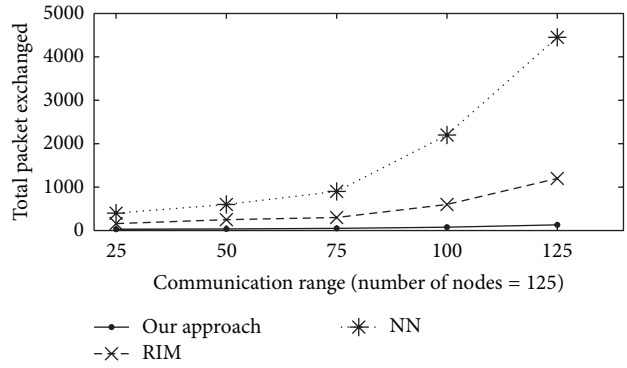


Figure 5: Total number of packets exchanged versus communication range (meters) for 25 to 125 nodes in the network.

there was an increase in the connectivity of the network for a large r_c. This was because interaction rarely took place between the other concerned nodes and the coordinator. Again, Figure 5 presents results based on one round only although it would increase with time. On the other hand, the proposed algorithm can be scaled for several rounds. It would also provide connectivity and coverage restoration at a reasonable cost. This is quite a substantial advantage in performance over solutions that are permanent topology adjustment-based. All in all, our algorithm has been proven to impose only a little bit of messaging overhead as is seen in Figure 5. Moreover, it is suitable to be used with sensor nodes that are bandwidth constrained.

4.3.3. Number of Moved Nodes. During the failure recovery the total number of moved nodes in all three baseline algorithm shown in Figure 6. Similar performance of proposed technique and RIM encountered as in manipulating of the total distance travelled by all nodes. For the reason that more the travelling of number of nodes, the total distance travelled will be higher. At this result the node movement of RIM is higher than NN. In any situation the less number of nodes movements is involved in proposed technique, because it limits the opportunity of recovery to only neighbouring nodes of the failed node.

4.3.4. Percentage of Reduction in Field Coverage. Percentage of reduction in the field coverage is referred to as connectivity centric restoration impacts coverage, measured in terms of the percentage of reduction in field coverage relative to a pre-failure level. In Figures 7–9, average percentage of reduction in the field coverage is plotted on y-axis, with varying number of sensor nodes (from 25 to 125) on x-axis. Figures 7 and 8 show how overall proposed algorithm could significantly limit the loss in coverage. For sparse networks where nodes are evenly distributed with minimal coverage overlap, the field coverage under proposed algorithm decreases by a

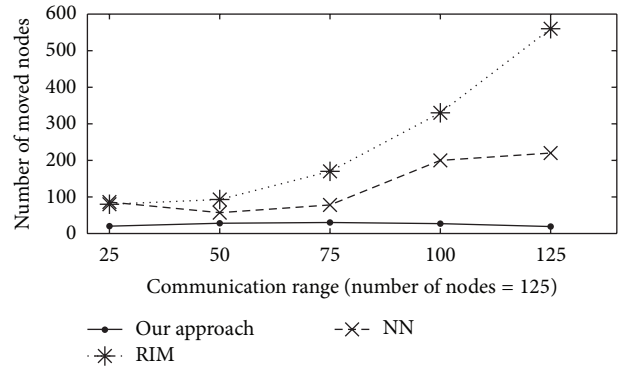


Figure 6: Total number of nodes moved versus communication range (meters) for 25 to 125 nodes in the network.

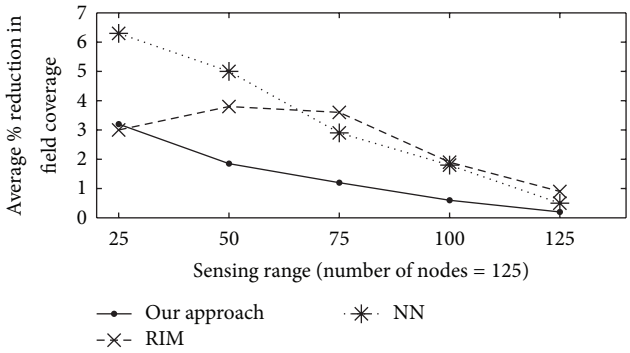


Figure 7: Reduction in field coverage versus sensing range (m) for 25 to 125 nodes.

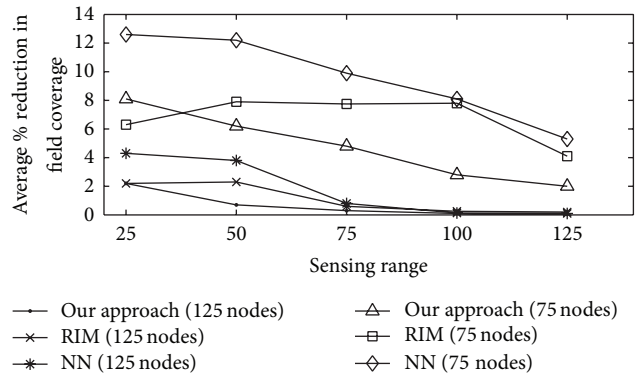


Figure 8: Reduction in field coverage versus sensing range (m) for 75 and 125 nodes.

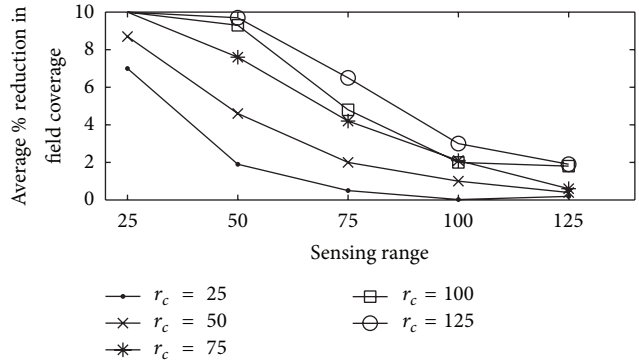


Figure 9: Reduction in field coverage for proposed technique versus sensing range (m) for different communication range (m).

similar amount to that in RIM. The field coverage level before the failure was sustained after our algorithm was applied. This was accomplished by the fact that, because of the increase in the coverage overlap, the replacement nodes needed only to move a short distance or not at all. Moreover, most of the home locations of the replacement nodes were still covered by other neighbour nodes when each node moved for its turn. Furthermore, there were several nodes available for the relocation process. However, networks with sparse node deployment did not have a lot of nodes that could be used to replace the node that failed. Further, a larger area was left unmonitored when the nodes were relocated. The result was a gap in the coverage of the network. The percentage of the field coverage reduction for various sensing and communication ranges is shown in Figure 9. Quite a bit of coverage reduction was noticed when r_c dominated r_s. This was because longer distances needed to be travelled by the nodes between their home area and the position of the node they were replacing. Even so, the reduction was limited to 10% with the proposed algorithm even for $r_c = 6\,r_s$.

4.3.5. Proposed Technique Summary of Results. This section discussed the simulations carried out in OMNeT++ to evaluate the performance of the proposed protocol and the results are compared against the contemporary baseline approaches. Simulation results demonstrate that proposed technique has achieved significant energy savings and enhanced the network lifetime. The results demonstrate that our approach is effective in improving QoS parameters, such as number of exchanged messages, average number of nodes moved, and percentage of reduction in field coverage, when compared with baseline approaches. Table 3 summarizes the findings from this research work.

5. Conclusion

In mobile sensor networks, it is vital that the topology of the connected internodes is maintained. The network can be partitioned because of node failure and this in turn causes the operation of the application to be disrupted. Unlike most previous works that take advantage of the relocation of the nodes so that connectivity can be restored, the proposed algorithm deals with not only the connectivity loss issue but also the issue in the loss of the field coverage. In general, when there is no awareness of the coverage in the process of the resorting connectivity, there is a potential to cause some locations not monitored by any sensor. To solve this, the proposed algorithm does not relocate the nodes permanently. The recovery failure is the responsibility of the neighbouring nodes. The failed node's neighbours coordinate amongst themselves to decide on each of their roles in the process of

Table 3: Results summary.

Protocol compared (125 nodes)	Average travelled distance (m)	Average number of exchanged messages	Average number of moved nodes	Average reduction in field coverage (%)
Proposed technique	2200	200	30	3
RIM	12000	1900	150	11
NN	18000	3500	500	17

the recovery. To restore connectivity, all of the neighbouring nodes taking part in the recovery process relocate themselves to the area of the failed node, one at a time, to provide coverage in that location. After spending an allotted amount of time replacing the failed node, each node returns to the location from which it started off. These neighbouring nodes take turns in this process. For a better application fit, for which the lifetime of the network is most valued, a balance between coverage and connectivity is provided by the proposed algorithm. It provides a balance of the load amongst the failed node's neighbours. The proposed method is a combination of algorithms that are localized and distributed. Messaging overhead is slight and the approach can be scaled for networks that are not small. Validation of the proposed method is achieved by simulation which has also proven the proposed method's effectiveness.

Conflict of Interests

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

Acknowledgment

The authors would like to thank the Management of Universiti Teknologi Petronas, Malaysia, for their support to this research work.

References

[1] V. Ranga, M. Dave, and A. K. Verma, "Network partitioning recovery mechanisms in WSANs: a survey," *Wireless Personal Communications*, vol. 72, no. 2, pp. 857–917, 2013.

[2] B. Nazir, H. Hasbullah, and S. A. Madani, "Sleep/wake scheduling scheme for minimizing end-to-end delay in multi-hop wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, article 92, 14 pages, 2011.

[3] H. Salarian, K.-W. Chin, and F. Naghdy, "Coordination in wireless sensor-actuator networks: a survey," *Journal of Parallel and Distributed Computing*, vol. 72, no. 7, pp. 856–867, 2012.

[4] B. Nazir and H. Hasbullah, "Energy efficient and QoS aware routing protocol for clustered wireless sensor network," *Computers and Electrical Engineering*, vol. 39, pp. 2425–2441, 2013.

[5] B. Nazir and H. Hasbullah, "Energy balanced clustering in wireless sensor network," in *Proceedings of the International Symposium on Information Technology (ITSim '10)*, pp. 569–574, June 2010.

[6] H. Jun, W. Zhao, M. H. Ammar, E. W. Zegura, and C. Lee, "Trading latency for energy in densely deployed wireless ad hoc networks using message ferrying," *Ad Hoc Networks*, vol. 5, no. 4, pp. 444–461, 2007.

[7] M. Chu, H. Haussecker, and F. Zhao, "Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks," *International Journal of High Performance Computing Applications*, vol. 16, no. 3, pp. 293–313, 2002.

[8] J. Chen, M. Díaz, L. Llopis, B. Rubio, and J. M. Troya, "A survey on quality of service support in wireless sensor and actor networks: requirements and challenges in the context of critical infrastructure protection," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1225–1239, 2011.

[9] H. Hasbullah and B. Nazir, "Region-based energy-aware cluster (REC) for efficient packet forwarding in WSN," in *Proceedings of the International Symposium on Information Technology (ITSim '10)*, pp. 1–6, June 2010.

[10] M. Younis, P. Munshi, and E. Al-Shaer, "Architecture for efficient monitoring and management of sensor networks," in *Proceedings of the IFIP/IEEE Workshop on End-to-End Monitoring Techniques and Services (E2EMON '03)*, pp. 140–146, Belfast, Northern Ireland, September 2003.

[11] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[12] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, and K. K. Saluja, "Sensor deployment strategy for target detection," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pp. 42–48, September 2002.

[13] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Communications of the ACM*, vol. 43, no. 5, pp. 51–58, 2000.

[14] K. Akkaya and M. Younis, "COLA: a coverage and latency aware actor placement for wireless sensor and actor networks," in *Proceedings of the IEEE 64th Vehicular Technology Conference (VTC '06)*, pp. 2649–2653, Montreal, Canada, September 2006.

[15] T. Melodia, D. Pompili, V. C. Gungor, and I. F. Akyildiz, "A distributed coordination framework for wireless sensor and actor networks," in *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '05)*, pp. 99–110, Urbana-Champaign, Ill, USA, May 2005.

[16] X. Bai, Z. Yun, D. Xuan, T. H. Lai, and W. Jia, "Optimal patterns for four-connectivity and full coverage in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 3, pp. 435–448, 2010.

[17] K. Akkaya, M. Younis, and M. Bangad, "Sink repositioning for enhanced performance in wireless sensor networks," *Computer Networks*, vol. 49, no. 4, pp. 512–534, 2005.

[18] X. Liu, L. Xiao, A. Kreling, and Y. Liu, "Optimizing overlay topology by reducing cut vertices," in *Proceedings of the 16th*

Annual International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV '06), pp. 112–119, Newport, RI, USA, May 2006.

[19] S. S. Dhillon and K. Chakrabarty, "Sensor placement for effective coverage and surveillance in distributed sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, pp. 1609–1614, 2003.

[20] M. A. Khan, H. Hasbullah, B. Nazir, I. A. Qureshi, and N. Pirzada, "Simultaneously node relocation algorithm for mobile sensor network," in *Wireless Sensor Networks for Developing Countries*, vol. 366 of *Communications in Computer and Information Science*, pp. 85–95, Springer, 2013.

[21] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: a distributed, scalable solution to the area coverage problem," in *Proceedings of the 6th International Symposium on Distributed Autonomous Robotic Systems*, pp. 299–308, Fukuoka, Japan, 2012.

[22] N. Tamboli and M. Younis, "Coverage-aware connectivity restoration in mobile sensor networks," *Journal of Network and Computer Applications*, vol. 33, no. 4, pp. 363–374, 2010.

[23] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: a survey," *Ad Hoc Networks*, vol. 6, no. 4, pp. 621–655, 2008.

[24] Y. Zhu, J. Song, and F. Dong, "Applications of wireless sensor network in the agriculture environment monitoring," *Procedia Engineering*, vol. 16, pp. 608–614, 2011.

[25] J. Chen and X. Koutsoukos, "Survey on coverage problems in wireless ad hoc sensor networks," in *IEEE SoutheastCon*, Richmond, Va, USA, March 2007.

[26] L. Bao and J. J. Garcia-Luna-Aceves, "Topology management in ad hoc networks," in *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '03)*, pp. 129–140, Annapolis, Md, USA, June 2003.

[27] S. Yang, F. Dai, M. Cardei, J. Wu, and F. Patterson, "On connected multiple point coverage in wireless sensor networks," *International Journal of Wireless Information Networks*, vol. 13, no. 4, pp. 289–301, 2006.

[28] G. Wang, G. Cao, and T. La Porta, "Movement-assisted sensor deployment," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '04)*, pp. 119–115, Hong Kong, March 2010.

[29] N. Heo and P. K. Varshney, "Energy-efficient deployment of intelligent mobile sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics Part A*, vol. 35, no. 1, pp. 78–92, 2005.

[30] G. Wang, G. Cao, and T. L. Porta, "Proxy-based sensor deployment for mobile sensor networks," in *Proceedings of the IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS '04)*, pp. 493–502, Fort Lauderdale, Fla, USA, October 2004.

[31] G. Wang, G. Cao, T. La Porta, and W. Zhang, "Sensor relocation in mobile sensor networks," in *Proceedings of the 24th Annual Joint Conference of the IEEE Conference on Computer Communications Societies (INFOCOM '05)*, vol. 4, pp. 2302–2312, Miami, Fla, USA, March 2005.

[32] J. Wu and S. Yang, "SMART: a scan-based movement assisted sensor deployment method in wireless sensor networks," in *Proceedings of the 24th International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05)*, pp. 2313–2324, Miami, Fla, USA, March 2005.

[33] A. A. Abbasi, K. Akkaya, and M. Younis, "A distributed connectivity restoration algorithm in wireless sensor and actor networks," in *Proceedings of the 32nd IEEE Conference on Local Computer Networks (LCN '07)*, pp. 496–503, Dublin, Ireland, October 2007.

[34] K. Akkaya, F. Senel, A. Thimmapuram, and S. Uludag, "Distributed recovery from network partitioning in movable sensor/actor networks via controlled mobility," *IEEE Transactions on Computers*, vol. 59, no. 2, pp. 258–271, 2010.

[35] P. Basu and J. Redi, "Movement control algorithms for realization of fault-tolerant ad hoc robot networks," *IEEE Network*, vol. 18, no. 4, pp. 36–44, 2004.

[36] K. Akkaya and M. Younis, "C2AP: coverage-aware and connectivity-constrained actor positioning in wireless sensor and actor networks," in *Proceedings of the IEEE International Performance, Computing, and Communications Conference (IPCCC '07)*, pp. 281–288, 2007.

[37] K. Akkaya and M. Younis, "Coverage and latency aware actor placement mechanisms in WSANs," *International Journal of Sensor Networks*, vol. 3, no. 3, pp. 152–164, 2008.

[38] A. Abbasi, U. Baroudi, M. Younis, and K. Akkaya, "C^2AM: an algorithm for application aware movement-assisted recovery in wireless sensor and actor networks," in *Proceedings of the International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, pp. 655–659, ACM, Leipzig, Germany.

[39] M. Younis, S. Lee, and A. A. Abbasi, "A localized algorithm for restoring internode connectivity in networks of moveable sensors," *IEEE Transactions on Computers*, vol. 59, no. 12, pp. 1669–1682, 2010.

[40] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment.," in *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems and Workshops*, pp. 1–10, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), Marseille, France, 2008.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 359868, 13 pages
http://dx.doi.org/10.1155/2014/359868

Research Article

Extraction of Multilayered Social Networks from Activity Data

Katarzyna Musial,[1] Piotr Bródka,[2] Przemysław Kazienko,[2] and Jarosław Gaworecki[3]

[1] *School of Natural and Mathematical Sciences, Department of Informatics, King's College London, London WC2R 2LS, UK*
[2] *Institute of Informatics, Wrocław University of Technology, Wyb. Wyspiánskiego 27, 50-370 Wrocław, Poland*
[3] *Research & Engineering Center Sp. z o.o., Ulica Strzegomska 46B, 53-611 Wrocław, Poland*

Correspondence should be addressed to Przemysław Kazienko; kazienko@pwr.wroc.pl

Received 23 January 2014; Accepted 6 March 2014; Published 2 July 2014

Academic Editors: H. R. Karimi, Z. Yu, and W. Zhang

The data gathered in all kinds of web-based systems, which enable users to interact with each other, provides an opportunity to extract social networks that consist of people and relationships between them. The emerging structures are very complex due to the number and type of discovered connections. In web-based systems, the characteristic element of each interaction between users is that there is always an object that serves as a communication medium. This can be, for example, an e-mail sent from one user to another or post at the forum authored by one user and commented on by others. Based on these objects and activities that users perform towards them, different kinds of relationships can be identified and extracted. Additional challenge arises from the fact that hierarchies can exist between objects; for example, a forum consists of one or more groups of topics, and each of them contains topics that finally include posts. In this paper, we propose a new method for creation of multilayered social network based on the data about users activities towards different types of objects between which the hierarchy exists. Due to the flattening, preprocessing procedure of new layers and new relationships in the multilayered social network can be identified and analysed.

1. Introduction

Nowadays, for the first time, we have possibility to process big data about interactions and activities of millions of individuals gathered in all sorts of web-based systems. Communication technologies allow us to form large networks, which in turn shape and catalyse our activities. Due to their scale, complexity, and dynamics, these networks are extremely difficult (or impossible) to analyse in terms of traditional social network analysis methods. The analysis of the network data is at the very early stages and requires a lot of efforts in both developing tools and approaches to tackle it as well as understanding the nature and functioning of networks extracted from this data. The process of network creation is not as straightforward as it seems to be. In the web-based systems, users can interact with each other via different communication channels and utilize various services. This implies that the relationships between users can be extracted based on both direct and indirect communications. The former one is, for example, sending emails or video calls where information is passed directly from one person (group of people) to other(s), whereas to the latter we can count in, for example, commenting objects in the multimedia sharing systems or using the same tags to describe the objects. In both situations there are objects (e.g., email, photo, and tag) that serve as medium in communication between users. The types of these objects differ depending on the web system; for example, at the Internet forum the objects are groups of topics, topics, and posts while in the email service it will be a single message. Additionally, within a single system, the hierarchies of objects can exist.

Extraction of a social network in the environment where users interact with each other using different objects, which create hierarchies, is the main contribution of this paper. In order to perform this task first the hierarchical presocial network (HPSN), where relation between users and objects and between objects exists (Section 5), must be created. After that, the flattening process, in which the hierarchy of objects is removed, is performed (Section 6). As a result, the flat presocial network (FPSN) is obtained where the only

connections that exist are between users and one type of the previously chosen objects. Based on FPSN, social network, where only the connections between users exist, is created (Section 7). The whole idea is presented using simple case study of the Internet forum (Section 8). Finally, the real-world experiments were performed and their outcomes are presented in Section 9.

2. Related Work

There are many types of complex network systems. One of the classifications distinguishes between infrastructures and natural complex systems [1]. The former are physical systems (energy and transportation networks) and virtual systems (Internet, WWW, and telecommunication), whereas the latter are biological networks, social networks, food webs, and ecosystems.

The type of complex systems that is investigated in this paper is a social network formed by people who interact with each other or take part in common activities. The concept of social network has been described by different researchers [2–6] and the definition that is commonly and widely used is that *a social network* is a finite set of individuals, who are the nodes of the network and activities or relations between them, which are represented by edges of the network. A social network (SN) commonly represents the mutual communication and/or activity occurring between users as well as their direction, intensity, and profile [7].

It should be noted that, during analysis of social networks, researchers usually take into account only one activity type while in most cases many different types of relationships exist between users. The special type of social networks that allows the representation of many different activities is called a multilayered social network [5, 8–15], a.k.a. multidimensional network [16–18], or multiplex networks [7, 19–22]. Even the same authors use different names for this kind of complex networks; compare, for example, [12, 21]. Overall, due to high complexity, such networks are more difficult to be extracted and analysed than simple one-layered networks.

Sociologists and psychologist typically create questionnaires and perform interviews in order to collect data which allow them to create and analyse social networks. However, nowadays, the rapid development of the Internet and telecommunication together with the ease of gathering vast amount of data have created the possibility for IT systems to provide vast amount of information about users activities. As a result, researchers have now easy access to big datasets about people's activities ready to analyse. Social networks can be extracted from, for example, bibliographic data [23], blogs [24], photos sharing systems like Flickr [25], e-mail systems [26], telecommunication data [27, 28], social services like Twitter [29] or Facebook [30, 31], video sharing systems like YouTube [32], Wikipedia [33], and much more. Moreover, the whole separate systems were created only for the extraction, aggregation, and visualization of social networks [34, 35].

Nevertheless, as mentioned before, only few scientists have focused their research interests on multilayer social network extraction from activity data [14, 16, 18, 20, 22, 36–39].

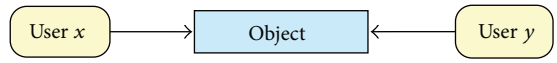


Figure 1: The object-based relationship in the social network on the Internet.

Moreover, no one has studied the hierarchy and relationships between objects in this data. The only hierarchical dependencies in the social networks that were analysed were associated with the hierarchy between users such as employee-employer and the employee-manager [40]. Thus, the analysis of hierarchy between objects presented in this paper is a new approach to extract multilayer social networks from the activity data.

3. Object-Based Relationships

In web-based social systems, there is always an object that plays a role of "a middleman" in a relationship between two users (Figure 1) [41]. In the case of direct communication, people send e-mails to each other or make videoconferences or phone calls via VoIP services. In those cases all participants are aware of the existing relationship. However, sometimes two users can be in a relationship but they do not maintain it actively and consciously, for example, people who comment on the same blog or participate in the same conference. These types of common activities can result in indirect relationships. The roles of both users in an indirect relationship towards the object can be either the same or different (Figure 2).

Object-based relation with equal roles r_{xy}^{aa} means that users x and y meet each other through the object and their role in relation to this object is the same. In other words, they participate in common activity related to the certain object with the same role a; for example, two users take part in the videoconference, two users comment on the same picture, or both of them add the same object to their favourites [25], Figure 2(a). *Object-based relation with different roles* r_{xy}^{ab}, r_{yx}^{ba} is the relation between two users x and y that are connected through the object but their roles a and b towards the object are different; for example, user x comments on a photo (role a, commentator) that was published by y (role b, author) [25], Figure 2(b). The nonzero relation r_{xy}^{ab} entails the nonzero relation r_{yx}^{ba}.

The examples of object-based relations with equal roles are as follows.

(i) *Commentator-commentator*. This relation is created between user x and user y when both of them have added the opinions about at least one common object; for example, they have commented on the same picture at the photo sharing system or on the same post at the forum.

(ii) *Favourite-favourite*. Such a relation from user x to y exists if both users have marked at least one common object as their favourite; for example, they have added the same film to their lists of favourites at the multimedia sharing system.

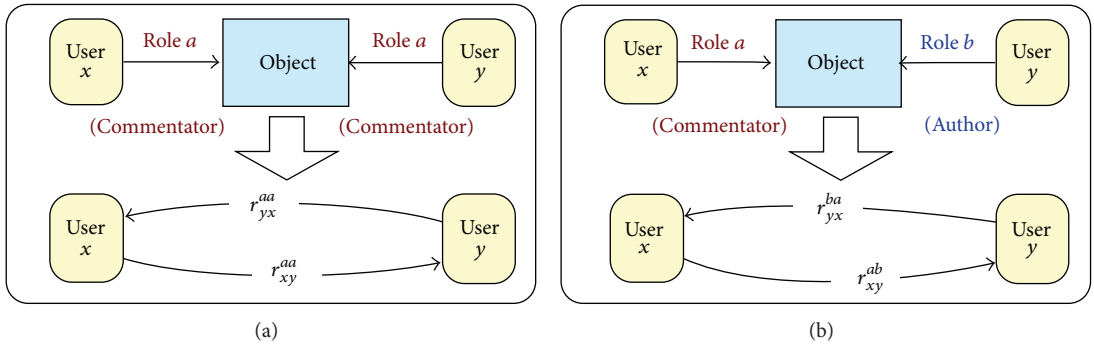


Figure 2: The object-based relation with equal roles: *commentator* (a) and different roles: *commentator* and *author* (b).

(iii) *Author-author*. Such a relation from user x to y exists when they are coauthors of at least one object; for example, they have written a scientific article together.

(iv) *Membership in the group/forum*. This relation from user x to y exists when both of them belong to at least one group together; for example, they belong to a group that gathers people who like dogs at the photo sharing system.

(v) *Utilization of keywords to describe objects (tags)*. Such a relation exists between two users if they use at least one common tag to describe their objects; for example, two users are in relation with each other at the photo sharing system when they use a word "cat" to describe some of their photos.

On the other hand, the examples of object-based relations with different roles can be as follows.

(i) *Opinion-author and author-opinion*. These relations between user x and y exist when user x commented on at least one object that is authored by user y.

(ii) *Favourite-author and author-favourite*. These relations between users x and y exist when user x added to its favourite list at least one object authored by user y.

(iii) *Citation-author and author-citation*. These relations between users x and y exist when user x quoted at least one object authored by user y.

4. Hierarchies between Objects

In all web-based social networks analysed in the literature, the relationships between users were extracted mainly based on a given type of communication or common activity. For example, if two users send e-mails to each other, then the relationship between them in the social network may be established. However, both user communication and common activities are always related somehow to the objects which serve as a medium in interactions between users and their common activities; see Section 3. This object may be "a message" in the case of e-mail exchange, "a video" in YouTube, or "a topic" in the Internet forums. These objects connect either a pair of users (an e-mail sent to a single recipient) or many users simultaneously (an e-mail passed to multiple recipients, a video commented on by many users, and a forum with many members). Besides, the IT system may provide many different functions, which can result in various user activities towards objects of different types. For example, an Internet forum may consist of topics aggregated into groups. Topics, in turn, contain a list of posts. Thus, the objects that enable interactions between users are in hierarchical relationships, Figure 3. Depending on the functionalities of the system, users can moderate a topic group, can subscribe to a topic (be a member of the topic), or provide their opinions about posts (play the role of commentator), Figure 10. Hence, users "meet" each other by performing activities towards objects that belong (i) to one specific level in the object hierarchy (many users can comment on a post authored by another user) or (ii) to two different levels of this hierarchy; for example, a moderator of the group topic is in the indirect relation with authors of the posts.

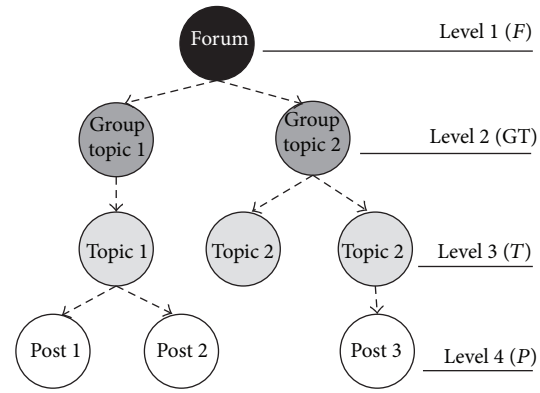


Figure 3: Hierarchies between objects in the Internet forum.

5. Hierarchical Presocial Network

In order to create the hierarchical presocial network (HPSN) based on gathered activity data, first such elements as users, objects, and hierarchy between objects and relations between users and objects need to be extracted. HPSN contains information about relations between users and objects towards

which users performed some activities. The main characteristic of HPSN is that there exist hierarchies between different objects. The whole process of HPSN extraction consists of the following four steps.

(1) *User extraction*. Users are network nodes both in the presocial network and in the final social network. Users perform different activities towards various types of objects; for example, they send e-mails to each other or comment on the photos uploaded by others. These activities are the basis to create the role of a user in relation to a specific object, for example, author, commentator, and so forth.

(2) *Object extraction*. Objects are the nodes in both hierarchical and in flat presocial network, that is, elements through which users communicate with each other (e.g., email and phone call) or items towards which users perform some activities (e.g., photo, video, and tag).

(3) *Extraction of the hierarchy between the objects*. Some objects can be in hierarchical relation with other objects; for example, an object "group of topics" contains one or more "topics" which may include many "posts" (Figure 3). The consequence of the existence of the hierarchy between different objects types is that the objects on the lower level cannot exist without objects on the higher level. These hierarchies exist within HPSN and are removed during the prenetwork flattening process (see Section 6).

(4) *Extraction of the relations between users and objects*. Relation between a given user a and object X exists if user a performed some activities towards object X; for example, user a commented on photo X. The type of activity that user performed towards an object is assigned to each relationship.

The concept of HPSN is presented in Figure 4 where the hierarchy between objects has three levels (Z, V, and Y); at each level some objects exist (e.g., at the level Y objects OB $Y1$ and OB $Y2$) with which users (a, b, and c) are in different types of relations (Z, V, and Y roles); that is, users performed some activities towards these objects.

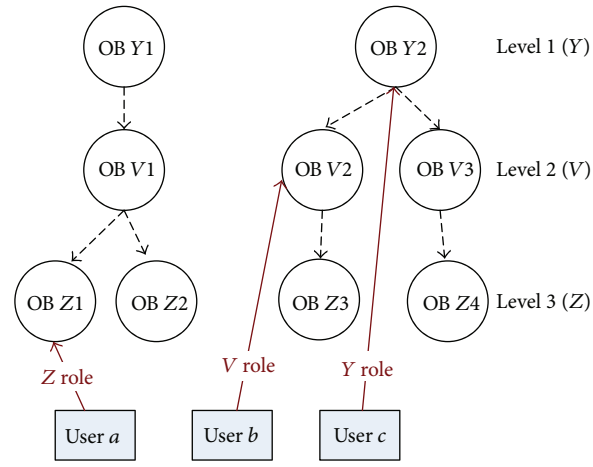


Figure 4: The concept of HPSN.

6. Flat Presocial Network

The flattening process aims at removing relationships between objects (hierarchies). A consequence of this process is based on the knowledge about existing hierarchies; both new user roles and relationships between users are created. The transformation from the presocial network where the hierarchies between objects exist (HPSN) into the flat presocial network (FPSN) without hierarchies will be performed in the following steps.

(a) The operator chooses the level in the hierarchy to which the flattening process will be performed—the end level. Note that, after each flattening process, the only object type in FPSN will be the one that is on the end level selected by the operator and all users will be in relation only towards these objects.

(b) If there exist levels that are lower in the hierarchy than the end level (Figure 5), then for those levels the *bottom-top* approach is used; that is, we get the following.

(i) Relationships between people and objects existing on the hierarchy levels that are below the end level (relation user a, OB $Z1$, and user b, OB $V2$, in Figure 5) are changed. The relation between a user and an object from the lower lever is moved to the upper level by

(1) identification of an object on the upper level that is "a father" of the object from the lower level ("child");
(2) creation of a new relation between the user and "the father" object;
(3) name of the relation between user and "father" object which is created by adding to the name of the relation user "child" the word that denotes the movement from the lower level. For example, in Figure 5, the relation user a, OB $Z1$ ("child"), has a name, Z *role*, and the name of the new relation user a, OB $V1$ ("father"), is ZV *role*;
(4) deletion of the relation between the user and the "child" object from the lower level.

NOTE. This process is repeated for other upper levels until the end level is reached (Figure 5).

(ii) Relationships between people and objects existing at the end level remain unchanged (relation user c, OB $Y2$).

The final FPSN presented in Figure 6 is an outcome of the bottom-up approach where the HPSN from Figure 5 is flattened to level 1 (Y).

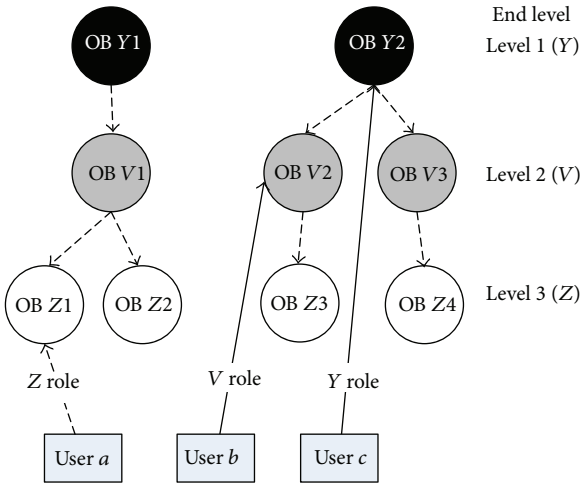


Figure 5: Relations between users and objects in the hierarchical presocial network HPSN.

(c) If there exist levels that are upper in the hierarchy than in the end level (Figure 7), then the top-bottom approach is applied for these levels; that is, we get the following.

(i) Relationships between people and objects existing on the hierarchy levels that are above the end level (relation user b, OB $V2$, and user c, OB $Y2$, in Figure 7) are changed. The relation between user and object from the upper lever is moved to the lower level by

(1) identification of all objects on the lower level that are "children" of an object from the upper level ("father");
(2) creation of the relation between the user and all "child objects;"
(3) name of the relation between the user and "child object" which is created by adding to the origin name information about the "child object." For the example, in Figure 7, in the relation user b, OB $V2$ ("father"), the relation name was V role and the new relation user b, OB $Z3$ ("child"), will have the name VZ role;
(4) deletion of the relation between the user and "father object" on the upper level.

NOTE. This process is repeated until the end level is reached (Figure 8).

(ii) Relationships between people and objects existing on the end level remain unchanged.

An example of top-bottom approach is presented below where the HPSN from Figure 7 is flattened to level 3 (Z), that is, to FPSN in Figure 8.

The goal of the flattening process is to facilitate the extraction of the unified structure that represents the social connections between pairs of users from user activity data and relations between objects. New types of user roles can be identified during the flattening process, for example, VZ or YVZ role in Figure 8. The newly obtained knowledge about these roles gives an opportunity to investigate the complex profile of user relationships in more detail and in consequence enables their more comprehensive analysis.

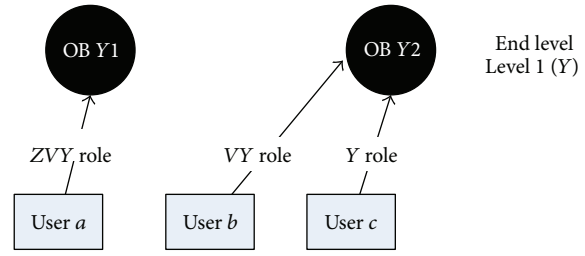


Figure 6: Relations between users and objects in the final FPSN after the bottom-top flattening process.

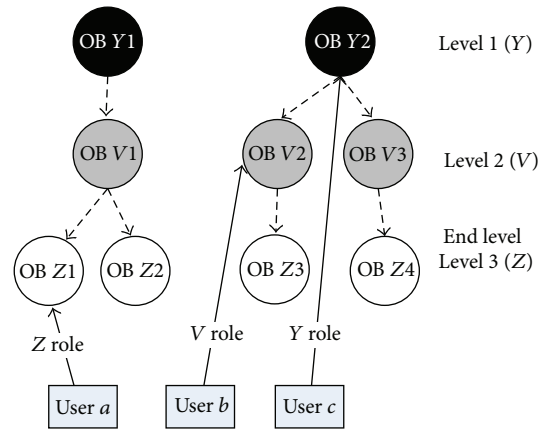


Figure 7: Relations between users and objects in the hierarchical presocial network HPSN.

7. Social Network

The flat presocial network structure (FPSN) is used to extract the social network (SN) where the relations user-object from FPSN no longer exist. These connections are converted into direct relations between users in SN. The process consists of the following steps.

(a) Extraction of SN layers is based on the type of the users' roles towards objects. Each network layer consists of users and their connections. In a single layer network, there exist object-based relationships of only one type either with equal or different roles (see Section 3).

(b) The operator chooses which social network layers need to be created.

(c) The operator chooses the SN model,

n-graph, each layer in the multilayered social network SN is represented by separate social network;

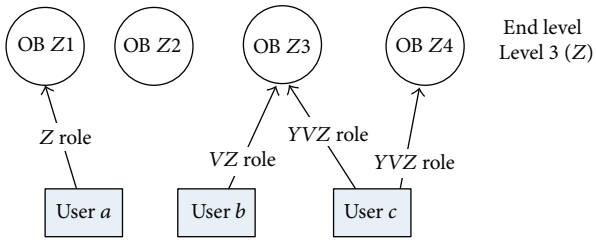


Figure 8: Relations between users and objects in the final FPSN after the top-bottom flattening process.

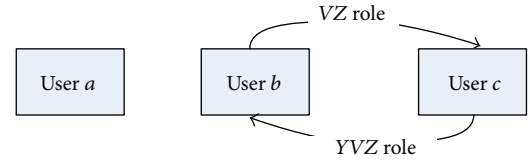


Figure 9: Social network created from FPSN presented in Figure 8.

multigraph, all layers are represented by a single social network and different layers are distinguished by different colours of edges (or another labelling mechanism is used).

(d) Extraction of relations *user_from-user_to* is by calculation of the relationship strengths and colours (labels) between SN nodes (users) using activity data stored in FPSN. There are many possible formulas for calculating the relationship strength. Most of them are based on the normalized quantity of shared user activities towards objects in FPSN (for some of the examples please see [25, 42]).

Social network SN created from FPSN in Figure 8 according to the process described above is presented in Figure 9.

Depending on the goal of analysis, the strength of a relationship can be a static measure calculated based on all available data and taking into consideration the number of activities of a given type. On the other hand, we can take into account time factor and split the data according to the time when activities occurred. In the latter case, the whole period from which the data comes from is divided into time frames and the relationship strength is calculated for each slot separately. The time frame can be created using two approaches as follows.

(i) *Sliding window*. A user defines the length of the time window (e.g., t_l) and the time interval that is used to move the window (e.g., t_i). In order to extract time frames the whole period of the length t_l is moved by t_i. In consequence the entire dataset is divided into partly overlapping frames. Note that both time window and time interval need to be specified in a way that the period from the start date to end date should be completely covered.

(ii) *Equal, separate periods*. A user sets the number of periods, for example, k, and then the data is divided into k separate, equal periods according to the dates of activity. This is equivalent to the situation (i) where t_l equals t_i.

After the time windows are created, the weight is assigned to each of them. Usually, the recent periods are more important than the previous ones and because of that greater weight is assigned to those recent time windows.

In this paper we present how to calculate the static version of relationship strength for two different types of relations: (i) object-based relations with equal roles and (ii) object-based relations with different roles (see Figure 2).

(a) Object-Based Relationships with Equal Roles. The object-based relationship with equal roles denotes a connection in which two users are related to each other through the object and their roles towards this object are the same; see Section 3. Note that the same formula is used in order to calculate the connection strength between user x and user y who (i) have commented on at least one common object, (ii) have marked at least one common object as their favourite, (iii) are coauthors of an object, (iv) are in the same group or forum, and (v) have used the same keywords to describe objects. In all of these relations, there is an object on which both users perform specific activity. To calculate the static strength of the relationship, the following formula may be applied:

$$s^a_{xy} = \frac{n^a_{xy}}{n^a_x}, \quad x \neq y, \tag{1}$$

where a is the type of activity that is performed by users towards an object, for example, membership to a group/forum, utilization of a tag to describe objects, coauthorship of an object, commenting on an object, and so forth. n^a_{xy} is the number of common a activities for users x and y performed together, for example, number of groups/forums to which both users x and y belong, the number of tags that both users x and y use commonly or the number of objects that were coauthored by both users x and y, and so forth. n^a_x is the number of a given a activity for user x, for example, the number of groups/forums to which user x belongs, the number of tags used by user x or the number of objects authored by user x, and so forth.

Let us consider the situation in which users of multimedia sharing system utilise tags to describe different multimedia content. In this case an object is *a tag* and relationship between two users is created when they utilise some common tags. Let us assume that the data obtained from the system contains the following information: user x utilised 20 identical tags as user y and user x used 60 tags in total. Then relation strength from user x to y is calculated as follows:

$$s^y_{ij} = \frac{n^y_{ij}}{n^y_i} = \frac{20}{60} = \frac{1}{3}. \tag{2}$$

(b) Object-Based Relationships with Different Roles. The object-based relation with different roles denotes a connection in which two users are related to each other through the object and their roles towards this object are different (see

Section 3). For example, one user can comment in a forum in which another user is a moderator and the relationship between users is moderator-commentator. Thus, in the case of calculating the relations strength, we will refer to the relations *activity_type_a-activity_type_b*.

Considering the relation *activity_type_a-activity_type_b* its strength will be calculated as follows:

$$s_{xy}^{ab} = \frac{n_{xy}^{ab}}{n_x^{ab}}, \quad x \neq y, \ a \neq b, \tag{3}$$

where a denotes the 1st activity type; b denotes the 2nd activity type (different that a); n_{xy}^{ab} is the number of activities a performed by user x towards objects for which user y performed activity b; n_x^{ab} is the total number of activities of type a performed by user x towards objects for which any other users performed activity b.

As an example, let us consider the case where a user adds to the list of favourites an object authored by another user. A relationship between two users is created when one user adds to its favourites an object authored by another person. Assume that the following data is available in the system: user x added to favourites 20 objects authored by user y. User x added to favourites 60 objects in total. The objects of user y were added to favourites by other 30 times in total. Moreover, a means the activity "authored by" and b means the activity "added to favourite by."

The relation strengths are calculated as follows:

from user x to y: $s_{xy}^{ba} = n_{xy}^{ba}/n_x^{ba} = 20/60 = 1/3$;

from user y to x: $s_{yx}^{ab} = n_{yx}^{ab}/n_y^{ab} = 20/30 = 2/3$.

8. Example of Flattening Process

One of the examples where hierarchies between objects exist is the Internet forum where people can create their own topics that contain posts added by users. The hierarchy between objects within a forum and the activities that can be performed towards these objects are presented in Figure 10.

The hierarchy that will be used for our case study is forum-topic-post. Both relationships between objects and between users and objects in the exemplary hierarchical prenetwork are presented in Figure 11. In order to create the flat presocial network (Figure 12) we perform the flattening process after which the relationships between objects will be removed but at the same time on other levels new relationships between users and objects will be created.

Two types of flattening process can be considered: bottom-top and top-bottom (see Section 6). We present here the bottom-top flattening process in which the relationships will be moved to the highest forum level; that is, the forum will be the final level.

The bottom-top approach applied to HPSN from Figure 11 results in several new relationships and roles (Figure 12). Some examples are enumerated as follows.

(i) User A is an author of post 1.2.1 in the topic 1.2 at the forum 1 and also of post 2.1.1 in the topic 2.1 at forum 2. Then a new relation between user A and forum is created: *PTF Is Author* (*PT, PostTopicForum*, a new name of relation; see Section 6); that is, user A authored at least one post in a topic that is in a given forum. A similar approach is applied to user B, who is an author of post 1.1.1 in the topic 1.1 and to user C, an author of post 2.2.2 in the topic 2.2. Moreover, user D is an author of posts 1.1.2 and 2.2.1. This is flattening (movement) of authorship activities on posts (role *P Is Author*) to the top level forum.

(ii) User B is a commentator of post 2.1.1. A new relation between user B and the forum is created: *PTF Is Commentator*. The same is done for user D, a commentator of posts 1.2.1, 2.1.1, and 2.2.2. In this way, the role *P Is Commentator* is moved to the forum level.

(iii) User A is a moderator of topic 1.1 at the forum; then a new relation between user A and the forum is created:

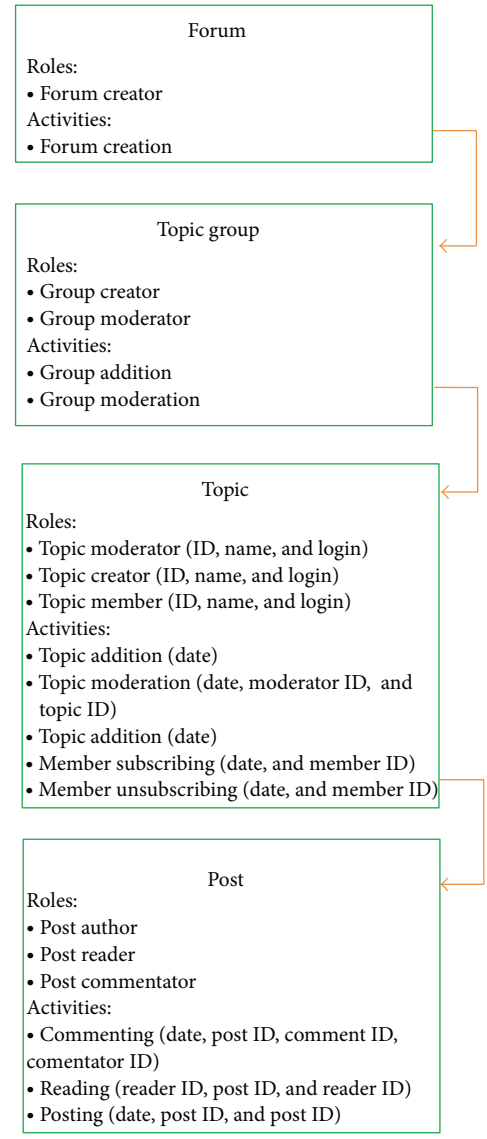


Figure 10: The hierarchy of objects in the forum.

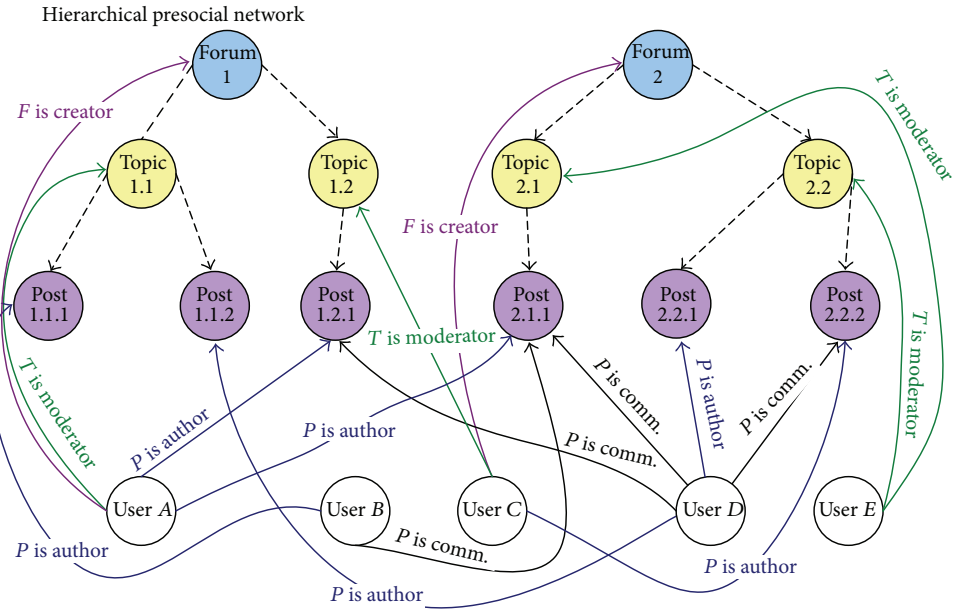


Figure 11: Relations between users and objects together with users' roles towards the objects.

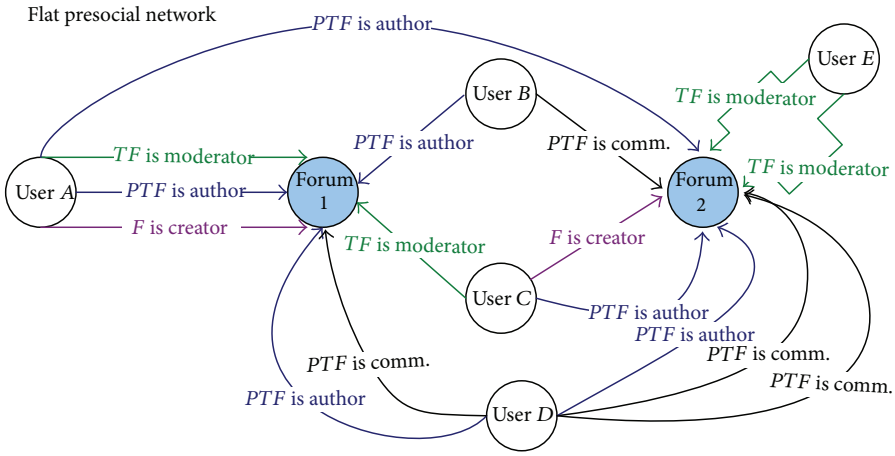


Figure 12: Flat presocial network (FPSN): the result of flattening process applied to HPSN from Figure 11.

TF Is Moderator (*TF, TopicForum*, a new name of relation; see Section 6). The similar method is utilized for user *C*, the moderator of topic 1.2, and user *E*, the moderator of topics 2.1 and 2.2. This is flattening of roles from the topic level to the final forum level.

(iv) *User A* is a creator of the forum 1 and user *C* is a creator of forum 2; then the existing relationship stays unchanged: *F Is Creator*.

Two different layers of the final multilayered SN derived from the flat presocial network FPSN (Figure 12) are presented in Figures 13 and 14. For instance, the relation between two users exists if one user was the moderator of the post that was commented on by another user, Figure 13; for example, user *A* moderates the post commented on by user *D* so there is a relation *moderator-commentator* from *A* to *D* in the final social network SN. This relation is an object-based relationship with different roles. In Figure 14, another layer *PTF Is Author*, *PTF Is Author,* is presented. This is a layer in which the extracted relationships are object-based with equal roles. The relationships' strengths presented on both figures are calculated using formulas from Section 7.

Note that user *D* and user *E* are connected only because of flattening process (Figure 13). The same is with user *E* and user *B* (Figure 13) and user *A* and user *C* (Figure 14). Any of these relationships would be revealed if the flattening process did not take place.

9. Experiments

The real-world dataset used for experiments was obtained from the social web site *extradom.pl*. The analysed dataset covers the period from August 21, 2008, to January 8, 2010.

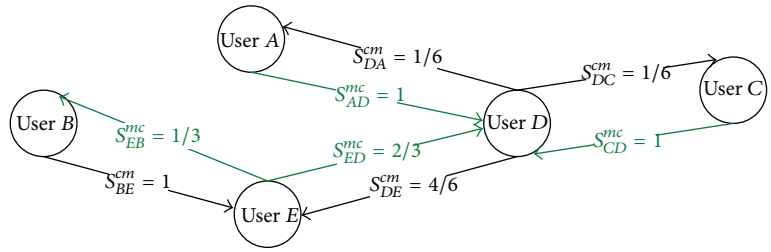


Figure 13: Example of the *PTF Is Commentator* and *MF Is Moderator* layer of the final SN extracted from the flat presocial network FPSN (Figure 12).

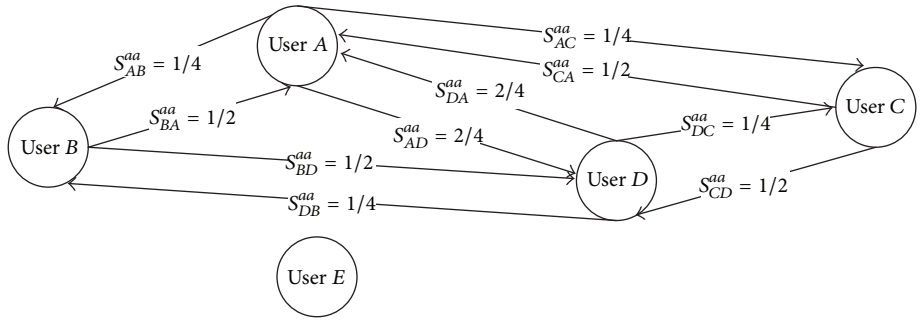


Figure 14: Example the *PTF Author PTF Author* layer of the final SN extracted from the flat presocial network FPSN (Figure 12).

Before the hierarchical presocial network HPSN was created, the dataset was cleansed and validated. Several rules were applied in the cleansing phase. Two most important ones were as follows:

(i) each object must have creation date;

(ii) each object must be assigned to its creator.

There were 104,625 users registered in the portal, but only 4.25% (4,404) of them were active on forum. The number of different types of objects in the forum is shown in Table 1. One or more activity types were identified for each object type. Objects with activities that were performed towards them are shown in Table 2.

As we can see in Table 2, there are two activity types (*topic group moderation* and *post reading*) that are not present in the dataset used for experiments and thus are not included in the further analyses. Additional assumptions that have been made and which helped to detect some of activities are as follows:

(i) a user who creates the first post in the topic will be treated as a topic creator;

(ii) creation of the first topic in the group is simultaneously treated as the creation of the entire group;

(iii) users who create their first post in the topic will be automatically subscribed to this topic.

Table 3 summarizes the profile of the hierarchical presocial network (HPSN) that was created from the extradom.pl dataset.

Three distinct flattening processes (see Section 6) with three separate final object levels, topics group, topic, and post, have been applied to the hierarchical presocial network HPSN. Some of the activities were multiplied after the flattening process (for detailed statistics please see Table 4). Such situation takes place when the hierarchical presocial network is flattened to the object type, which is not the highest level in the hierarchy. For example, when HPSN was flattened to the topic groups (level 2 in the hierarchy, see Figure 3), the forum creation activity was multiplied by 692 because there were distinct 692 groups in the dataset.

Table 1: Objects quantity in the experimental dataset.

Object type	Number of objects
Forum	1
Topic group	692
Topic	2,336
Post	13,272
Comment	49

Table 2: Activities assigned to object types.

Object type	Activity type	Number of activities
Forum	Forum creation activity	1
Topics group	Topic group addition	692
	Topic group moderation	0
Topic	Topic addition	2,336
	Topic member subscribing	5,788
Post	Post reading	0
	Post authoring	13,272
Comment	Post commenting	49

Table 3: Activities in the hierarchical presocial network.

Activity type	Number of activities	Number of users with the activity	Percentage of total users with a given activity
Forum creation activity	1	1	0.02%
Topic group addition	692	266	6.04%
Topic addition	2,336	1,464	33.24%
Topic member subscribing	5,788	4,359	98.98%
Post posting	13,272	4,359	98.98%
Post commenting	49	39	0.89%

Table 4: Number of activities before and after flattening; "+" denotes activates which were created during a given flattening process.

Activity type	Number of activities before flattening (in HPSN)	Number of activities after flattening (in FPSN)					
		New	Final objects: topic groups	New	Final objects: topics	New	Final objects: posts
Forum creation activity	1	+	692	+	2336	+	1,3272
Topic group addition	692		692	+	2336	+	13,272
Topic addition	2,336		2,336		2336	+	13,272
Topic member subscribing	5,788		5,788		5788		793,245
Post authoring	13,272		13,272		13,272		13,272
Post commenting	49		49		49		49

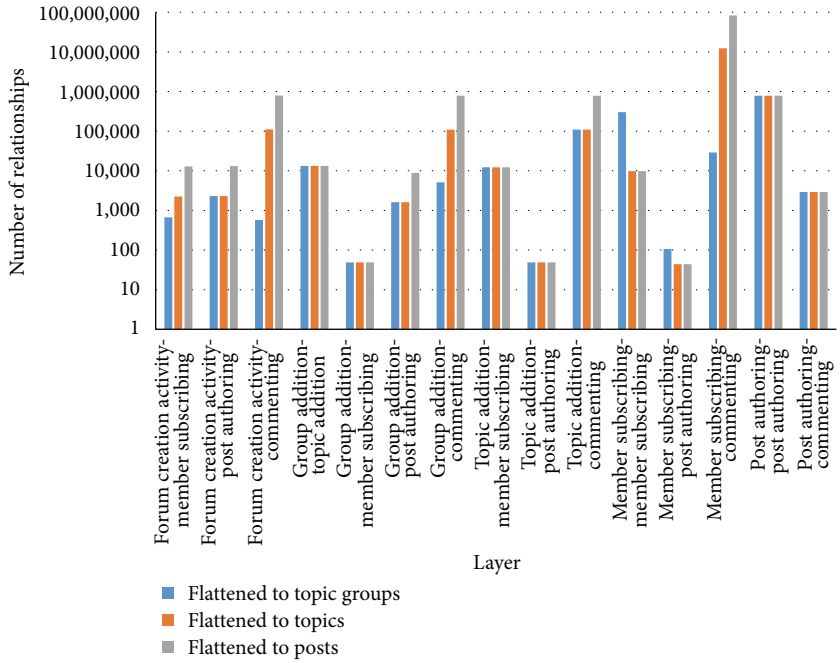


Figure 15: Number of activities (relationships) within created layers for three different flattening processes.

Once the flattening process has been accomplished, the separate layers in the multilayered social network were identified and relationships between users within these layers were extracted (see Section 7). Both layers and number of distinct relationships existing within each layer are presented in Table 5.

As a result of the flattening process 14 new layers in the multilayered social network were created. Additionally, percentages of new user relationships in SN are 90% in the case of topic groups as the final object, 10% for topics, and 3% for posts (see Table 5 and Figure 15). Note that these new relationships would not be visible without the flattening process. It means that the method of preprocessing with flattening of object relations reveals completely new knowledge about the complexity of connections between people.

Table 5: Layers in SN and their profile for three different flattening processes.

Layer	New	Moved	Number of pairs of users (relationships) with common activities on the layer of SN: Topic groups	Moved	Topics	Topics/topic groups	Moved	Posts	Posts/topics
Forum creation activity—group addition	+		671		2,252	3.36		12,922	5.74
Forum creation activity—topic addition	+		2,313		2,313	1.00		13,157	5.69
Forum creation activity—member subscribing	+		5,760		111,544	19.37		792,732	7.11
Forum creation activity—post authoring	+		13,233		13,233	1.00		13,233	1.00
Forum creation activity—commenting	+		49		49	1.00		49	1.00
Group addition—topic addition	+		1,614		1,614	1.00		8,821	5.47
Group addition—member subscribing	+		5,116		109,377	21.38		780,312	7.13
Group addition—post authoring	+		12,206		12,206	1.00		12,206	1.00
Group addition—commenting	+		49		49	1.00		49	1.00
Topic addition—member subscribing		+	109,376		109,376	1.00	+	780,445	7.14
Topic addition—post authoring	+		302,961		9,849	0.03		9,849	1.00
Topic addition—commenting	+		1,060		44	0.04		44	1.00
Member subscribing—member subscribing		+	289,740	+	12,342,690	42.60	+	83,495,278	6.76
Member subscribing—post authoring	+		779,973		779,973	1,00		779,973	1.00
Member subscribing—commenting	+		2,903		2,903	1.00		2,903	1.00
Post authoring—post authoring	+		2,449,226		376,978	0.15		0	0
Post authoring—commenting		+	9,340	+	1,703	0.18		44	0.03
Sum:			3,985,590 New: 90% Moved: 10%		13,876,153 New: 10% Moved: 90%	3.48		86,702,017 New: 3% Moved: 97%	6.25

10. Conclusions

The wide variety and availability of Web 2.0 systems, where users can interact with each other and perform different types of activities, give us an opportunity, by analysing the large-scale data gathered in these systems, to better understand human social behaviour. A very interesting research problem is to investigate social connections that emerge between people based on their shared activities. However, the extraction of these relations is not a trivial task. The main reason is that user behaviour in such systems is often very complex due to the variety of available services and functionalities. As presented in the paper, people can perform different activities towards different objects. Additional challenge is that the relationships existing between these objects can form a hierarchical structure. In this paper, we propose the process to extract the multilayered social network from the data about both user behaviours and relations between objects. The whole method consists of three main phases: (i) extraction of the hierarchical presocial network HPSN, (ii) creation of the flat presocial network PFSN, and finally (iii) creation of the multilayered social network SN. We believe that such systematic approach to the problems is necessary to be able to cope with the massive volume of data being generated by social-based systems every day. Moreover, the proposed process is generic and robust in a way that it is able to accommodate new ways of interactions between users.

The new flattening concept enables discovering in the multilayered social network new layers with new types of relationships which otherwise would not be available for analysis. It is possible due to the presented above process in which the object hierarchy is removed. Thus, the new method of preprocessing enables revealing new information, which is invisible in the regular analysis of user activities and this in turn opens new possibilities for network analysis. The experiments confirmed that some new types of relations between users can be extracted in the flattening process. This enables a deeper insight into analysis of multilayered social networks as more information is included in the final network structure.

Conflict of Interests

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

Acknowledgments

The authors are indebted to Elżbieta Kukla, Ph.D. holder, and Tomasz Filipowski, M.S. holder, for their valuable discussion. The work was partially supported by Fellowship cofinanced by the European Union within European Social

Fund, The European Commission under the 7th Framework Programme, Coordination, and Support Action, Grant Agreement no. 316097, ENGINE, European Research Centre of Network intelliGence for INnovation Enhancement http://engine.pwr.wroc.pl/, and The National Science Centre, Decision no. DEC-2013/09/B/ST6/02317.

References

[1] A. Barrat, M. Barthelemy, and A. Vespignani, *Dynamical Processes on Complex Networks*, Cambridge University Press, Cambridge, UK, 2008.

[2] L. A. Adamic and E. Adar, "Friends and neighbors on the Web," *Social Networks*, vol. 25, no. 3, pp. 211–230, 2003.

[3] J. Scott, *Social Network Analysis: A Handbook*, SAGE Publications, London, UK, 2000.

[4] S. H. Strogatz, "Exploring complex networks," *Nature*, vol. 410, no. 6825, pp. 268–276, 2001.

[5] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*, Cambridge University Press, New York, NY, USA, 1994.

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

[7] R. Hanneman and M. Riddle, *Introduction to Social Network Methods*, University of California, Riverside, Calif, USA, 2005, http://faculty.ucr.edu/~hanneman/nettext/.

[8] M. S. Baptista, C. Grebogi, and R. Köberle, "Dynamically multilayered visual system of the multifractal fly," *Physical Review Letters*, vol. 97, no. 17, Article ID 178102, 2006.

[9] P. Bródka, P. Kazienko, K. Musiał, and K. Skibicki, "Analysis of neighbourhoods in multi-layered dynamic social networks," *International Journal of Computational Intelligence Systems*, vol. 5, no. 3, pp. 582–596, 2012.

[10] P. Bródka, K. Skibicki, P. Kazienko, and K. Musiał, "A degree centrality in multi-layered social network," in *Proceedings of the International Conference on Computational Aspects of Social Networks (CASoN '11)*, pp. 237–242, October 2011.

[11] P. Bródka, P. Stawiak, and P. Kazienko, "Shortest path discovery in the multi-layered social network," in *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining (ASONAM '11)*, pp. 497–501, July 2011.

[12] M. De Domenico, A. Sole-Ribalta, E. Cozzo et al., "Mathematical formulation of multi-layer networks," *Physical Review X*, vol. 3, Article ID 041022, 2013.

[13] P. Kazienko, P. Brodka, and K. Musial, "Individual neighbourhood exploration in complex multi-layered social network," in *Proceedings of the 3rd IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT '10)*, pp. 5–8, September 2010.

[14] M. C. Pasqual and O. L. de Weck, "Multilayer network model for analysis and management of change propagation," *Research in Engineering Design*, vol. 23, no. 4, pp. 305–328, 2012.

[15] F. Darabi Sahneh, C. Scoglio, and P. Van Mieghem, "Generalized epidemic mean-field model for spreading processes over multilayer complex networks," *IEEE/ACM Transactions on Networking*, vol. 21, no. 5, pp. 1609–1620, 2013.

[16] M. Berlingerio, M. Coscia, F. Giannotti, A. Monreale, and D. Pedreschi, "The pursuit of hubbiness: analysis of hubs in large multidimensional networks," *Journal of Computational Science*, vol. 2, no. 3, pp. 223–237, 2011.

[17] 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.

[18] M. Zignani, C. Quadri, S. Gaitto, and G. P. Rossi, "Exploiting all phone media? A multidimensional network analysis of phone users' sociality," http://arxiv.org/abs/1401.3126.

[19] I. Abraham, S. Chechik, D. Kempe, and A. Slivkins, "Low-distortion inference of latent similarities from a multiplex social network," in *Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '13)*, pp. 1853–1872, January 2013.

[20] N. S. Contractor, P. R. Monge, and P. M. Leonardi, "Multidimensional networks and the dynamics of sociomateriality: bringing technology inside the network," *International Journal of Communication*, vol. 5, no. 1, pp. 682–720, 2011.

[21] S. Gómez, A. Díaz-Guilera, J. Gómez-Gardeñes, C. J. Pérez-Vicente, Y. Moreno, and A. Arenas, "Diffusion dynamics on multiplex networks," *Physical Review Letters*, vol. 110, no. 2, Article ID 028701, 2013.

[22] S. Lee and P. Monge, "The coevolution of multiplex communication networks in organizational communities," *Journal of Communication*, vol. 61, no. 4, pp. 758–779, 2011.

[23] 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.

[24] N. Agarwal, M. Galan, H. Liu, and S. Subramanya, "WisColl: collective wisdom based blog clustering," *Information Sciences*, vol. 180, no. 1, pp. 39–61, 2010.

[25] P. Kazienko, K. Musiał, and T. Kajdanowicz, "Multidimensional social network in the social recommender system," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 41, no. 4, pp. 746–759, 2011.

[26] J. R. Tyler, D. M. Wilkinson, and B. A. Huberman, "Email as spectroscopy: automated discovery of community structure within organizations," in *Communities and Technologies*, pp. 81–96, Kluwer BV, Deventer, The Netherlands, 2003.

[27] 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.

[28] P. Kazienko, D. Ruta, and P. Bródka, "The impact of customer churn on social value dynamics," *International Journal of Virtual Communities and Social Networking*, vol. 1, no. 3, pp. 60–72, 2009.

[29] B. A. Huberman, D. M. Romero, and F. Wu, "Social networks that matter Twitter under the microscope," *First Monday*, vol. 14, no. 1, 2009.

[30] N. B. Ellison, C. Steinfield, and C. Lampe, "The benefits of facebook "friends:" social capital and college students'use of online social network sites," *Journal of Computer-Mediated Communication*, vol. 12, no. 4, pp. 1143–1168, 2007.

[31] A. L. Traud, E. D. Kelsic, P. J. Mucha, and M. A. Porter, "Comparing community structure to characteristics in online collegiate social networks," *SIAM Review*, vol. 53, no. 3, pp. 526–543, 2011.

[32] X. Cheng, C. Dale, and J. Liu, "Statistics and social network of YouTube videos," in *Proceedings of the 16th International Workshop on Quality of Service (IWQoS '08)*, pp. 229–238, June 2008.

[33] A. Capocci, V. D. P. Servedio, F. Colaiori et al., "Preferential attachment in the growth of social networks: the internet encyclopedia Wikipedia," *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 74, no. 3, Article ID 036116, 2006.

[34] P. Mika, "Flink: semantic Web technology for the extraction and analysis of social networks," *Web Semantics*, vol. 3, no. 2-3, pp. 211–223, 2005.

[35] Y. Matsuo, J. Mori, M. Hamasaki et al., "POLYPHONET: an advanced social network extraction system from the Web," *Web Semantics*, vol. 5, no. 4, pp. 262–278, 2007.

[36] P. Bródka and P. Kazienko, *Multi-Layered Social Networks. Encyclopedia of Social Network Analysis and Mining*, Springer, 2014.

[37] P. Kazienko, P. Bródka, K. Musial, and J. Gaworecki, "Multi-layered social network creation based on bibliographic data," in *Proceedings of the 2nd IEEE International Conference on Social Computing (SocialCom '10)*, pp. 407–412, August 2010.

[38] R. Michalski, P. Kazienko, and J. Jankowski, "Convince a dozen more and succeed—the influence in multi-layered social networks," in *Proceedings of the 9th International Conference on Signal Image Technology & Internet based Systems*, pp. 499–505, IEEE Computer Society, 2013.

[39] T. J. Zagenczyk, R. L. Purvis, M. K. Shoss, K. L. Scott, and K. S. Cruz, "Social influence and leader perceptions: multiplex social network ties and similarity in leader-member exchange," *Journal of Business and Psychology*, 2013.

[40] A. Lomi, D. Lusher, P. E. Pattison, and G. Robins, *Inter-Organizational Hierarchies, Social Networks, and Identities in Multi-Unit Organizations*, American Sociological Association, New York, NY, USA, 2007, http://www.allacademic.com/meta/p183413_index.html.

[41] K. Musiał and P. Kazienko, "Social networks on the Internet," *World Wide Web*, vol. 16, no. 1, pp. 31–72, 2013.

[42] G. Caldarelli and A. Vespignani, Eds., *Large Scale Structure and Dynamics of Complex Networks, from Information Technology to Finance and Natural Science, Complex Systems and Interdisciplinary Science*, vol. 2, World Scientific, 2007.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 180590, 14 pages
http://dx.doi.org/10.1155/2014/180590

Research Article

A Social Diffusion Model with an Application on Election Simulation

Jing-Kai Lou,[1,2] Fu-Min Wang,[3] Chin-Hua Tsai,[3] San-Chuan Hung,[3] Perng-Hwa Kung,[3] Shou-De Lin,[3] Kuan-Ta Chen,[1] and Chin-Laung Lei[2]

[1] *Institute of information Science, Academia Sinica, 128 Academia Road, Section 2, Nankang, Taipei 115, Taiwan*
[2] *Department of Electrical Engineering, National Taiwan University, No. 1, Section 4, Roosevelt Road, Taipei 10617, Taiwan*
[3] *Department of Computer Science and Information Engineering, National Taiwan University, No. 1, Section 4, Roosevelt Road, Taipei 10617, Taiwan*

Correspondence should be addressed to Jing-Kai Lou; kaeaura@gmail.com

Received 27 December 2013; Accepted 17 March 2014; Published 5 June 2014

Academic Editors: H. R. Karimi, Z. Yu, and W. Zhang

Issues about opinion diffusion have been studied for decades. It has so far no empirical approach to model the interflow and formation of crowd's opinion in elections due to two reasons. First, unlike the spread of information or flu, individuals have their intrinsic attitudes to election candidates in advance. Second, opinions are generally simply assumed as single values in most diffusion models. However, in this case, an opinion should represent preference toward multiple candidates. Previously done models thus may not intuitively interpret such scenario. This work is to design a diffusion model which is capable of managing the aforementioned scenario. To demonstrate the usefulness of our model, we simulate the diffusion on the network built based on a publicly available bibliography dataset. We compare the proposed model with other well-known models such as independent cascade. It turns out that our model consistently outperforms other models. We additionally investigate electoral issues with our model simulator.

1. Introduction

Huge success of viral marketing nowadays clearly shows that acquaintances indeed greatly influence people adopting a new or different opinion. This implicates that people, in a way, attempt to plant their intrinsic ideas, opinions, or preferences in others' minds through exchanging opinions over and over in different circumstances. One interesting and long-discussed scenario is election. Elections in the modern world are an essential mechanism to aggregate the opinions of the masses and to make joint decisions for a variety of purposes. People share thoughts and even attempt to convince others to adopt their attitudes during the election season.

As social media such as Facebook are widely utilized, it becomes quite convenient for people to manifest themselves. Social media exposure grants people a hitherto wide range to deliver their views. Social media extremely accelerates and facilitates such opinion-exchange interactions among individuals. As opinions interflow, the intrinsic opinions of an irresolute person could eventually be assimilated to those of the determined ones. Then a consensus or a public opinion appears.

From a research aspect, understanding the progress of human negotiation benefits the real world applications. For instance, social scientists would wonder to what extent the opinions' exchange among friends can affect each other's viewpoints. Campaign companies would inquire how to promote a candidate with limited budgets. Such questions are not easy to answer via a user study, particularly when the number of participants becomes huge.

Opinion diffusion on social networks has been studied for decades. Unfortunately, many previous models, such as the Independent Cascade Model, Linear Threshold Model, SIR/SIS model, and heat diffusion model, cannot manage the election scenario intuitively due to the following two reasons. First, people have their intrinsic opinions more or less, which

is absent in the aforementioned models. In such a manner, people may not serve as neutral relays. People amplify the opinions they stand for and deamplify opinions they stand against. Second, opinions could be multidimensional, for example, a viewpoint for multiple electoral candidates. Most diffusion models adopt single values to represent the opinions for simplicity. A single value clearly cannot directly represent the views about multiple candidates. Our goal is to design a suitable diffusion model which is capable of managing the propagation of viewpoints. Up to date we have not yet seen too many computational approaches with systematic and quantifiable studies on this issue.

Inspired by the real world phenomena, we have realized several preferable properties to manage the information diffusion like opinion polling. The properties are *high-dimensional media*, *input dependence*, *deterministic convergence*, and *consensus*. A summary of the properties is as follows. First, we prefer the media (which represents preference toward candidates) propagated throughout the process being a unit vector because, democratically, individuals (or nodes) have equal rights in casting votes. Second, the preference distribution should be significantly affected by the initial intrinsic preference as well as the neighbors through social network. Finally, we hope the propagation converges eventually, and a common trend appears after numerous interactions [1]. In this paper, we show that our model is the only one satisfying all properties among the existing models.

The novelty and contributions of this paper can be viewed from several different angles.

(1) This work strategically demonstrates a plausible process to answer a set of real world problems.

   (a) We start by designing a preference negotiation model (with theoretical guarantees) to manage high-dimensional information. We assess the quality of this model by proving its convergence and several other important properties.

   (b) We conduct an experiment to demonstrate the validity of our model in predicting the change of citation preference among authors through collaboration networks.

   (c) To build the diffusion simulation framework, we further devise suite of satellite algorithms for preference profile sampling, deployment, and seed-voter selection.

(2) Our case studies, in practice, have provided the concrete solutions to the real world problems of concern.

   (a) The simulation shows that election outcomes can be significantly affected by the social factor. We find that each individual preference profile extremely changes after a preference diffusion. Using Kendal τ coefficient to measure the similarity before and after the diffusion, it usually results in a coefficient below 0.5.

   (b) With the simulation, we additionally examine several well-known voting schemas to verify their vulnerability to vote-buying. Among them, Borda Count voting schema performs best to resist vote-buying. Plurality voting is the most vulnerable to manipulation.

2. Preliminary

We review the previously done works related to the information diffusion and electoral issues.

2.1. Diffusion Model. To develop models for diffusion simulation or prediction, researchers unearth the underlying mechanisms or the inherent patterns of information diffusion from real word phenomena and utilize these findings.

The Linear Threshold Model (LT model for short) [2, 3] and Independent Cascade Model (IC model for short) [3, 4] are the most well-known and fundamental ones to describe how the information propagates step by step in a network. Inspired by the ideas of the two models, various models have been proposed later for more specific scenarios.

The LT model at first intends to describe the process of shutdowns due to chain effect of energy overload in a power grid. The concept is then adopted for simulating the information diffusion. In the LT model, nodes in a network are the containers of energy (information) and the amounts carried are represented as real values. Each node has a predesignated carrying capacity and initially carries no energy. Once the simulation proceeds, some nodes are assigned as the early adopters, the first groups gaining energy (information), to carry energy, and the carried amount increases progressively. Once the amount of carried energy exceeds their capacity, the nodes become active (overload) and pass excessive energy to other linked nodes. This leads to a propagation of power overload.

With an operation similar to the LT model, the IC model further simplifies the carried information as a binary value. Nodes become active once they receive the information passed from the linked neighbors. There is no predetermined capacity for nodes in IC model. Instead, each edge is associated with a real number representing the probability of successful information pass along it. During the diffusion, the active nodes continually attempt to send out the information through edges until all linked neighbors become active.

Kempe et al. (2003) [3] generalized the IC model by introducing a General Cascade Model. Gruhl et al. (2004) [5] and Leskovec et al. (2006) [6] proposed generative model to simulate blog essay generation based on the IC Model. These models assume nodes can turn from inactive to active given a certain probability for cascading. Based on the LT model and the IC model, Saito et al. (2010) [7] proposed the Asynchronous Linear Threshold Model and Asynchronous Independent Cascade Model.

Another influential line of research, following the success of the PageRank algorithm, puts the propagation process in an explicit recursive mathematical form. Heat diffusion [8, 9] is a physics phenomenon describing heat flows from high temperature positions to low temperature positions. Inspired by the heat diffusion, Ma et al. (2008) [8] proposed a model to

analyze candidate selection strategies for market promotion. The process is formulated as

$$\frac{f_i(t+\Delta t)-f_i(t)}{\Delta t}=\alpha\sum_{j:(v_j,v_i)\in E}\left(f_j(t)-f_i(t)\right), \quad (1)$$

where $f_i(t)$ is the heat of node i at time t and α is the thermal conductivity, namely, the heat diffusion coefficient.

In heat diffusion process, each vertex receives heat from its neighbors, which is similar to our model. The major difference, which will be discussed in detail in the following section, is that heat diffusion model lacks a normalization phase (since it considers only the propagation of one value) and a fusion phase (because the heat itself can disappear after diffusion, so there is no need to fuse on heat diffusion model).

Inspired by these previous works, our model takes the strong points of these approaches, namely, their focus on mimicking social interaction traits such as forming consensus as well as their incorporation of structural information into the propagation process, and blends them into a more coherent framework that could be used to answer real world problems mentioned in the introduction.

2.2. Electoral Issues. In 1992, Bartholdi III et al. [10] first studied the complexity of the process to determine needed actions by organizer to add or remove candidates to manipulate election results (where it is recognized as the classical social choice theory). However, they did not propose any model for the interactions between voters. Gibbard [11] and Satterthwaite [12] showed that every election scheme with at least three possible outcomes is subject to individual manipulation. This means the minority has a chance to manipulate the group decision to secure a preferred outcome. Gibbard and Satterthwaite also addressed the computational difficulty in minority manipulation. However, their model assumes the independence of voters, which does not consider nor discuss the effect from other voters on voter's preference. Existing studies in this direction mainly focus on the complexity and feasibility issues, which is very different from our goal.

Liu (2009) [13] attempted to check whether the preference distribution changes if the number of political experts in a communication network increases. They use an agent-based model for simulation. Each agent in the model maintains a binary value toward a candidate (instead of a real value or ranking) and simply disseminates the values to other agents in the nearby 3 by 3 matrix.

Yoo et al. (2009) [14] proposed semisupervised importance propagation model. Their idea is, to some extent, similar to our "fusion phase" by adding the original score into the accumulated score obtained from the neighbor. The difference between their model and ours is that theirs deals with a single value instead of a vector, and therefore they do not perform the normalization over candidate scores like we do.

The election manipulation is a long-discussed issue. Nevertheless, the social factor is absent in these works. Here we bring a marriage between the social network analysis and the electoral issues.

3. The Proposed Model

We here propose the diffusion model to unearth how the communications affect the individual decisions. Abbreviations section lists the notations used in the rest of the paper.

3.1. Preference Propagation Model. We first define a preference profile p_v of an individual v, which is a k-dimensional vector that represents v's preference toward k different candidates. The jth element in p_v is an integer in $[1,k]$ indicating this individual's preference for candidate j (smaller numbers denote higher ranks). To facilitate the operation of the preference profiles, we translate p_v into a score vector s_v, for all v, using the following equation:

$$s_v[i]=\frac{(k-p_v[i]+1)}{T}, \quad \forall i\in 1,2,\ldots,k, \quad (2)$$

where $T=k(k+1)/2$. This transformation can be regarded as a normalization process as in s_v not only does the preferred candidate receive higher score but also the sum of all elements equals 1. Using the score vector of each individual, we can create an n by k matrix $S=(s_{v_1},s_{v_1},\ldots,s_{v_n})^t$ denoted by the preference matrix. We denote the preference matrix of a given time stamp t since the propagation process starts as $S(t)$.

The information propagates one iteration after the other in our model, and each iteration consists of three phases: *propagation*, *normalization*, and *fusion*.

In the propagation phase, each node v synchronically propagates the preference score vector s_v to the neighboring nodes. To describe such operation mathematically, we define an $n\times n$ forward transition matrix F such that the multiplication of F and $S(t)$ represents the score of each node obtained from all neighbors after this phase. We denoted it by $S_p(t)$.

We assume the edge directions in a network G reveal the direction of influence. Therefore, $F=(KA)^t$, where K is a diagonal matrix with the inverse of degree of each node in the diagonal and A is the adjacency matrix of G. Note that F is identical to the forwarding matrix of a random walk algorithm. The only difference is that F in a random walk algorithm is multiplied by a vector instead of by a matrix S.

In S_p, each row represents the neighbors' accumulated preference scores toward each candidate. Unlike S, the elements in each row of S_p do not add up to one. To ensure that every individual has equal influence while casting votes, we normalize each row of S_p so that its elements add up to one. Therefore, in the second phase, S_p is multiplied by an $n\times n$ diagonal normalization matrix N, where each element in the diagonal of N is equal to the sum of all elements in the corresponding row of S_p. After the second phase, we will obtain a new scoring matrix $S_n(t)=NFS(t)$.

The major difference between our propagation model and the diffusion models for electricity/heat (see Section 2 for more details) lies in the intrinsic difference of the media that are propagated. Electricity or heat flows from one place to another (that is, a flow from node A to node B implies the material does not exist in A anymore). Opinions, by contrast, do not vanish after propagation (that is, A's inclination towards a candidate does not disappear even after bringing

his or her opinions to B). Therefore we add a third phase to include a fusion model that integrates an individual's own opinions $S(t)$ with the opinion $S_n(t)$ gathered from its neighbors.

In the fusion phase, we introduce a parameter for each individual: the susceptible ratio, a real number $\epsilon \in [0, 1]$ that represents how easily an individual can be affected by others. Given a susceptibility parameter for each individual, we can then create a susceptible matrix E, an $n \times n$ diagonal matrix with the ϵ value of each individual in the diagonal. If E is equal to the identity matrix I, which would imply all individuals are equally and highly susceptible to one another, then $S(t + 1)$ should be equivalent to its neighbors' opinion $S_n(t)$. On the opposite side, if E is equal to the zero matrix, implying all individuals are impervious to one another, then $S(t+1)$ should be identical to $S(t)$. Thus, after one iteration of propagation, the preference score matrix can be represented as

$$S(t+1) = (I - E)S(t) + ENFS(t) = ((I - E) + ENF)S(t). \tag{3}$$

Note that we assume that E does not change over time, and neither does F (which is only dependent upon topology). Interestingly, at first glance one might assume that N changes iteratively; it actually does not. Because the sum of each column in F equals 1 and the scores are always normalized for all candidates, it is not hard to prove that

$$N_{ij} = \begin{cases} \left(\sum_{j=1}^{n} F_{i,j}\right)^{-1} & \text{when } i = j \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

which depends only on F. Therefore, we can rewrite $S(t + 1)$ as $\mathcal{X}S(t)$, where $\mathcal{X}$ is a time-independent matrix, which becomes an important feature for the proof of convergence in the next section.

The above concludes one iteration of propagation. In the next iteration, $S(t + 1)$ becomes the initial preference score for the individuals and the same process can be executed to obtain another round of propagation results $S(t + 2)$. Algorithm 1 is the algorithm for our model.

3.2. Proof of Convergence and Consensus. In this section, we show the convergent property of our proposed scheme. The score matrix becomes invariant after a sufficient number of propagations. Moreover, we show that given certain conditions all rows in the converged score matrix are identical. In other words, a consensus within a community will eventually be reached through information propagations in our model.

Let $\mathcal{X}$ denote the overall preference propagation operation of all three phases explicitly laid down in the previous section,

$$S(t+1) = \mathcal{X}S(t) = [(I - E) + ENF]S(t). \tag{5}$$

To provide intuition for the forthcoming deductions and to borrow results of the properties of $\mathcal{X}$ from Section 3.1, we start by pointing out the similarities as well as the differences between $\mathcal{X}$ and the PageRank matrix $\mathcal{G}$. First, the entity $\mathcal{X}$ acting on $S(t)$ is actually a matrix consisting of the vectors of probabilities instead of a simple vector of probabilities. As a result, the columns of $\mathcal{X}$ do not add up to 1 (only the rows do) and therefore it is not a stochastic matrix. Furthermore, a social personal relationship network is intrinsically more localized compared to the World Wide Web, and, as such, the favorable positive definite property enjoyed by $\mathcal{G}$ does not necessarily hold for $\mathcal{S}$. That said, these complexities, while no doubt complicating the theoretical treatment of our algorithm, are in fact a natural manifestation of the increased richness of our target of research in hand—social networks.

We start our deduction of the convergence of $\mathcal{X}$ by enlisting the Perron-Frobenius theorem [15] which states that an irreducible, acyclic matrix has a single eigenvalue that is strictly larger than the others. Under the assumption that the graph being induced by $\mathcal{X}$, $G_{\mathcal{X}}$, is strongly connected and that the weights matrix E has entries smaller than one but not all zeros, $\mathcal{X}$ is irreducible and acyclic and thus applies to the Perron-Frobenius theorem. We denote the dominant real positive eigenvalue of $\mathcal{X}$ by r. Armed with this fact, we are able to transform $\mathcal{X}$ into its Jordan canonical form

$$\mathcal{X} = P^{-1} J_{\mathcal{X}} P, \quad J_{\mathcal{X}} = \begin{pmatrix} J_{\mathcal{X}_1} & 0 & \cdots \\ 0 & J_{\mathcal{X}_2} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}, \tag{6}$$

by which the leading block $J_{\mathcal{X}_1}$ is a 1×1 matrix $[r]$, and other $J_{\mathcal{X}_i}$'s correspond to their strictly smaller eigenvalues $\lambda_{\mathcal{X}_i}$. Since, by the rules of matrix multiplication, the effects of $\mathcal{X}$ on $S(t)$ can be analyzed one by one with respect to $S(t)$'s column vectors without loss of generality, we will proceed with our proof of $S(t)$'s convergence by concentrating on $S(t)$'s column vectors which we denote by lower case $s(t)$. Decomposing $s(0)$ into the sum of $\mathcal{X}$'s eigenvectors, $c_1 v_1 + c_2 v_2 + \cdots$, we obtain the general form of the time evolution of $s(t)$,

$$s(t) = J_{\mathcal{X}}^{t} (c_1 v_1 + c_2 v_2 + \cdots) = r^t (c_1 v_1 + b_t), \tag{7}$$

where

$$\begin{aligned} \|b_t\| &= \frac{1}{r^t} \left\| J_{\mathcal{X}_2}^{t} c_2 v_2 + \cdots \right\| \\ &\le \sum_{i=2}^{|V|} \left(\frac{|\lambda x_i|}{r} \right)^t \|c_i v_i\| \longrightarrow 0, \quad \text{as } t \longrightarrow \infty. \end{aligned} \tag{8}$$

The above shows that $\|b_t\|$ converges to zero when t is large, and therefore $S(t)$ converges to $r^t(c_1 v_1)$. To get an intuition for the speed of this convergence, we turn to a special case where the susceptible ratios are identical; that is, E is a scalar ϵ. In this case, we apply the Perron-Frobenius theorem again on NF and we again obtain NF's Jordan form

$$NF = P^{-1} J_{NF} P, \quad J_{NF} = \begin{pmatrix} J_{NF_1} & 0 & \cdots \\ 0 & J_{NF_2} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}. \tag{9}$$

However, note that since it needs not to be acyclic, it is strictly larger than the other. Now, using this basis we find that $\mathcal{X}$ is equal to

R: iteration number; P: initial preference profiles
E: susceptible matrix; F: forwarding matrix
N: normalization matrix
$S(0) = PreferenceToScore(P)$
for $t = 0$ **to** R **do**
  $S_p(t) = FS(t)$
  $S_n(t) = NS_p(t)$
  $S(t+1) = ES(t) + (I - E)S_n(t)$
end for
return $S(R)$

Algorithm 1: Preference propagation model.

$$\epsilon \begin{pmatrix} \ddots & \ddots & & 0 \\ & \dfrac{\left(1-\epsilon+\epsilon\lambda_{NF_i}\right)}{\epsilon} & 1 & \\ & & \dfrac{\left(1-\epsilon+\epsilon\lambda_{NF_i}\right)}{\epsilon} & 1 \\ 0 & & & \dfrac{\left(1-\epsilon+\epsilon\lambda_{NF_i}\right)}{\epsilon} \quad \ddots \\ & & & \ddots \end{pmatrix}. \tag{10}$$

Since a Jordan canonical form is unique, we obtain $\lambda_{\mathcal{X}_i} = (1 - \epsilon + \epsilon\lambda_{NF_i})/\epsilon$. From this result, we confirm that when $\epsilon = 0$, $\mathcal{X}$ degenerates to the trivial diagonal case and that as ϵ approaches 1, the rate of convergence is geometrically proportional to ϵ/r.

We are now one step away from the final proof of S's convergence. Recalling that $s(t) \to r^t c_1 v_1$, once $r \leq 1$ is established, $S(t)$ converges. To prove this, we take advantage of the Collatz-Wielandt theorem which gives the following formula for r: $r = \max_{x \in N} f(x)$, where $f(x) = \min_{1 \leq i \leq n; x_i \neq 0} [\mathcal{X}x]/x_i$ and $N = \{x \mid x \geq 0 \text{ with } x \neq 0\}$.

We begin by asserting that the upper bound of $f(x)$ is 1. To prove this, we suppose the opposite holds, that means there exists x such that $f(x) = \min_{1 \leq i \leq n; x_i \neq 0} [\mathcal{X}x]/x_i = \alpha > 1$. This implies the following list of equations:

$$\begin{aligned} 1 < \alpha &\leq \frac{1}{x_1}\left(\mathcal{X}_{11}x_1 + \mathcal{X}_{12}x_2 + \cdots + \mathcal{X}_{1n}x_n\right) \\ &\vdots \\ 1 < \alpha &\leq \frac{1}{x_n}\left(\mathcal{X}_{n1}x_1 + \mathcal{X}_{n2}x_2 + \cdots + \mathcal{X}_{nn}x_n\right). \end{aligned} \tag{11}$$

Note that $\sum_{j=1}^{n} \mathcal{X}_{ij} = 1, \forall i$. Thus, the above list of equations can be arranged into

$$\begin{aligned} &\mathcal{X}_{12}\left(\frac{x_2}{x_1} - 1\right) + \mathcal{X}_{13}\left(\frac{x_3}{x_1} - 1\right) + \cdots + \mathcal{X}_{1n}\left(\frac{x_n}{x_1} - 1\right) \\ &> 0 \\ &\vdots \\ &\mathcal{X}_{n1}\left(\frac{x_1}{x_n} - 1\right) + \mathcal{X}_{n2}\left(\frac{x_2}{x_n} - 1\right) + \cdots + \mathcal{X}_{n(n-1)}\left(\frac{x_{n-1}}{x_n} - 1\right) \\ &> 0. \end{aligned} \tag{12}$$

However, by denoting i by the subscript that has $x_i = \max_{1 \leq j \leq n} x_j$ and remembering that $\mathcal{X}$ is a nonnegative matrix, one of the above equations would not hold:

$$\mathcal{X}_{i1}\left(\frac{x_1}{x_i} - 1\right) + \mathcal{X}_{i1}\left(\frac{x_2}{x_i} - 1\right) + \cdots + \mathcal{X}_{in}\left(\frac{x_n}{x_i} - 1\right) > 0. \tag{13}$$

This justifies the assertion that $f(x) \leq 1$. Combining this result with the observation that the trivial vector $(1, 1, \ldots)$ yields $f(x) = 1$, we conclude that $\max_{x \in N} f(x) = 1$. Therefore, $r = 1$, and $S(t)$ converges to $c_1 v_1$.

For networks that are not strongly connected we can always find the SCCs in linear time, and the problem reduces to the smaller "source SCCs" of the network since the matrices of all the other SCCs have a Perron root smaller than 1 and their elements eventually vanish. For the remaining source SCCs, since no vertices have susceptibility ratios equal to 1, according to the above results they all converge. The net effect is exemplified by the stark difference between the individuals belonging to the various source SCCs and the rest. Whereas source SCC vertices will converge to their own respective common values, the others may converge to different values

and act as followers in terms of aligning their own preferences to the weighted average of those belonging to the sources. Figure 1 gives an example of such phenomenon. Let the initial preference matrix of all the nodes in Figure 1 be

$$\begin{pmatrix} s_A & s_B & s_C & s_D & s_E \\ s'_A & s'_B & s'_C & s'_D & s'_E \\ s''_A & s''_B & s''_C & s''_D & s''_E \end{pmatrix} \begin{matrix} \text{candidate1} \\ \text{candidate2} \\ \text{candidate3,} \end{matrix} \quad (14)$$

where each row in the preference matrix denotes each node's preference for candidates 1, 2, and 3, respectively. Then after infinite number of propagations, the preference matrix will become

$$\begin{pmatrix} s(\infty)_{AB} & s(\infty)_{AB} & s(\infty)_C & s(\infty)_D & s(\infty)_E \\ s(\infty)'_{AB} & s(\infty)'_{AB} & s(\infty)'_C & s(\infty)'_D & s(\infty)'_E \\ s(\infty)''_{AB} & s(\infty)''_{AB} & s(\infty)''_C & s(\infty)''_D & s(\infty)''_E \end{pmatrix}, \quad (15)$$

in which the preferences of nodes A and B in Figure 1 for candidate 1 converge to the common value $s(\infty)_{AB}$, for candidate 2 converge to the common value $s(\infty)'_{AB}$, and for candidate 3 converge to the common value $s(\infty)''_{AB}$. However, for nodes D and E, given that the SCC composed by them {D, E} is under the influences of both opinion leaders SCC {A, B} and {C}, their eventual preferences instead of aligning themselves to a common value become a linear combination of the preferences of {A, B} and {C}. The exact details of this combination depend on the structure of the network.

The preference propagation model simulates this unique behavior of people by projecting the preferences' vector onto the leading uniform eigenvector denoting equilibrium. In addition, it also attempts to mimic the real world by distinguishing the opinion leaders from the followers. As with its real world counterpart, this process is solely determined by the initial preferences of every individual and the structure of the embedding social network.

Another example is shown in Figure 2, time evolution of preferences held by nodes in a social network demonstrating the effects of opinion leaders, creating their own consensus, and passing it down to opinion followers in a cascading manner. We see that the opinion follower SCC composed by nodes 15 to 20 is colored with various shades of gray depending on its distance to the two opinion leader SCCs composed by nodes 1 to 3 and 4 to 7. We observe that the preference of the opinion leader SCCs 1 to 3 is first passed to the opinion follower SCCs 8 to 11 (in 10th propagation round) and then subsequently passed to the opinion follower SCCs 12 to 14 through the efforts of SCCs 8 to 11 in a cascaded manner.

This simple example demonstrates that the strongly connected source components form the opinion leader groups, while each follower node is affected by (i.e., linear combination) the opinions of its surrounding opinion leader groups. Our framework models the real world observation about how less-convinced personnel are affected by the mass opinions they encountered.

3.3. Comparison with Other Models. We here discuss what the most salient characteristics of a successful social model are based on common observations and beliefs, in an attempt to contrast the most distinguishing features of our model with the other previously proposed frameworks.

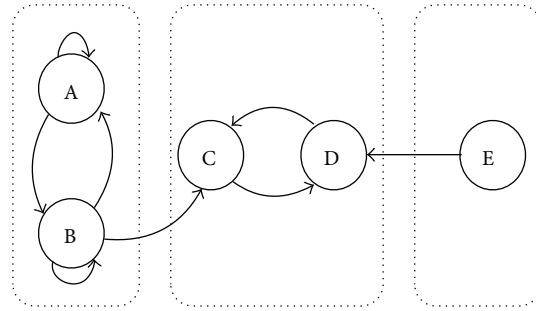


Figure 1: Nodes A, B form an opinion leader SCC, while node C by itself is another opinion leader SCC. Nodes E and D form an opinion follower SCC.

High Dimension Media. Since a personal preference describes the order of preference of all possible candidates, the media in an ideal model should be represented as ordered lists instead of as a single value. Most of the propagation models such as Linear Threshold Model, Cascade Independent Model, or DiffusionRank model, unfortunately, only handle binary or real value in propagation.

Topology Dependence and Input Dependence. The word of mouth is the main strategy for a person to affect others. The real world process of guiding friends toward the adoption of self-preference goes mutually and simultaneously. To state such phenomenon, the outgoing persuasions of a person should ideally become a combination of self-preference and the incoming preferences. An ideal model should take into account both network structure and initial personal preference. Moreover, we would like a model's way of incorporating these two factors to be as natural as possible, instead of relying on ad hoc stopping designs or simply restricting the number of time nodes or individuals' interactions.

Deterministic Convergence. Of course an ideal model should converge or end eventually, or else it would be difficult for the modeler to interpret the results. As far as we know, there are currently two kinds of designs to achieve such a convergence. The first one, such as LT model and IC model, attaches a binary status to each node in a network to determine whether it is visited. The *inactive* status means the node is not yet visited while the *active* status means the node is visited. With such design, preference propagation to inactive nodes can be easily monitored. Moreover, the propagation converges in such model when none of the existing nodes can change the status anymore.

Following the success of the PageRank algorithm, the second popular approach is building the convergence mechanism into a model inherently, so that after sufficient iterations the model converges and produces a definite result.

To make results easily analyzable, convergent models that can generate identical results, given both the same initial preferences of nodes and network structure, are preferred.

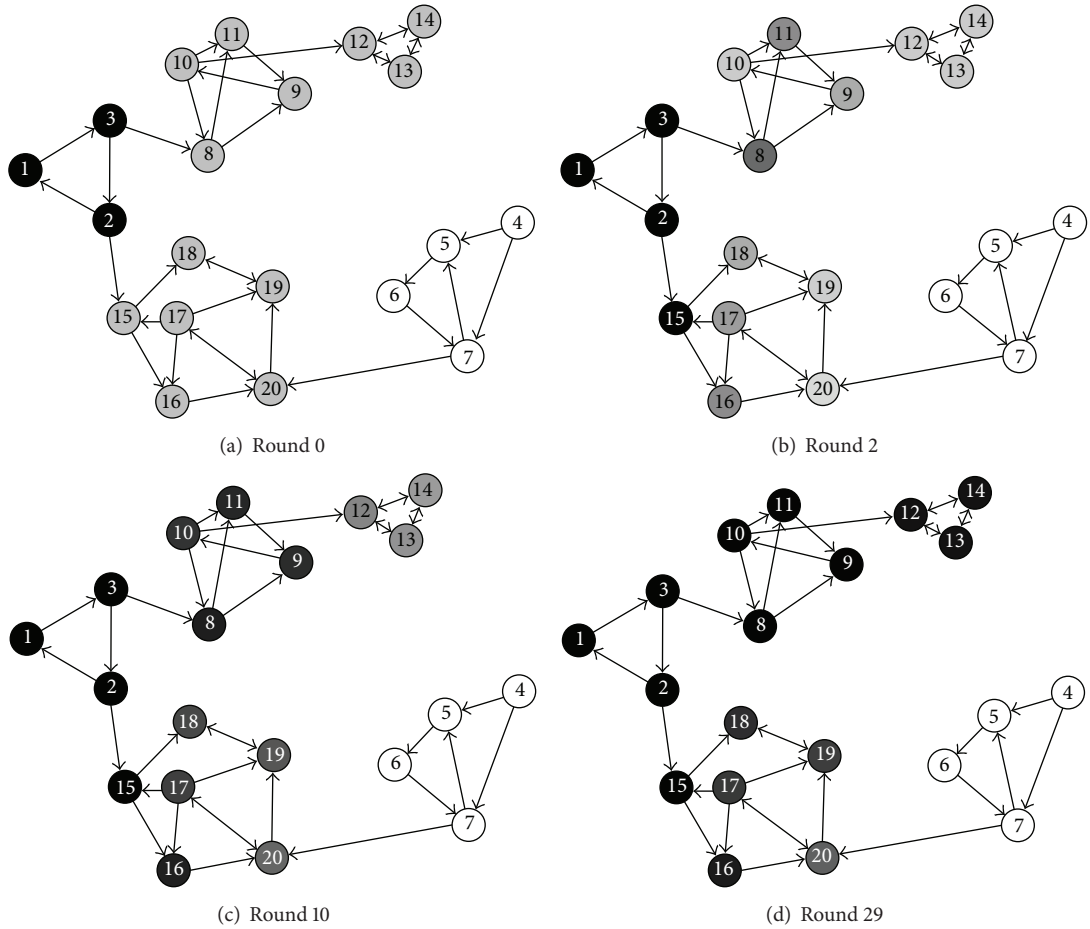


Figure 2: Time evolution of preferences held by nodes in a social network, demonstrating the effects of opinion leaders creating their own consensus and passing it down to opinion followers in a cascading manner.

Consensus. The problem of reaching a consensus among agents has been studied since around 1970 [16, 17] with simulation models such as the voter model [1]. Mossel et al. gave a theoretical proof that the consensus could be reached with the voter model [18]. Thus an ideal model should be able to reflect specific common traits. In particular, we observe that one such universal trait is people in the same community (i.e., SCC) having the tendency to align their preference after sufficient exchanges. This translates into the fact that an ideal model should contain some kind of homogeneity inside a group.

To see how our model and other proposed frameworks capture the above characteristics of real world social interactions, we conducted several experiments and recorded their results in Table 1 for ease of comparison. We particularly chose models that are most representative in their own stance, namely, the Linear Threshold Model, Independent Cascade Model, PageRank model, and DiffusionRank model, for comparison. Note that since the propagating media in these models are not a vector of preference, we made the following enhancements for each of them to handle such cases. For the LT and IC models, we assume that each vertex initially held approval for its top k preferred candidates (nonapproval for the others), and thus, for every candidate, we can obtain a list of seeds as inputs into the LT and IC models. We then execute the model separately on each candidate, gather their results, and normalize them to form the final preference of each vertex. For the PageRank and DiffusionRank models, given that they can take real values as inputs, we simply executed these models separately for each candidate in the preference list and then integrated the results to be a vector of real numbers.

As shown in Table 1, we see that our model is the only model that operates directly on a list of preferences, whereas other models work restrictively on single boolean or real values and have to be executed separately to obtain a joint preference, which fail to consider the correlation of the preference score among candidates. We note that all models provide convergent results. Besides, since the IC model carries a random component, it does not deliver repeatable final preference results.

To examine whether these models can give a kind of consensus to nodes that belong to a strongly connected network, we execute all models on a strongly connected graph until they naturally stop or converge. It turns out that, except for our model, none showed signs of reaching consensus

Table 1: Comparison of models on the abilities to capture characteristics of real world social networks interactions.

	Convergence	Repeatability of final state	Consensus	Input dependence	Media space
Proposed model	√	√	√ (if SCC)	√	R^k
LT model	√	√		√	Boolean
IC model	√	√		√	Boolean
PageRank	√	√			R
DiffusionRank	√	√		√	R

among the final output preferences. Note that our model does not produce consensus given non-SCC components.

To see whether these models take into account the initial preferences held by nodes, we fed all models with six different initial preferences and see whether they give six different results. It is not surprising that the PageRank model returns identical results regardless of the input, indicating that it takes into account only the structure of the network but ignores the initial preferences held by each node or individual. In conclusion, our model is the only framework that supports all five criteria set by observations from real world social networks.

4. Experiment

In this section, we compare our model with well-known diffusion models to evaluate the performance. We examine whether all the aforementioned algorithms, including ours, can capture the preference transition in social networks to a certain extent. To conduct such validation, we require the information such as the network structure and the node preferences over time.

4.1. Preference Data. The citations of scientific research papers implicitly reveal the research interests of the authors. In other words, we believe that the acts such as citing or submitting to the journals or the conferences would be an indicator of the authors' interests. By utilizing this fact, we can infer the researchers' preference from their corresponding top frequently cited conferences and journals. We have further realized that one author's preference could be influenced by the other coauthors. It is particularly correct for advisor-student relationship since the advisors and students usually affect each other's research interests. We therefore have designed an experiment to model how researchers' preferences can be affected by the collaborators.

We use KDD Cup 2003 ArXiv HEP-TH (High Energy Physics Theory) citation network [19] with the corresponding paper meta information as our evaluation dataset. This dataset contains the citations from 1992 to 2003. We select the top 16 journals that possess most papers as the candidates to construct the preference lists and construct the yearly preference lists for all authors. A preference list consists of the citation count of the corresponding journals within one year. Thus, for each author, we have 12 lists representing their preferences.

The reason we use the citations rather than the publications of authors is that the publications imply not only preference but also capability. To fairly present the interests, we use the citations. In addition, we construct a collaborative network from this dataset as the underlying social preference diffusion backbone. To easily perceive the changes in interests, we remove the authors who had fewer than 5 publications in the dataset, which results in a network with 2683 nodes.

4.2. Model Comparison. Since we already have all the required information including network structure and preference transition, the next step is to study which diffusion model predicts the preference transition better. We assume a good diffusion model could capture the progression of the authors' research interests through collaborations. To do so, we initially set up the node preference according to the actual data in year x and then compare the predicting results with the actual preference in year $x + k$. The following issues are noted in the experiment.

High Dimension Media. To represent the order in preference toward all candidates, the media in an ideal model ought to be an ordered list instead of a single value. Nonetheless, most well-known diffusion models, such as LT, IC, and DiffusionRank, only treat the media as boolean or real number. For comparison, we exploit these models in our problem by executing them independently for each candidate. We evaluate the candidate rank based on each independent diffusion result.

Determinism of the Final State. Except for the IC model, outcome of all the models mentioned above is deterministic. Because the parameter (i.e., diffusion probability) in IC model is a nondeterministic factor, we execute the experiment 20 times and average the results.

Initialization. Because the media in LT and IC models are not native for high dimension, we singly process the propagation for each candidate. That means, in our experiment, the active mode of top 1% authors to a specific publisher is initially set active in LT and IC models, while the rest of publishers are set inactive. We further set the diffusion probability of each edge as $1/N$, where N is the degree of its source node in IC model. In LT model, we assign links with identical weight and nodes with the same threshold. The parameters in LT and IC are then tuned to find the optimal outcome. The propagation process is executed multiple times with different thresholds and the performance is averaged. For DiffusionRank model, we use the parameter settings suggested by the authors of [9].

Table 2: Compare the result after one round for each model with the ground truth of years 1997, 1998, and 1999.

	Kendall's tau			Top 3 Jaccard coefficients		
Year	1997	1998	1999	1997	1998	1999
Independent Cascade	0.007	0.012	0.015	0.011	0.014	0.015
Linear Threshold	0.172	0.167	0.167	0.171	0.195	0.212
DiffusionRank	0.221	0.181	0.160	0.216	0.222	0.213
Proposed(0.00)	0.240	0.204	0.178	0.242	0.243	0.225
Proposed(0.25)	0.243	0.206	0.180	0.248	0.244	0.226
Proposed(0.50)	0.243	0.206	0.180	0.247	0.243	0.227
Proposed(0.75)	0.243	0.206	0.180	0.246	0.243	0.226
Proposed(1.00)	0.230	0.190	0.163	0.204	0.179	0.156

Table 3: Consider the result after $k \times R$ rounds for each model, and compare it with the ground truth of year $1996 + k$. The table shows the average of the similarity scores for 1997, 1998, and 1999.

	Kendall's tau					Top3 Jaccard coefficients				
Round	1	2	3	4	5	1	2	3	4	5
Independent Cascade	0.011	0.011	0.011	0.011	0.011	0.013	0.013	0.013	0.013	0.013
Linear Threshold	0.168	0.168	0.168	0.168	0.168	0.192	0.192	0.192	0.192	0.192
DiffusionRank	0.186	0.186	0.186	0.186	0.186	0.217	0.217	0.217	0.217	0.217
Proposed(0.00)	0.208	0.209	0.207	0.206	0.205	0.238	0.240	0.237	0.236	0.234
Proposed(0.25)	**0.210**	0.209	0.208	0.207	0.207	0.241	0.241	0.240	0.239	0.238
Proposed(0.50)	0.209	0.210	0.209	0.209	0.208	0.240	0.242	0.242	0.241	0.240
Proposed(0.75)	0.209	0.209	0.209	0.209	0.209	0.239	0.241	0.242	**0.242**	0.241
Proposed(1.00)	0.194	0.194	0.194	0.194	0.194	0.179	0.179	0.179	0.179	0.179

4.3. Experiment Result. Diffusion models are evaluated by comparing their predictions about preference in 1997, 1998, and 1999, while using the real preference during the period from 1993 to 1996 as initial status. To measure the similarity between predicting and real results, we adopt Kendall's tau coefficient [20] and the Jaccard coefficient. We individually measure the similarity for each author, each node in the network, and then average them as a performance indicator. Because Kendall's tau coefficient is not well defined with tie scores, we manually set Kendall's tau score as 0 when there is a tie on all 16 publishers. Furthermore, we calculate the Jaccard coefficient performing on the top 3 highest scored publishers.

Firstly, for the sake of knowing the correspondence between the extent of changes in iterations and in years, we execute one-iteration propagation in each model and then compare the results with the ground truth in 1997, 1998, and 1999, respectively. We also try different susceptible ratio ϵ in our model, as $\epsilon = 1.0$ implies the authors stick to their own preferences without considering the effect from the neighbors. Table 2 shows the results, which we find quite suitable to take one iteration as a period of a year. The results demonstrate that our model consistently outperforms the 2nd best model DiffusionRank, regardless of which susceptible ratio is used as long as it is not 1.0.

Secondly, we execute the diffusion algorithms for multiple rounds and compare them with the ground truth of years 1997–1999. Table 3 shows the average of the scores for 1997, 1998, and 1999. Note that LT and IC models stop when there is no possible activation (regarded as one round), which implies that authors are not affected by their neighbors after the first round is completed. Tables 2 and 3 additionally show that the impervious preferences ($\epsilon = 0$) reach a performance similar to the best result, which might reveal the slowly changing nature. Nevertheless, the results show that our model can faithfully capture the trait of the social influence even if the authors' interests change slowly.

5. A Social-Based Simulation Framework for Election Behavior

Based on the preference negotiation model, we implement a simulation framework, shown in Figure 4, granting us to analyze the social impact to elections.

5.1. System Architecture. Voters possess their own preference profiles to each candidate in the early stage of an election. A faithful simulation ought to produce preference profiles that are similar to the practical cases. Thus, we produce profiles satisfying certain distributions according to the data collected from a historical election.

Although we have the voter's preference data, unfortunately, there is no information telling us the relationship between the voters. To deal with this, we propose several plausible scenarios to deploy the profile on a given social network. Because people generally become friends due to similar tastes and thoughts, the profiles should not be distributed randomly. We design several plausible ways to distribute preference profiles on a social network.

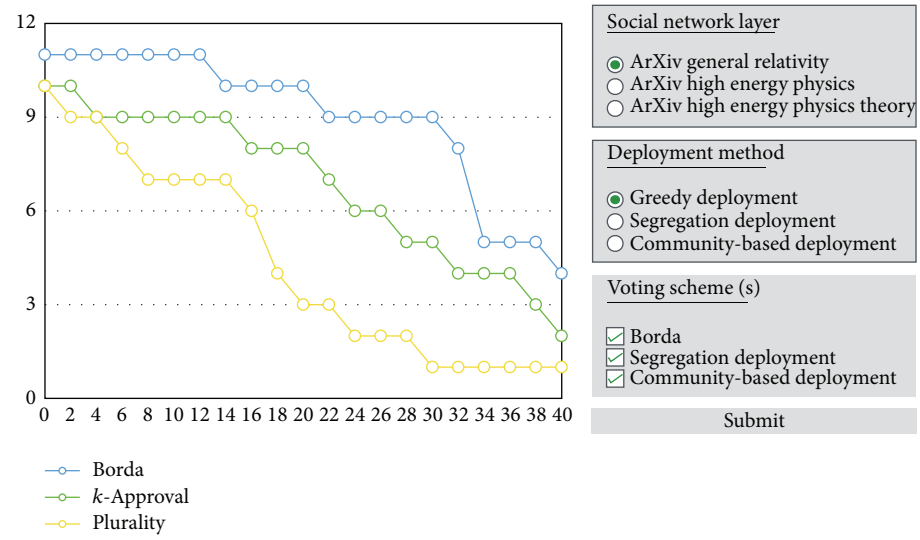


Figure 3: Our user-interface for the convenience of doing experiments.

Once the preference profiles are assigned to each node, the system starts executing the preference negotiation model. Nodes in the network start to persuade others and be persuaded. It is not necessary for a simulation to run until convergence, because in the real world there exist some elections which leave insufficient time for people to negotiate, debate, or exchange opinions before having to cast their votes.

Finally, since in each stage there are several parameters we can adjust, we have designed a user-interface that allows the easy execution for experiments (see Figure 3).

5.2. Preference Profile Generation. In the Preference Profile Generation stage, we create preference profiles based on historical election data obtained from OpenSTV, an online voting record database. We choose to use the "Melbourne City Council Victoria Australia 2008—Lord Mayor Leadership team" data set because it is by far the most complete dataset we have found. This dataset was recorded in November 29, 2008, and has the ballot size of 57,961 and 11 candidates. It consists of the preference lists for all voters. A preference list is a sorted list of candidates revealing the preference order of this particular ballot.

We propose a ranking-preserved sampling method to produce the preference profiles based on the historical data, with the aim of preserving the rank of each candidate. Given our historical data, we first learn a $k \times k$ matrix M_r, where k is equal to the total number of candidates. The (i, j)th elements of M_r encode the probability that the jth position in ranking belongs to candidate i according to the historical dataset, $\text{Prob}(C_i \mid \text{Rank} = j)$. Each column of M_r yields a probability distribution for each candidate of a given rank. Given M_r, we can iteratively sample candidates in each rank (from higher to lower) based on the distribution (with the natural restriction of prohibiting the same candidate in different positions in a single ballot).

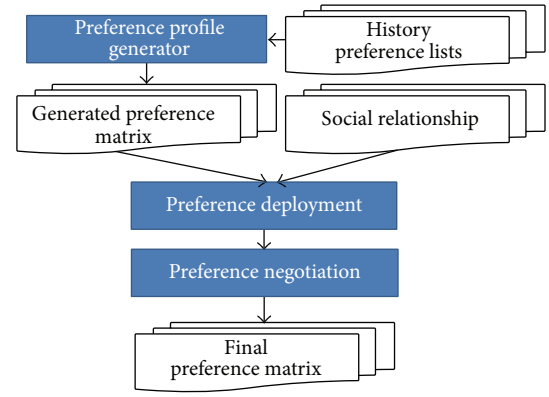


Figure 4: The flow chart of the proposed framework.

5.3. Preference Deployment. As the old saying goes "birds of a feather flock together," we presume that the people of similar preference profiles have a higher likelihood of being close to each other in the network. Below are three algorithms to realize such idea.

Greedy Deployment. It can be realized by first randomly picking a profile and assigning it to a node in the social network; then we assign the most similar unassigned profiles to their neighbors. Then iteratively for each unassigned node, the algorithm allocates it a profile that is the most similar to its neighbors. To measure the similarity, we exploit the commonly used Kendall τ coefficient:

$$\tau = \frac{|\text{concordant pairs}| - |\text{discordant pairs}|}{(1/2)\, n\, (n-1)}. \quad (16)$$

Community-Based Deployment. Our underlying idea is to match social network communities with clusters of preference profiles. We first conduct a community detection

algorithm to identify communities in a social network (in the experiment, we apply [21]), which groups the community and determines the number of communities m automatically based on the maximization of the modularity. Next, we apply a clustering algorithm (in this experiment, we used k-means) to group preference profiles into m clusters based on Kendall τ similarity.

In the final step, we assign profiles to each node. The main idea is to assign each profile in the ith largest cluster to each node in the ith largest community. This, however, is not a straightforward task because the ith community and the corresponding cluster are likely to have different sizes. Here we propose a method to adjust the cluster sizes to match those of the communities. To accomplish this, we first sort both the sets of communities and their clusters by their respective sizes. Then from the largest community to the smallest, we compare the size of the ith community with that of the ith cluster. When the ith community is of the same size as the ith cluster, we randomly assign the profile in the cluster to a node in the community. When the cluster size is smaller, we add the unassigned points outside this cluster but closest to the cluster center into the cluster until its size matches the size of the community. When the cluster is larger, by contrast, we remove the points from the cluster that are farthest away from the center and join them to the closest unassigned neighbor cluster. Doing this iteratively will gradually assign profiles to nodes and guarantee that nodes in the same community have similar profiles. An example is illustrated in Figure 5, and pseudo code is listed in Algorithm 2.

Segregation Deployment. The idea is from the setting of Schelling's segregation model [22]: blacks and whites may not mind, even prefer, each other's presence, but people will move if they are the minority. At the beginning, we deploy profiles randomly. Then, in each iteration, the nodes which have fewer than 30% neighbors with positive Kendall τ similarity values will be selected, and their preference profiles will be shuffled. In this experiment, 1000 iterations are performed.

5.4. Preference Negotiation. Once profiles and communities have been assigned, the framework executes the core preference negotiation model to the network. There are two parameters that can have some impact on the process: the iterations of negotiations R and the susceptible ratio matrix E. R controls the negotiation iterations taken before the voters have to cast the votes. In the experiments below, we set $R = 20$.

We set ϵ as 0.5 for all voters by default for our case studies. As suggested in our proof, if E is a constant matrix c, then the resulting converged preference matrix S is indifferent no matter what c is (c only controls the speed of convergence).

6. Case Studies

In this section, we answer two questions based on the proposed simulation framework.

(1) To what extent does the negotiation process in a social network affect election results?

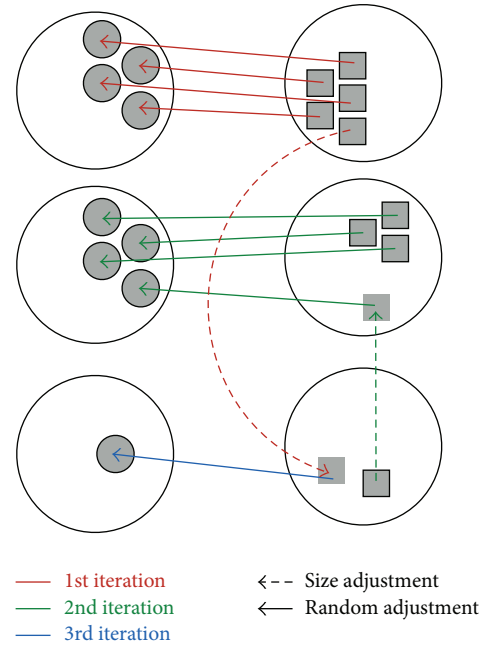


Figure 5: A diagram to demonstrate our deployment algorithm.

(2) Among the widely known voting schemas, Borda Count, k-approval, and plurality, which is the most vulnerable to vote-buying (i.e., easiest to be manipulated)?

In order to construct simulations, we use three collaboration networks (ca-GrQc containing 5,242 nodes and 2,890 edges, ca-HepPh containing 12,008 nodes and 237,010 edges, and ca-HepTh containing 9,877 nodes and 51,971 edges) as the underlying social network dataset. Once the negotiation process ends, the preference scoring vectors will be examined to determine the final rank of the candidates using different voting schemas: Borda Count, k-approval, and plurality [23]. We conduct experiments on all three plausible deployments proposed in Section 5.3. Ideally, we hope the simulation on all three deployment methods can produce similar conclusions, which would consequently offer users higher confidence about the results.

The Borda Count determines the final rank of the candidates by giving each candidate a certain number of points corresponding to the position in which it is ranked by each voter. Once all votes have been counted the candidate can be ranked by their total points. For each ballot, the top-ranked candidate will receive k points, the second $k - 1$ points, and so on.

The ranking in k-approval and plurality schemas is determined similar to Borda Count. The only difference among the three voting schemas is the definition of points to be given to each candidate. In the k-approval voting schema, the top k candidates in each ballot will each receive one point, while the rest will not receive any points. In the plurality voting schema, only the top candidate receives one point.

```
S: social-networks; P: preference-matrix
K: community-number
M[i]: the ith largest community
C[i]: the ith largest cluster
(M[], K) = CommunityDetection(S)
C[] = KmeansForPreference(P, K)
for i = 1 to K do
   if size(C[i]) = size(M[i]) then
      next
   else if size(C[i]) < size(M[i]) then
      add the unassigned preferences which is outside
     C[i] but closest to center(C([i]) into C[i] to
      match size(M[i])
   else
      remove the preferences most away from
      center(C[i]) to the closest unassigned neighbor
      cluster
   end if
   randomly assign the preferences in C[i] to nodes in
   M[i]
end for
return the pairs of (node, preference)
```

Algorithm 2: Preference deployment method.

6.1. Effectiveness of Negotiation. The first question is whether the social-network-based negotiation process can significantly affect the election results. To quantify changes in a voters' preference profile through negotiations, we compute the average Kendall τ coefficient between the preference orders before and after each negotiation.

As shown in Figure 6, no matter which deployment method is used, the average Kendall *tau* coefficient generally decreases as we enter deeper rounds of negotiation. The slope is steepest in the beginning, revealing the fact that the effect of negotiation reaches the peak in the beginning and gradually declines, which matches the real world experience. Eventually the Kendall *tau* value decreases to below 0.5, implying that negotiation through social networks can significantly change the election results.

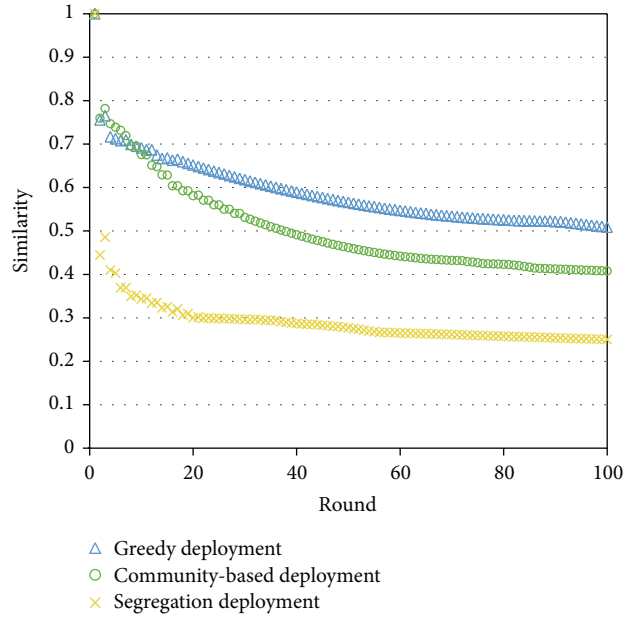
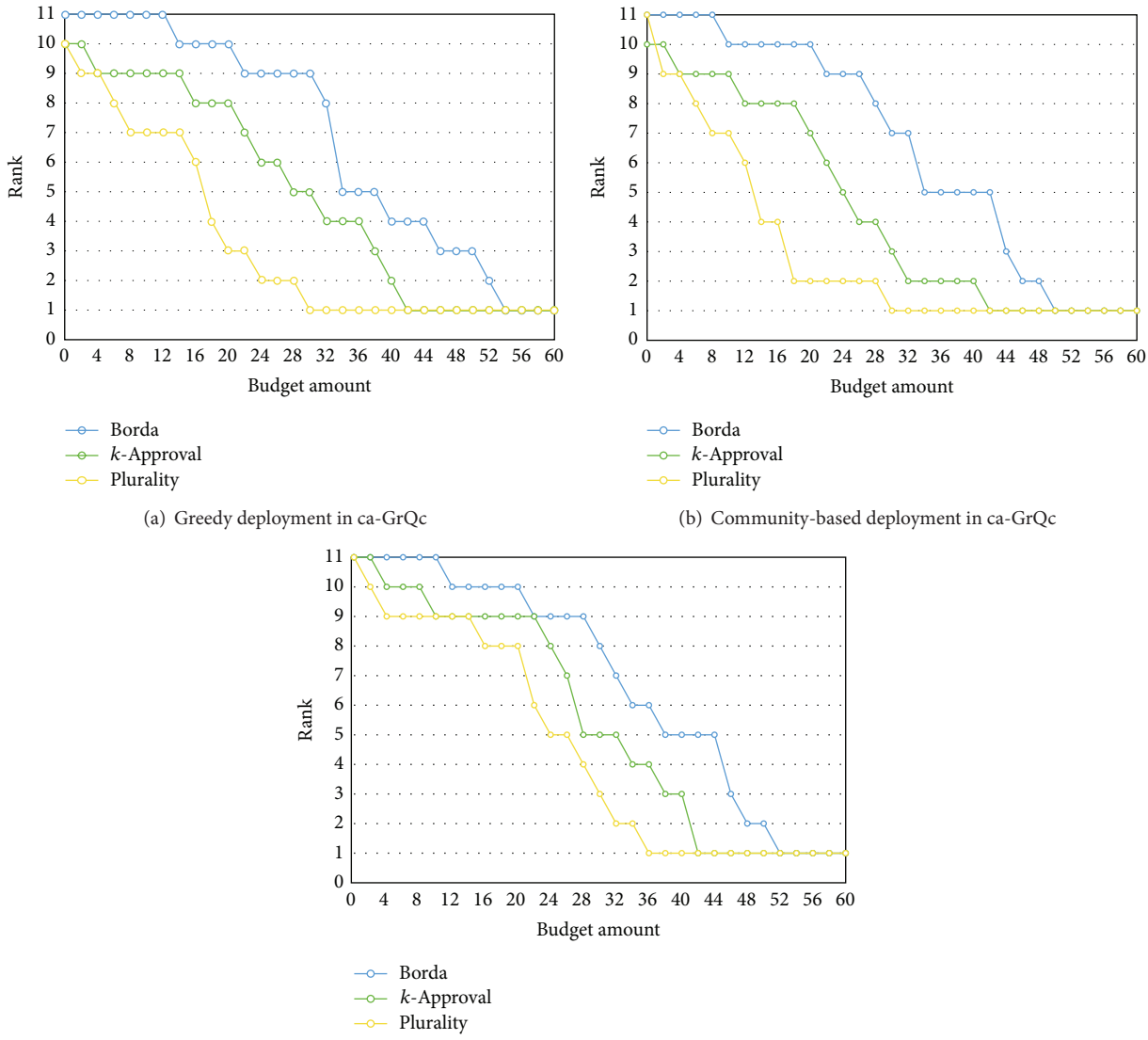


Figure 6: The similarity (average Kendall τ coefficient) between the initial preference matrix and the preference matrix after negotiations.

6.2. Vulnerability to Vote-Buying. This section discusses a key question about elections: if an organization can boost the vote count of a candidate through manipulating certain seed nodes' preference profiles (pejoratively, we can call this "vote-buying"). We define one successful vote-buying to a voter as "raising the target candidate's preference score to slightly higher than the score required to obtain a vote from the voter." For example, if each voter is allowed to cast 3 votes, then the buyers would attempt raising its score to slightly above the 3rd place candidate. To quantify such manipulation cost, we defined it as the difference between the scores before and after manipulating. The scoring vector after vote-buying must be renormalized before further computation can proceed.

Discarding the effect of negotiation, it is intuitive to assume that the best promotion strategy is aiming at the voters whose costs are the smallest. Under such attack, which voting schema is the most vulnerable to vote-buying? This answer can be deduced from Figure 7, where the x-axis stands for the budget spent while the y-axis stands for the rank of a given candidate for promotion. Note that the higher value of y stands for the less favorable, and a candidate has to move to lower position in order to be elected. The



(a) Greedy deployment in ca-GrQc

(b) Community-based deployment in ca-GrQc

(c) Segregation deployment in ca-GrQc

FIGURE 7: The figure shows the rank of promoted candidate in three voting schemas, where the x-axis stands for the budget spent while the y-axis stands for the rank of a given candidate for promotion. The figures in each row show the 3rd-round results with different deployments in networks ca-GrQc. The results in networks ca-HepTh and ca-HepPh are consistent. They are excluded due to the space.

results show that, regardless of the deployment, Borda Count schema consistently requires more budget to advance a target candidate, while the plurality schema is the most vulnerable to vote-buying since it generally requires less budget to advance a candidate to higher rank.

7. Conclusion

Analyzing the effect of social networks upon group decisions outcomes is a difficult problem because it is both costly and time consuming to perform user studies to collect people's private preferences. Indeed, it is the change of preferences through social propagation in particular that we care most about, and to our knowledge this is the first ever study that provides not only theoretical analysis but also the empirical justification of this problem. This study provides an example of how to perform such research with limited data through exploiting algorithm and model design, theoretical justification, and computer simulation.

Our other significant contribution is that we provide an alternative evaluation plan and data to verify a preference propagation model. Acknowledging the lack of real world data to evaluate how the voter's preference can change through social diffusion, we have come up with a novel idea to identify a publicly available bibliography dataset to evaluate how researchers gradually change their research fields according to the influence of their collaborators. Our evaluation plan opens a new possibility that allows

researchers working on preference diffusion problems to be able to evaluate their models without having to identify a highly private voter preference dataset.

Abbreviations

V: Individuals
C: Candidates
n: Number of individuals
k: Number of candidates
p_v: Preference profile vector of individual $v \in V$
s_v: Preference score vector for individual $v \in V$
S: Preference scoring matrix with size $|V| \times |C|$
G: Social network layer.

Conflict of Interests

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

Acknowledgments

This work was supported by the National Science Council, National Taiwan University, and Intel Corporation under Grants NSC101-2911-I-002-001, NSC101-2628-E-002-028-MY2, and NTU102R7501.

References

[1] T. Liggett, *Interacting Particle Systems*, Grundlehren der Mathematischen Wissenschaften, Springer, 1985.

[2] M. Granovetter, "Threshold models of collective be- havior," *American Journal of Sociology*, vol. 83, no. 6, pp. 1420–1443, 1978.

[3] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '03)*, pp. 137–146, August 2003.

[4] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '09*, pp. 199–207, ACM, July 2009.

[5] 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, ACM, May 2004.

[6] J. Leskovec, M. Mcglohon, C. Faloutsos, N. Glance, and M. Hurst, "Cascading behavior in large blog graphs: patterns and a model," Tech. Rep., 2006.

[7] K. Saito, M. Kimura, K. Ohara, and H. Motoda, "Selecting information di usion models over social networks for behavioral analysis," *Machine Learning and Knowledge Discovery in Databases*, vol. 6323 of *Lecture Notes in Computer Science*, 2010.

[8] H. Ma, H. Yang, M. R. Lyu, and I. King, "Mining social networks using heat diffusion processes for marketing candidates selection," in *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM '08)*, pp. 233–242, October 2008.

[9] H. Yang, I. King, and M. R. Lyu, "DiffusionRank: a possible penicillin for web spamming," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*, pp. 431–438, July 2007.

[10] J. J. Bartholdi III, C. A. Tovey, and M. A. Trick, "How hard is it to control an election?" *Mathematical and Computer Modelling*, vol. 16, no. 8-9, pp. 27–40, 1992.

[11] A. Gibbard, "Manipulation of voting schemes: a general result," *Econometrica*, vol. 41, no. 4, pp. 587–601, 1973.

[12] M. A. Satterthwaite, "Strategy-proofness and Arrow's conditions: existence and correspondence theorems for voting procedures and social welfare functions," *Journal of Economic Theory*, vol. 10, no. 2, pp. 187–217, 1975.

[13] F. C. Liu, "Modeling political individuals using the agent-based approach: a preliminary case study on po- litical experts and their limited in uence within com- munication networks," *Journal of Computers*, vol. 19, no. 4, pp. 8–19, 2009.

[14] S. Yoo, Y. Yang, F. Lin, and C. Moon II, "Mining social networks for personalized email prioritization," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '09)*, pp. 967–975, July 2009.

[15] C. D. Meyer, Ed., *Matrix Analysis and Applied Linear Algebra*, Society for Industrial and Applied Mathematics, Philadelphia, Pa, USA, 2000.

[16] M. H. DeGroot, "Reaching a consensus," *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 118–121, 1974.

[17] R. L. Winkler, "The consensus of subjective probability distributions," *Management Science*, vol. 15, no. 2, pp. B61–B75, 1968.

[18] E. Mossel and G. Schoenebeck, "Reaching consensus on social networks," in *Proceedings of the International Conference on Supercomputing (ICS '10)*, pp. 214–229, 2010.

[19] J. Gehrke, P. Ginsparg, and J. Kleinberg, "Overview of the 2003 KDD Cup," *SIGKDD Explorations Newsletter*, vol. 5, no. 2, pp. 149–151, 2003.

[20] M. G. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, no. 1-2, 1938.

[21] 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.

[22] T. Schelling, *Models of Segregation*, The American Economic Review, 1969.

[23] P. E. Johnson, "Voting systems," http://pj.freefaculty.org/Ukraine/PJ3_VotingSystemsEssay.pdf.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 248041, 13 pages
http://dx.doi.org/10.1155/2014/248041

Research Article

Distributed Leader-Following Finite-Time Consensus Control for Linear Multiagent Systems under Switching Topology

Xiaole Xu,[1,2] Shengyong Chen,[2] and Lixin Gao[3]

[1] *Wenzhou Vocational College of Science & Technology, Zhejiang 325006, China*
[2] *College of Information Engineering, Zhejiang University of Technology, Zhejiang 310023, China*
[3] *Institute of Intelligent Systems and Decision, Wenzhou University, Zhejiang 325027, China*

Correspondence should be addressed to Lixin Gao; gao-lixin@163.com

Received 18 January 2014; Accepted 24 March 2014; Published 24 April 2014

Academic Editors: H. R. Karimi and X. Yang

This paper investigates the finite-time consensus problem of leader-following multiagent systems. The dynamical models for all following agents and the leader are assumed the same general form of linear system, and the interconnection topology among the agents is assumed to be switching and undirected. We mostly consider the continuous-time case. By assuming that the states of neighbouring agents are known to each agent, a sufficient condition is established for finite-time consensus via a neighbor-based state feedback protocol. While the states of neighbouring agents cannot be available and only the outputs of neighbouring agents can be accessed, the distributed observer-based consensus protocol is proposed for each following agent. A sufficient condition is provided in terms of linear matrix inequalities to design the observer-based consensus protocol, which makes the multiagent systems achieve finite-time consensus under switching topologies. Then, we discuss the counterparts for discrete-time case. Finally, we provide an illustrative example to show the effectiveness of the design approach.

1. Introduction

Cooperative control of multiagent systems (MAS) has received increasing attention over the last ten years with rather diverse background such as biology, physics, mathematics, information science, computer science, and control science. Many topics such as swarm, aggregation, formation, schooling, and synchronization are involved in a critical problem known as the consensus problem [1–5]. The objective of consensus for multiagent systems is to design the distributed protocols based on the local relative information so that the states of a team of agents can reach an agreement [1].

The consensus problems have a long history in the field of computer science. In [1], Jadbabaie et al. studied the consensus protocols motivated by biological group behaviors, which stirred the excitement of the research on distributed cooperative control in the control community. In most existing works on consensus, the agent dynamics are restricted to first-, second- and, sometimes, high-order integrators [1, 3, 6–12]. In [7], Ren and Atkins showed that in sharp contrast to the first-order consensus problem, consensus for a group of agents with second-order dynamics many fail to be achieved even if the network topology has a directed spanning tree. Recently, the consensus problem with a general linear dynamical agent has been probed by [13–19]. The interacting topology of multiagent systems is a key factor to achieve consensus. For fixed topology, the eigenvalue decomposition method can be used to solve the multiagent consensus problem [10, 14, 16]. For multiagent systems with high-order dynamics under switching interacting topology, the common Lyapunov function method is involved to analyze consensus problem of multiagent systems [6, 9, 17, 18].

Since some state variables cannot be obtained directly in many practical systems, the state observer is involved in proposed control law to achieve control aim. Till now, the observer-based design technique became an important control approach. Much of the attention has been devoted to achieving state consensus for a network of identical agents, where each agent has access to a linear combination of

its own states relative to those of neighboring agents [1, 7, 10, 13]. In many practical systems, the agent cannot obtain full state information but only obtain output information of its neighbors. Usually, observer-based approach is proposed for agent to solve the state consensus problem. Distributed estimation via observers design for multiagent coordination is an important topic with wide applications especially in sensor networks and robot networks. To track the active leader, the tracking protocols based on state observers were proposed for the first-order and second-order agents [6, 9]. The observer-based protocols were provided to solve multiagent consensus problem with general linear dynamics in [14–19]. The leader-following configuration is very useful to design the multiagent systems, which has been discussed in [1, 6, 9, 13, 16–19].

Most of the existing control techniques related to the stability focus on Lyapunov asymptotic stability, which is defined over an infinite-time interval. However, in some practical applications, we mainly concern the behaviors of the system over a fixed finite-time interval, such as convergence to an equilibrium state in finite time. For these cases, the finite-time stability (FTS) is involved. Finite-time convergence to a Lyapunov stable equilibrium was investigated in [20]. Finite-time stabilization for a chain of power-integrator systems was considered in [21, 22]. A general framework for finite-time stability analysis based on vector Lyapunov functions was developed in [23]. The concept of FTS has been revisited by [24–28], which provided operative test conditions in light of linear matrix inequality (LMI). More recently, the concept of FTS was generalized to the finite-time consensus. In [29], Sun et al. studied the finite-time consensus problems of the leader-following multiagent systems with jointly reachable leader and switching jointly reachable leader. The finite-time synchronization between two complex networks with nondelayed and delayed was proposed by using the impulsive control and the periodically intermittent control in [30]. The consensus problems of second-order multiagent systems in the presence of one and multiple leaders under a direction graph were investigated in [31].

Motivated by the concept of finite-time stability (FTS) which was first introduced in the control literature by Dorato in [32] and correspondingly previous works (see [24–28]), we extend the concept of FTS to finite-time consensus (FTC), which is different from the concept involved in [29–31]. Compared with classical Lyapunov consensus problem, finite-time consensus here is an independent concept, which concerns the consensus of multiagent systems over a finite-time interval and may play an important part in the study of the transient behavior of system. First, we discuss the continuous-time FTC problem. Then, the discrete-time counterpart is probed. The dynamical model of agents is assumed as a general form of linear system, and the interconnection topology among the agents is assumed to be switching. While the full state information cannot be available, observer-based consensus protocols are provided to solve FTC problem. In light of LMI, we present some computationally appealing conditions to construct the gain matrices involved in the proposed protocols. Because the proposed consensus protocols are distributed, the computational complexity of design technique is only dependent on the dimension of agent's state and independent on the number of agents. LMI conditions can be solved effectively by interior-point method, and a number of software packages such as MATLAB LMI Toolbox can be available to solve LMI problems [33].

The subsequent sections are organized as follows. In Section 2, the formulation of finite-time consensus is given. Sufficient condition for finite-time consensus via state feedback and for existence of an output feedback controller guaranteeing finite-time consensus is provided, respectively, in Sections 3 and 4. This condition requires solution of an LMI problem. And discrete-time multiagent systems are investigated in Section 5. An illustrative example to verify the effectiveness of the theoretical results is provided in Section 6. Conclusion remarks are drawn in Section 7.

2. Preliminaries and Problem Formulation

2.1. Notations and Graph Theory. We first introduce the notations used in this paper. $\mathbb{R}$ (or $\mathbb{C}$) is the real (or complex) number set. I denotes an appropriate dimensioned identity matrix and $\mathbf{1}$ denotes a column vector with all components equal to one. For a given matrix A, A^T denotes its transpose and A^{-1} denotes its inverse. $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ represent the maximum and minimum eigenvalues of matrix A with real spectrum, respectively. For a symmetric matrix P, by $P > 0$ (≥ 0, <0, or ≤ 0), we mean that P is positive definite (positive semidefinite, negative, or negative semidefinite). $\|\cdot\|$ denotes Euclidean norm. The condition number of matrix A is denoted by $\mathrm{cond}(A) = \|A\| \cdot \|A^{-1}\|$. Furthermore, $\mathrm{cond}(Q) = \lambda_{\max}(Q)/\lambda_{\min}(Q)$ for positive definite Q. $\otimes$ denotes the Kronecker product, which satisfies the following: (1) $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$ and (2) if $A \geq 0$ and $B \geq 0$, then $A \otimes B \geq 0$.

We use an undirected graph to describe the involved information interaction topology, which is modeled by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ is the set of vertices representing N agents and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edges set. v_j is called a neighbor of v_i if $(v_i, v_j) \in \mathcal{E}$, and the neighbor set of vertex v_i is denoted as $\mathcal{N}_i = \{j \mid (v_i, v_j) \in \mathcal{E}\}$. $W = [w_{ij}]_{N\times N}$ represents weighted adjacency matrix associated with graph $\mathcal{G}$, where $w_{ij} > 0$ if $(v_i, v_j) \in \mathcal{E}$ and $w_{ij} = 0$ otherwise. Correspondingly, the Laplacian matrix L is defined as $l_{ii} = \sum_{j=1, j\neq i}^{N} w_{ij}$ and $l_{ij} = -w_{ij}$.

We use $\overline{\mathcal{G}}$ of order $N+1$ to model the interaction topology of the leader-following multiagent system, where the leader is represented by vertex v_0. $\overline{\mathcal{G}}$ contains a subgraph $\mathcal{G}$ and v_0 with the directed edges from some agents to the leader, where $\mathcal{G}$ described the interaction topology of N following agents. Note that the graph describing the interaction topology can vary with time. Suppose that the interconnection topology is switched among finite possible interconnection graphs, which is denoted as $\mathcal{S} = \{\overline{\mathcal{G}}_1, \overline{\mathcal{G}}_2, \ldots, \overline{\mathcal{G}}_{M_0}\}$ with index set $\mathcal{P} = \{1, 2, \ldots, M_0\}$. The switching signal $\sigma(t) \to \mathcal{P}$ is used to express the index of topology graph. Certainly, it is assumed that the chatter does not occur; that is, σ switches finite times in any bounded time interval.

Next, we introduce following well-known result, which will be used in the sequel.

Lemma 1 (see [34]). *Let S be a symmetric matrix with the partitioned form $S = [S_{ij}]$, where $S_{11} \in \mathbb{R}^{r\times r}$, $S_{12} \in \mathbb{R}^{r\times(n-r)}$, and $S_{22} \in \mathbb{R}^{(n-r)\times(n-r)}$. Then $S < 0$ if and only if*

$$S_{11} < 0, \qquad S_{22} - S_{21}S_{11}^{-1}S_{12} < 0 \tag{1}$$

or equivalently

$$S_{22} < 0, \qquad S_{11} - S_{12}S_{22}^{-1}S_{21} < 0. \tag{2}$$

2.2. Problem Formulation. Consider an MAS consisting of N following agents and a leader agent. The dynamics of agent i is

$$\begin{aligned} \dot{x}_i(t) &= Ax_i(t) + Bu_i(t), \\ y_i(t) &= Cx_i(t), \\ i &= 1, 2, \ldots, N, \end{aligned} \tag{3}$$

where $x_i \in \mathbb{R}^n$ is the agent i's state, $u_i \in \mathbb{R}^m$ is agent i's control input, and $y_i \in \mathbb{R}^p$ is the agent i's measured output. A, B, C are constant matrices with appropriate dimensions. We always assume that the system satisfies following property.

Assumption 2. For system (3), (A, B) is stabilizable and (A, C) is observable.

The leader is an isolated agent and labeled as v_0, which is described by

$$\begin{aligned} \dot{x}_0(t) &= Ax_0(t) + Bu_0(t), \\ y_0(t) &= Cx_0(t), \end{aligned} \tag{4}$$

where $x_0 \in \mathbb{R}^n$ is the leader's state and $y_0 \in \mathbb{R}^p$ is the leader's measured output. The input $u_0(t)$ can be regarded as the common policy which is known by all following agents. Without loss of generality, we can assume that $u_0(t) = 0$. The leader-following multiagent system modeled by (3) and (4) has been investigated in many references such as [13, 16–18].

The goal of this paper is to find some sufficient conditions which guarantee the existence of a dynamic feedback controller for leader-following multiagent systems such that the consensus can be achieved over the finite interval $[0, T]$. Let $x = (x_1^T, x_2^T, \ldots, x_N^T)^T$. Based on the tracking error $x - \mathbf{1}x_0$, the concept of leader-following finite-time consensus can be formalized through the following definition, which is an extension to multiagent systems of the one give in [32].

Definition 3 (leader-following finite-time consensus). Given three positive scalars c_1, c_2, T, with $c_1 < c_2$, and a positive definite matrix R, the system (3)-(4) is said to be FTC with respect to (c_1, c_2, T, R), if

$$\begin{aligned} &\left(x(0) - \mathbf{1}x_0(0)\right)^T (I \otimes R)\left(x(0) - \mathbf{1}x_0(0)\right) \\ &\quad \le c_1 \Longrightarrow \left(x(t) - \mathbf{1}x_0(t)\right)^T (I \otimes R)\left(x(t) - \mathbf{1}x_0(t)\right) \\ &\quad \le c_2, \quad \forall t \in [0, T]. \end{aligned} \tag{5}$$

Remark 4. The linear system $\dot{x}(t) = Ax(t)$, $x(0) = x_0$, is said to be FTS with respect to (c_1, c_2, T, R), if

$$x_0^T R x_0 \le c_1 \Longrightarrow x^T(t) R x(t) < c_2, \quad \forall t \in [0, T]. \tag{6}$$

Lyapunov asymptotic stability and FTS are independent concepts: a system which is FTS may not be Lyapunov asymptotically stable; conversely a Lyapunov asymptotically stable system could not be FTS if, during the transients, its state exceeds the prescribed bounds [32].

3. Finite-Time Consensus with State Feedback

In this section, we investigate the finite-time consensus problem via distributed state feedback control protocol. The proposed protocol for the following agent i, which is based on the relative state error of agent i with its neighbor agents, is given as follows:

$$\begin{aligned} u_i(t) = -cK \Bigg[&\sum_{j\in\mathcal{N}_i(t)} a_{ij}(t)\left(x_i(t) - x_j(t)\right) \\ &+ d_i(t)\left(x_i(t) - x_0(t)\right) \Bigg], \end{aligned} \tag{7}$$

where c is the positive coupling strength, $a_{ij}(t)$, $(i, j = 1, 2, \ldots, N)$ and $d_i(t)$, $(i = 1, 2, \ldots, N)$ are connection weights, which are chosen as follows:

$$\begin{aligned} a_{ij}(t) &= \begin{cases} \alpha_{ij} & \text{if agent } i \text{ is connected to agent } j, \\ 0 & \text{otherwise}, \end{cases} \\ d_i(t) &= \begin{cases} \beta_i & \text{if agent } i \text{ is connected to the leader}, \\ 0 & \text{otherwise}, \end{cases} \end{aligned} \tag{8}$$

where α_{ij} is connection weight constant between agent i and agent j, and $\beta_i > 0$ $(i = 1, \ldots, N)$ is connection weight constant between agent i and leader.

Let $x(t) = (x_1^T(t), x_2^T(t), \ldots, x_N^T(t))^T \in \mathbb{R}^{Nn}$. Then, the overall system dynamics is

$$\begin{aligned} \dot{x}(t) = &\left[I_N \otimes A - L_{\sigma(t)} \otimes (cBK)\right] x(t) \\ &- \left[D_{\sigma(t)} \otimes (cBK)\right]\left(x(t) - \mathbf{1} \otimes x_0(t)\right), \end{aligned} \tag{9}$$

where $L_{\sigma(t)}$ is the Laplacian matrix of the interaction graph $G_{\sigma(t)}$ and $D_{\sigma(t)}$ is an $N\times N$ diagonal matrix whose ith diagonal element is $d_i(t)$. For convenience, let $H_{\sigma(t)} = L_{\sigma(t)} + D_{\sigma(t)}$.

Lemma 5 (see [6]). *If graph $\overline{\mathcal{G}}$ is connected and undirected, then the symmetric matrix H is positive definite.*

Since we assume that the graphs $\overline{\mathcal{G}}_{\sigma(t)}$ are always connected, then $H_{\sigma(t)}$ are positive definite. According to Lemma 5 and the fact that $\mathcal{S}$ is a finite set, define $\underline{\lambda} := \min\{\lambda_{\min}(H_p) : p \in \mathcal{S}\}$, $\overline{\lambda} := \max\{\lambda_{\max}(H_p) : p \in \mathcal{S}\}$, which are well defined and positive.

The parameter matrix K and the coupling gain c can be constructed as follows.

Algorithm 6. (1) Let P be a solution of the inequality

$$PA^T + AP - BB^T - \alpha P < 0, \tag{10}$$

where α is a nonnegative scalar. Choose the feedback gain matrix K as

$$K = \frac{1}{2}B^T P^{-1}. \tag{11}$$

(2) Select the coupling gain c satisfied as

$$c \geq \frac{1}{\lambda}. \tag{12}$$

Remark 7. If (A, B) is stabilizable and $\overline{Q}$ is a symmetric positive definite matrix, then the following Riccati equation

$$A^T P + PA - PBB^T P + \overline{Q} = 0 \tag{13}$$

has a unique positive definite matrix P [35]. Since (A, B) is stabilizable, we know that $(A - (1/2)\alpha I, B)$ is stabilizable too. Thus, for any positive definite $\overline{Q}$, the following Riccati equation

$$\left(A - \frac{1}{2}\alpha I\right)^T \overline{P} + \overline{P}\left(A - \frac{1}{2}\alpha I\right) - \overline{P}BB^T\overline{P} + \overline{Q} = 0 \tag{14}$$

has a unique positive definite matrix $\overline{P}$. Let $P = \overline{P}^{-1} > 0$, which satisfies (10). Therefore, the LMI (10) is solvable.

Now we can obtain the following result.

Theorem 8. *Consider the multiagent system* (3)-(4) *whose topology graph* $\overline{\mathcal{G}}_{\sigma(t)}$ *that is associated with any interval* $[t_j, t_{j+1})$ *is undirected and connected. The feedback gain matrix* K *and the coupling strength* c *are able to be constructed by Algorithm 6. If the positive definite matrix* $Q = R^{1/2}P^{-1}R^{1/2}$ *satisfies the following condition:*

$$\mathrm{cond}(Q) < \frac{c_2}{c_1}e^{-\alpha T}, \tag{15}$$

then under the state feedback controller (7)*, the leader-following multiagent system* (3) *and* (4) *is finite-time consensus with respect to* (c_1, c_2, T, R).

Proof. Denote $\varepsilon(t) = x(t) - \mathbf{1} \otimes x_0(t)$, which represents the tracking error vector. In view of (4), (9) and $L_{\sigma(t)}\mathbf{1} = 0$, the dynamics of tracking error is expressed as

$$\dot{\varepsilon}(t) = \left(I_N \otimes A - cH_{\sigma(t)} \otimes BK\right)\varepsilon(t). \tag{16}$$

Then, the leader-following finite-time consensus problem is converted into finite-time stability problem.

Let P be a solution of (10) such that the condition (15) is satisfied. Consider the following common Lyapunov function:

$$V(t) = \varepsilon^T(t)\left(I_N \otimes \overline{P}\right)\varepsilon(t), \tag{17}$$

where $\overline{P} = P^{-1}$. Let $\sigma(t) = p$. The derivative of (17) along the trajectories of (16) yields

$$\begin{aligned}
\dot{V}(t) &= \varepsilon^T(t)\left[I_N \otimes \left(A^T\overline{P} + \overline{P}A\right)\right. \\
&\qquad \left. - c\left(H_p^T + H_p\right) \otimes \overline{P}BK\right]\varepsilon(t) \\
&= \varepsilon^T(t)\left[I_N \otimes \left(A^T\overline{P} + \overline{P}A\right)\right. \\
&\qquad \left. -\frac{1}{2}c\left(H_p^T + H_p\right) \otimes \overline{P}BB^T\overline{P}\right]\varepsilon(t) \\
&\leq \varepsilon^T(t)\left[I_N \otimes \left(A^T\overline{P} + \overline{P}A - \overline{P}BB^T\overline{P}\right)\right]\varepsilon(t) \\
&= \varepsilon^T(t)\left[I_N \otimes \overline{P}\left(PA^T + AP - BB^T\right)\overline{P}\right]\varepsilon(t) \\
&< \varepsilon^T(t)\left(I_N \otimes \alpha\overline{P}\right)\varepsilon(t) \\
&= \alpha V(t).
\end{aligned} \tag{18}$$

By integrating inequality (18) between 0 and t it follows that

$$V(\varepsilon(t)) < V(\varepsilon(0))e^{\alpha t}. \tag{19}$$

By the fact $\overline{P} = R^{1/2}QR^{1/2}$, we can get the following chain of inequalities:

$$\begin{aligned}
V(\varepsilon(t)) &\geq \lambda_{\min}(Q)\varepsilon^T(t)\left(I_N \otimes R\right)\varepsilon(t), \\
V(\varepsilon(0))e^{\alpha t} &\leq \lambda_{\max}(Q)\varepsilon^T(0)\left(I_N \otimes R\right)\varepsilon(0)e^{\alpha t} \\
&\leq \lambda_{\max}(Q)c_1 e^{\alpha t}.
\end{aligned} \tag{20}$$

Putting together (19) and (20), we have

$$\varepsilon^T(t)\left(I_N \otimes R\right)\varepsilon(t) < \frac{\lambda_{\max}(Q)}{\lambda_{\min}(Q)}c_1 e^{\alpha t} < \frac{c_2}{c_1}e^{-\alpha T}c_1 e^{\alpha t} < c_2. \tag{21}$$

From (21), the proof is complete. □

Remark 9. LMI (10) must be solvable for any $\alpha \geq 0$. Let $\Gamma(\alpha)$ be the positive definite solution set of (10) with parameter $\alpha \geq 0$. It is easy to see that while $0 \leq \alpha_1 < \alpha_2$, $\Gamma(\alpha_1) \subset \Gamma(\alpha_2)$. Additionally, if c_2/c_1 is big enough, then condition (15) must hold. By (15), consider the optimization problem:

$$\min_{\alpha>0, P\in\Gamma(\alpha)} \mathrm{cond}\left(R^{1/2}P^{-1}R^{1/2}\right)e^{\alpha T}. \tag{22}$$

Obviously, we can construct P and α satisfying (15) based our design approach, while c_2/c_1 is greater than the optimal value of (22). Then, the finite-time consensus problem with respect to (c_1, c_2, T, R) can be solved by the proposed protocol in this case. Furthermore, if there exists P such that (10) and (15) with $\alpha = 0$ are satisfied, it is not difficult to obtain $\dot{V}(x) < 0$ from (18), which means that the leader-following multiagent system (3) and (4) is not only finite-time consensus but also asymptotically consensus. Obviously, condition (15) is

satisfied if there exists a positive definite solution P of (10) such that the following LMI holds:

$$\lambda_0 e^{\alpha T} R^{-1} < P < \lambda_0 \frac{c_2}{c_1} R^{-1}, \tag{23}$$

with positive constant λ_0. Once a value for α is fixed, the design of a state feedback controller to make multiagent system achieve finite-time consensus is to solve LMIs (10) and (23). The LMI problems can be solved by a number of software packages such as the LMI Control Toolbox of MATLAB [33].

4. Finite-Time Consensus with State Observer

This section investigates the finite-time consensus problem with state observer-based protocol. In some practical systems, the full state is unavailable. At time t, agent i At time t, the relative output error with its neighbor agents can be available for agent i, which is denoted by, which is denoted by

$$\xi_i(t) = \sum_{j \in \mathcal{N}_i(t)} a_{ij}(t) \left(y_i(t) - y_j(t)\right) + d_i(t) \left(y_i(t) - y_0(t)\right). \tag{24}$$

To solve the leader-following multiagent finite-time consensus problem, consider the Luenberger observer for agent i with form

$$\begin{aligned} \dot{v}_i(t) = {} & Av_i(t) + Bu_i(t) \\ & - cG \left[\sum_{j \in \mathcal{N}_i(t)} a_{ij}(t) C \left(v_i(t) - v_j(t)\right) \right. \\ & \left. + d_i(t) Cv_i(t) - \xi_i(t) \right], \quad v_i(0) = 0, \end{aligned} \tag{25}$$

where $v_i \in \mathbb{R}^n$ is the protocol state, c is the coupling strength, and $G \in \mathbb{R}^{n \times p}$ is a given gain matrix.

The feedback controller is

$$u_i(t) = -Kv_i(t), \tag{26}$$

where K is a given feedback gain matrix. It is assumed that conditions (10) and (15) are solvable and K is designed by (11).

Let $\varepsilon_i(t) = x_i(t) - x_0(t)$, $e_i(t) = \varepsilon_i(t) - v_i(t)$, $\varepsilon(t) = [\varepsilon_1^T(t), \ldots, \varepsilon_N^T(t)]^T$, and $e(t) = [e_1^T(t), \ldots, e_N^T(t)]^T$. Then, we can get

$$\dot{\varepsilon}(t) = I_N \otimes (A - BK)\varepsilon(t) + \left(I_N \otimes BK\right) e(t), \tag{27}$$

$$\dot{e}(t) = \left(I_N \otimes A - cH_{\sigma(t)} \otimes GC\right) e(t), \tag{28}$$

with $\varepsilon(0) = x(0) - \mathbf{1} \otimes x_0(0)$ and $e(0) = x(0) - \mathbf{1} \otimes x_0(0)$.

Therefore the system state evolution is determined by the closed loop $I_N \otimes (A - BK)$ and by the behavior of the exogenous input $e(t)$. The goal of this section is to design an observer gain G in (25) such that the leader-following FTC property of the system is not lost in the presence of the estimation error. If such a control gain G exists, the corresponding observer is also a dynamic output feedback controller which can solve the following problem. Certainly, the existence of such a controller implies finite-time consensus via state feedback. Therefore, without loss of generality, we present the following assumption.

Assumption 10. A state feedback matrix K which guarantees the leader-following multiagent finite-time consensus via state feedback exists and has been designed using the results of Theorem 8.

In the sequel, we try to solve the following observer-based finite-time consensus problem.

Problem 11 (FTC via observer-based output feedback). Given a gain matrix K such that the multiagent system (3)-(4) is FTC wrt (c_1, c_2, T, R) via state feedback, find an observer gain G such that system (27) is FTC wrt (c_1, c_2, W_G, T, R), where W_G is the set

$$\begin{aligned} W_G := \big\{ & e(t) \mid \dot{e}(t) = \left(I_N \otimes A - cH_{\sigma(t)} \otimes GC\right) e(t), \\ & e(0) = \varepsilon(0), \varepsilon^T(0) \left(I_N \otimes R\right) \varepsilon(0) \le c_1 \big\}. \end{aligned} \tag{29}$$

From (27) and (28), the tracking error dynamical system can be expressed as

$$\dot{\eta}(t) = F_{\sigma(t)} \eta(t), \tag{30}$$

where $\eta = (\varepsilon^T, e^T)^T$ and

$$F_{\sigma(t)} = \begin{pmatrix} I_N \otimes (A - BK) & I_N \otimes (BK) \\ 0 & I_N \otimes A - cH_{\sigma(t)} \otimes (GC) \end{pmatrix}. \tag{31}$$

Obviously, the finite-time stability of system (30) implies that the finite-time consensus of leader-following system (3)-(4). Thus, the leader-following finite-time consensus problem of multiagent system is transformed into the finite-time stability problem of error dynamic system (30).

Now, we can present our main result as follows.

Theorem 12. *Consider the multiagent system* (3)-(4) *whose topology graph* $\overline{\mathcal{G}}_{\sigma(t)}$ *that is associated with any interval* $[t_j, t_{j+1})$ *is undirected and connected. Problem 11 is solvable if, letting* $c \ge 1/\overline{\lambda}$, $P_1 = R^{1/2} Q_1 R^{1/2}$, *and* $P_2 = R^{1/2} Q_2 R^{1/2}$, *there exist a nonnegative scalar* α, *two symmetric positive definite matrices* Q_1 *and* Q_2, *and positive scalars* λ_k, $k = 1, 2, 3$, *such that*

$$\begin{pmatrix} A^T P_1 + P_1 A - P_1 BK - K^T B^T P_1 - \alpha P_1 & P_1 BK \\ \left(P_1 BK\right)^T & A^T P_2 + P_2 A - C^T C - \alpha P_2 \end{pmatrix} < 0, \tag{32}$$

$$\lambda_3 I < Q_1 < \lambda_1 I, \tag{33a}$$

$$0 < Q_2 < \lambda_2 I, \tag{33b}$$

$$c_1(\lambda_1 + \lambda_2) \le c_2 e^{-\alpha T}\lambda_3. \tag{33c}$$

In this case the consensus protocols (25) *and* (26) *with gain matrix* $G = (1/2)P_2^{-1}C^T$ *can make the multiagent system* (3)-(4) *FTC with respect to* (c_1, c_2, W_G, T, R).

Proof. Set $\sigma(t) = p$, $p \in \{1, 2, \ldots, M_0\}$. Since H_p is symmetric, there exists an orthogonal matrix T_p such that

$$T_p H_p T_p^T = \Lambda_p = \operatorname{diag}(\lambda_1 p, \lambda_2 p, \ldots, \lambda_N p), \tag{34}$$

where $\lambda_i p$ is the ith eigenvalue of H_p. By using the following orthogonal transformation to system (30):

$$\widetilde{\eta} = \begin{pmatrix} I_N \otimes I_n & 0 \\ 0 & T_p \otimes I_n \end{pmatrix} \eta, \tag{35}$$

we can get the equivalent system of system (30) as

$$\dot{\widetilde{\eta}}(t) = \widetilde{F}_p \widetilde{\eta}(t), \tag{36}$$

where $\widetilde{\eta} = (\widetilde{\varepsilon}^T, \widetilde{e}^T)^T$ and

$$\widetilde{F}_p = \begin{pmatrix} I_N \otimes (A - BK) & I_N \otimes BK \\ 0 & I_N \otimes A - c\Lambda_p \otimes GC \end{pmatrix}. \tag{37}$$

That is

$$\dot{\widetilde{\eta}}_i(t) = \begin{pmatrix} A - BK & BK \\ 0 & A - c\lambda_{ip} GC \end{pmatrix} \widetilde{\eta}_i(t), \tag{38}$$

where $\widetilde{\eta}_i = (\widetilde{\varepsilon}_i^T, \widetilde{e}_i^T)^T$.

Consider the following Lyapunov function:

$$V(\eta(t)) = \eta^T(t)\,\widetilde{P}\,\eta(t), \tag{39}$$

where

$$\widetilde{P} = \begin{pmatrix} I_N \otimes P_1 & 0 \\ 0 & I_N \otimes P_2 \end{pmatrix}, \tag{40}$$

$V(\eta(t))$ is continuously differentiable at any time except for switching instants.

Consider

$$\begin{aligned} &[I_N \otimes (A - BK)]^T (I_N \otimes P_1) \\ &\quad + (I_N \otimes P_1)[I_N \otimes (A - BK)] \\ &= I_N \otimes \left(A^T P_1 + P_1 A - P_1 BK - K^T B^T P_1\right). \end{aligned} \tag{41}$$

Noting that $P_2 GC = (1/2)C^T C$, we have

$$\begin{aligned} &\left(I_N \otimes A - cH_p \otimes (GC)\right)^T (I_N \otimes P_2) \\ &\quad + (I \otimes P_2)\left(I \otimes A - cH_p \otimes (GC)\right) \\ &= I_N \otimes \left(A^T P_2 + P_2 A\right) - \frac{1}{2} c \left(H_p^T + H_p\right) \otimes C^T C \\ &< I_N \otimes \left(A^T P_2 + P_2 A - C^T C\right). \end{aligned} \tag{42}$$

Then derivative of (39) along the trajectories of (30) yields

$$\begin{aligned} \dot{V}(\eta(t)) &= \eta^T(t)\left(F_p^T \widetilde{P} + \widetilde{P} F_p\right)\eta(t) \\ &= \widetilde{\eta}^T(t)\left(\widetilde{F}_p^T \widetilde{P} + \widetilde{P}\widetilde{F}_p\right)\widetilde{\eta}(t) \\ &\le \widetilde{\eta}^T(t)\, Q_p \widetilde{\eta}(t) \\ &= \sum_{i=1}^{N} \widetilde{\eta}_i^T(t)\, Q_{ip} \widetilde{\eta}_i(t), \end{aligned} \tag{43}$$

where

$$Q_{ip} = \begin{pmatrix} A^T P_1 + P_1 A - P_1 BK - K^T B^T P_1 & P_1 BK \\ (P_1 BK)^T & A^T P_2 + P_2 A - C^T C \end{pmatrix}. \tag{44}$$

From (32), we obtain

$$Q_{ip} < \begin{pmatrix} \alpha P_1 & 0 \\ 0 & \alpha P_2 \end{pmatrix}, \tag{45}$$

$$\dot{V}(t) < \alpha V(t). \tag{46}$$

By integrating inequality (46) between 0 and t it follows that

$$V(\eta(t)) < V(\eta(0))\, e^{\alpha t}. \tag{47}$$

We have the following chain of inequalities:

$$\begin{aligned} V(\eta(t)) &\ge \lambda_{\min}(Q_1)\,\varepsilon^T(t)(I_N \otimes R)\,\varepsilon(t) \\ &\quad + \lambda_{\min}(Q_2)\, e^T(t)(I_N \otimes R)\, e(t) \\ &\ge \lambda_{\min}(Q_1)\,\varepsilon^T(t)(I_N \otimes R)\,\varepsilon(t). \end{aligned} \tag{48}$$

In additionally, one has

$$\begin{aligned} V(\eta(0))\, e^{\alpha t} &\le \left(\lambda_{\max}(Q_1)\,\varepsilon^T(0)(I_N \otimes R)\,\varepsilon(0)\right. \\ &\quad \left. + \lambda_{\max}(Q_2)\, e^T(0)(I_N \otimes R)\, e(0)\right) e^{\alpha t} \\ &\le \left(\lambda_{\max}(Q_1) + \lambda_{\max}(Q_2)\right) c_1 e^{\alpha T}. \end{aligned} \tag{49}$$

Putting together (47), (48), and (49), we have

$$\varepsilon^T(t)\left(I_N \otimes R\right)\varepsilon(t) < \frac{\left(\lambda_{\max}(Q_1) + \lambda_{\max}(Q_2)\right)}{\lambda_{\min}(Q_1)} c_1 e^{\alpha T}. \quad (50)$$

Since

$$\begin{aligned} \lambda_3 &< \lambda_{\min}(Q_1), \qquad \lambda_{\max}(Q_1) < \lambda_1, \\ 0 &< \lambda_{\min}(Q_2), \qquad \lambda_{\max}(Q_2) < \lambda_2, \\ \lambda_1 c_1 &+ \lambda_2 c_1 \le c_2 e^{-\alpha T}\lambda_3, \end{aligned} \quad (51)$$

which in turn guarantees that

$$\left(\lambda_{\max}(Q_1) + \lambda_{\max}(Q_2)\right) c_1 < c_2 e^{-\alpha T}\lambda_{\min}(Q_1), \quad (52)$$

then we can get

$$\varepsilon^T\left(I_N \otimes R\right)\varepsilon < \frac{1}{\lambda_{\min}(Q_1)} c_2 e^{-\alpha T}\lambda_{\min}(Q_1) e^{\alpha T} = c_2, \quad (53)$$

for all $t \in [0, T]$. □

5. Discrete-Time Multiagent Systems

This section focuses on the discrete-time counterpart of the last section. Consider a network of N identical discrete-time linear agents and one leader, with the dynamics of the ith agent described by

$$\begin{aligned} x_i(k+1) &= Ax_i(k) + Bu_i(k), \\ y_i(k) &= Cx_i(k), \\ i &= 1, 2, \ldots, N \end{aligned} \quad (54)$$

and the dynamics of the leader is described by

$$\begin{aligned} x_0(k+1) &= Ax_0(k), \\ y_0(k) &= Cx_0(k). \end{aligned} \quad (55)$$

Definition 13 (leader-following finite-time consensus). Given three positive scalars c_1, c_2, M with $c_1 < c_2$, and a positive definite matrix R, the discrete-time multiagent system (54)-(55) is said to be finite-time consensus with respect to (c_1, c_2, M, R), if

$$\begin{aligned} &\left(x(0) - 1 \otimes x_0(0)\right)^T \left(I_N \otimes R\right)\left(x(0) - 1 \otimes x_0(0)\right) \le c_1 \\ &\Longrightarrow \left(x(k) - 1 \otimes x_0(k)\right)^T \\ &\quad \times \left(I_N \otimes R\right)\left(x(k) - 1 \otimes x_0(k)\right) \le c_2, \\ &\qquad k = 1, 2, \ldots, M. \end{aligned} \quad (56)$$

5.1. Discrete-Time Finite-Time Consensus with State Feedback. First, we investigate finite-time consensus problem via distributed state feedback control protocol. The proposed protocol for the following agent i is constructed as follows:

$$\begin{aligned} u_i(k) = &- cK \sum_{j \in \mathcal{N}_i(k)} a_{ij}(k)\left(x_i(k) - x_j(k)\right) \\ &+ d_i(k)\left(x_i(k) - x_0(k)\right). \end{aligned} \quad (57)$$

Let $\varepsilon(k) = x(k) - 1 \otimes x_0(k)$. Similarly, we can get

$$\varepsilon(k+1) = \left(I_N \otimes A - cH_{\sigma(k)} \otimes BK\right)\varepsilon(k). \quad (58)$$

Here we present our result about discrete-time finite-time consensus with state feedback.

Theorem 14. *Consider the multiagent system* (54)-(55) *whose topology graph* $\overline{\mathcal{G}}_{\sigma(k)}$ *that is associated with any interval* $[t_j, t_{j+1})$ *is undirected and connected. If there exists a positive definite matrix* $Q \in \mathbb{R}^{n\times n}$*, a matrix* $S \in \mathbb{R}^{m\times n}$ *and a scalar* $\gamma \ge 1$ *such that*

$$\begin{pmatrix} -\gamma Q & \left(AQ - c\overline{\lambda}BS\right)^T \\ \left(AQ - c\overline{\lambda}BS\right) & -Q \end{pmatrix} < 0, \quad (59)$$

$$\begin{pmatrix} -\gamma Q & \left(AQ - c\widehat{\lambda}BS\right)^T \\ \left(AQ - c\widehat{\lambda}BS\right) & -Q \end{pmatrix} < 0, \quad (60)$$

$$\frac{\lambda_{\max}\left(\widetilde{Q}\right)}{\lambda_{\min}\left(\widetilde{Q}\right)} < \frac{1}{\gamma^M}\frac{c_2}{c_1}, \quad (61)$$

where $\widetilde{Q} = R^{1/2}QR^{1/2}$*, and the feedback gain matrix* K *is taken by* $K = SQ^{-1}$*. Then the multiagent system* (54)-(55) *is finite-time consensus with respect to* (c_1, c_2, R, M).

Proof. Let $\sigma(k) = p$, $p \in \{1, 2, \ldots, M_0\}$. Due to $\lambda_{ip} \in [\overline{\lambda}, \widehat{\lambda}]$, there exist $\alpha_{ip} \ge 0$ and $\beta_{ip} \ge 0$ satisfying $\lambda_{ip} = \alpha_{ip}\overline{\lambda} + \beta_{ip}\widehat{\lambda}$ and $\alpha_{ip} + \beta_{ip} = 1$.

From (59), (60), we get

$$\begin{aligned} &\begin{pmatrix} -\gamma Q & \left(AQ - c\lambda_{ip}BS\right)^T \\ \left(AQ - c\lambda_{ip}BS\right) & -Q \end{pmatrix} \\ &= \alpha_{ip}\begin{pmatrix} -\gamma Q & \left(AQ - c\overline{\lambda}BS\right)^T \\ \left(AQ - c\overline{\lambda}BS\right) & -Q \end{pmatrix} \\ &\quad + \beta_{ip}\begin{pmatrix} -\gamma Q & \left(AQ - c\widehat{\lambda}BS\right)^T \\ \left(AQ - c\widehat{\lambda}BS\right) & -Q \end{pmatrix} < 0. \end{aligned} \quad (62)$$

Pre- and postmultiplying (62) by the symmetric matrix $\begin{pmatrix} Q^{-1} & 0 \\ 0 & I \end{pmatrix}$, the following equivalent condition is obtained:

$$\begin{pmatrix} -\gamma Q^{-1} & \left(A - c\lambda_{ip}BK\right)^T \\ \left(A - c\lambda_{ip}BK\right) & -Q \end{pmatrix} < 0. \quad (63)$$

Consider the following common Lyapunov function:

$$V(\varepsilon(k)) = \varepsilon^T(k)\left(I_N \otimes P\right)\varepsilon(k), \quad (64)$$

where $P = Q^{-1}$; then we get

$$\begin{aligned} V(\varepsilon(k+1)) &= \varepsilon^T(k+1)\left(I_N \otimes P\right)\varepsilon(k+1) \\ &= \sum_{i=1}^{N} \varepsilon_i^T(k)\left(A - c\lambda_{ip}BK\right)^T \\ &\quad \times P\left(A - c\lambda_{ip}BK\right)\varepsilon_i(k), \end{aligned} \tag{65}$$

where equation (63) implies

$$V(\varepsilon(k+1)) < \gamma V(\varepsilon(k)). \tag{66}$$

Applying iteratively (66), we obtain

$$V(\varepsilon(k)) < \gamma^k V(\varepsilon(0)), \quad k = 1, 2, \ldots, M. \tag{67}$$

Now letting $\widetilde{P} = R^{-1/2}PR^{-1/2}$, it is obvious to see that $\lambda_{\min}(\widetilde{P}) = 1/\lambda_{\max}(\widetilde{Q})$ and $\lambda_{\max}(\widetilde{P}) = 1/\lambda_{\min}(\widetilde{Q})$. And using the fact that $\gamma \geq 1$, we have

$$\begin{aligned} \gamma^k V(\varepsilon(0)) &= \gamma^k\left[\varepsilon^T(0)\left(I_N \otimes P\right)\varepsilon(0)\right] \\ &\leq \gamma^k\left[\lambda_{\max}\left(\widetilde{P}\right)\varepsilon^T(0)\left(I_N \otimes R\right)\varepsilon(0)\right] \\ &\leq \gamma^M \lambda_{\max}\left(\widetilde{P}\right)c_1, \\ V(\varepsilon(k)) &= \left[\varepsilon^T(k)\left(I_N \otimes P\right)\varepsilon(k)\right] \\ &\geq \lambda_{\min}\left(\widetilde{P}\right)\varepsilon^T(k)\left(I_N \otimes R\right)\varepsilon(k). \end{aligned} \tag{68}$$

Putting together (67) and (68), we obtain

$$\varepsilon^T(k)\left(I_N \otimes R\right)\varepsilon(k) < \frac{\lambda_{\max}\left(\widetilde{P}\right)}{\lambda_{\min}\left(\widetilde{P}\right)}\gamma^M c_1 = \frac{\lambda_{\max}\left(\widetilde{Q}\right)}{\lambda_{\min}\left(\widetilde{Q}\right)}\gamma^M c_1 < c_2. \tag{69}$$

Then, system (54)-(55) is finite-time consensus with respect to (c_1, c_2, R, M). □

Remark 15. Once we have fixed a value for γ, the feasibility of the conditions stated in the Theorem 14 can be turned into LMI-based feasibility problem. To this aim, it is easy to check that condition (61) is guaranteed by imposing the conditions:

$$\mu_1 R^{-1} < Q < R^{-1}, \tag{70}$$

$$\frac{c_1}{\mu_1} < \frac{c_2}{\gamma^M}, \tag{71}$$

for positive number μ_1. Letting $\delta = \sqrt{c_1}$, inequality (71) is equivalent to the following LMI by using Schur Complement Lemma:

$$\begin{pmatrix} \frac{c_2}{r^M} & \delta \\ \delta & \mu_1 \end{pmatrix} > 0. \tag{72}$$

5.2. Discrete-Time Finite-Time Consensus with State Observer. Now we consider that the full state of agent i is unknown, and we only know the output of agent i. At time k, agent i can be available to the relative output error with its neighbor agents, which is denoted by

$$\begin{aligned} \xi_i(k) = &\sum_{j \in \mathcal{N}_i(k)} a_{ij}(k)\left(y_i(k) - y_j(k)\right) \\ &+ d_i(k)\left(y_i(k) - y_0(k)\right). \end{aligned} \tag{73}$$

To solve the leader-following multiagent finite-time consensus problem, consider the discrete-time Luenberger observer for agent i with form

$$\begin{aligned} v_i(k+1) = &Av_i(k) + Bu_i(k) \\ &- cG\left[\sum_{j \in \mathcal{N}_i(k)} a_{ij}(k)C\left(v_i(k) - v_j(k)\right)\right. \\ &\left. + d_i(k)Cv_i(k) - \xi_i(k)\right], \quad v_i(0) = 0, \end{aligned} \tag{74}$$

where $v_i(k) \in \mathbb{R}^n$ is the protocol state, c is the coupling strength, $G \in \mathbb{R}^{n \times p}$ is a gain matrix. The feedback controller is taken by

$$u_i(k) = -Kv_i(k), \tag{75}$$

where K is a given feedback gain matrix.

Taking the similar step as the continuous-time case, we can get

$$\varepsilon(k+1) = \left[I_N \otimes (A - BK)\right]\varepsilon(k) + \left(I_N \otimes BK\right)e(k), \tag{76}$$

with

$$e(k+1) = \left(I_N \otimes A - cH_{\sigma(k)} \otimes GC\right)e(k). \tag{77}$$

The goal of this section is to design an observer gain G in (77) such that the leader-following FTC property of the system is not lost under given controller K in present of the estimation error. Here, we also assume that K has been designed by result of Theorem 14. Similarly, we consider the following problem.

Problem 16 (FTC via observer-based output feedback). Given a gain matrix K such that the multiagent system (54)-(55) is FTC with respect to (c_1, c_2, M, R) via state feedback, find an observer gain G such that system (76) is FTC with respect to (c_1, c_2, W_G, M, R), where W_G is the set

$$\begin{aligned} W_G := \{e(k) \mid &e(k+1) = \left(I_N \otimes A - cH_{\sigma(k)} \otimes GC\right)e(k), \\ &e(0) = \varepsilon(0), \varepsilon^T(0)\left(I_N \otimes R\right)\varepsilon(0) \leq c_1\}. \end{aligned} \tag{78}$$

Here we give our result for the discrete-time multiagent systems.

Theorem 17. *Consider the multiagent system* (54) *and* (55) *whose topology graph* $\overline{\mathcal{G}}_{\sigma(k)}$ *that is associated with any interval* $[t_j, t_{j+1})$ *is undirected and connected. The problem is solvable if there exist positive-definite matrices* Q_1, Q_2, *a matrix* T, *and a scalar* $\gamma \geq 1$ *such that*

$$\begin{pmatrix} (A-BK)^T Q_1 (A-BK) - \gamma Q_1 & (A-BK)^T Q_1 (BK) & 0 \\ (BK)^T Q_1 (A-BK) & (BK)^T Q_1 (BK) - \gamma Q_2 & \left(Q_2 A - c\overline{\lambda} TC\right)^T \\ 0 & \left(Q_2 A - c\overline{\lambda} TC\right) & -Q_2 \end{pmatrix} < 0, \tag{79}$$

$$\begin{pmatrix} (A-BK)^T Q_1 (A-BK) - \gamma Q_1 & (A-BK)^T Q_1 (BK) & 0 \\ (BK)^T Q_1 (A-BK) & (BK)^T Q_1 (BK) - \gamma Q_2 & \left(Q_2 A - c\widehat{\lambda} TC\right)^T \\ 0 & \left(Q_2 A - c\widehat{\lambda} TC\right) & -Q_2 \end{pmatrix} < 0, \tag{80}$$

$$\frac{\lambda_{\max}\left(\widetilde{Q}_1\right) + \lambda_{\max}\left(\widetilde{Q}_2\right)}{\lambda_{\min}\left(\widetilde{Q}_1\right)} < \frac{1}{\gamma^M} \frac{c_2}{c_1}, \tag{81}$$

where $\widetilde{Q}_1 = R^{-1/2} Q_1 R^{-1/2}$ *and* $\widetilde{Q}_2 = R^{-1/2} Q_2 R^{-1/2}$. *In this case the consensus protocols* (74) *and* (75) *with gain matrix* $G = Q_2^{-1} T$ *can make the multiagent system* (54)-(55) *FTC with respect to* (c_1, c_2, W_G, M, R).

Proof. Let $\sigma(k) = p$, $p \in \{1, 2, \ldots, M_0\}$. From (79) and (80), we get

$$\begin{pmatrix} (A-BK)^T Q_1 (A-BK) - \gamma Q_1 & (A-BK)^T Q_1 (BK) & 0 \\ (BK)^T Q_1 (A-BK) & (BK)^T Q_1 (BK) - \gamma Q_2 & \left(Q_2 A - c\lambda_{ip} TC\right)^T \\ 0 & \left(Q_2 A - c\lambda_{ip} TC\right) & -Q_2 \end{pmatrix} < 0. \tag{82}$$

Obviously, (82) is equivalent to the following inequality:

$$\begin{pmatrix} (A-BK)^T Q_1 (A-BK) - \gamma Q_1 & (A-BK)^T Q_1 (BK) \\ (BK)^T Q_1 (A-BK) & (BK)^T Q_1 (BK) + \left(A - c\lambda_{ip} GC\right)^T Q_2 \left(A - c\lambda_{ip} GC\right) - \gamma Q_2 \end{pmatrix} < 0. \tag{83}$$

Consider the following common Lyapunov function:

$$\begin{aligned} V(\varepsilon(k), e(k)) &= \varepsilon^T(k) \left(I_N \otimes Q_1\right) \varepsilon(k) \\ &\quad + e^T(k) \left(I_N \otimes Q_2\right) e(k), \end{aligned} \tag{84}$$

where equation (83) implies

$$V(\varepsilon(k+1), e(k+1)) < \gamma V(\varepsilon(k), e(k)). \tag{85}$$

Applying iteratively (85), we obtain

$$V(\varepsilon(k), e(k)) < \gamma^k V(\varepsilon(0), e(0)), \quad k = 1, 2, \ldots, M. \tag{86}$$

Since $\widetilde{Q}_1 = R^{-1/2} Q_1 R^{-1/2}$, $\widetilde{Q}_2 = R^{-1/2} Q_2 R^{-1/2}$ and the fact that $\gamma \geq 1$, we have

$$\begin{aligned} \gamma^k V(\varepsilon(0), e(0)) &= \gamma^k \left[\varepsilon^T(0) \left(I_N \otimes Q_1\right) \varepsilon(0) \right. \\ &\quad \left. + e^T(0) \left(I_N \otimes Q_2\right) e(0)\right] \\ &\leq \gamma^k \left[\lambda_{\max}\left(\widetilde{Q}_1\right) \varepsilon^T(0) \left(I_N \otimes R\right) \varepsilon(0) \right. \\ &\quad \left. + \lambda_{\max}\left(\widetilde{Q}_2\right) e^T(0) \left(I_N \otimes R\right) e(0)\right] \\ &\leq \gamma^k \left(\lambda_{\max}\left(\widetilde{Q}_1\right) + \lambda_{\max}\left(\widetilde{Q}_2\right)\right) c_1, \end{aligned} \tag{87}$$

$$V(\varepsilon(k), e(k)) = \varepsilon^T(k)(I_N \otimes Q_1)\varepsilon(k) + e^T(k)(I_N \otimes Q_2)e(k) \geq \lambda_{\min}(\widetilde{Q}_1)\varepsilon^T(k)(I_N \otimes R)\varepsilon(k). \quad (88)$$

Putting together (86)–(88) we obtain

$$\varepsilon^T(k)(I_N \otimes R)\varepsilon(k) \leq \frac{\lambda_{\max}(\widetilde{Q}_1) + \lambda_{\max}(\widetilde{Q}_2)}{\lambda_{\min}(\widetilde{Q}_1)}\gamma^k c_1 < c_2. \quad (89)$$

So the multiagent system (54)-(55) is finite-time consensus with respect to (c_1, c_2, M, R). □

Remark 18. Once we have fixed a value for γ, the feasibility of the conditions stated in Theorem 17 can be turned into LMI-based feasibility problem. To this aim, it is easy to check that condition (81) is guaranteed by imposing the conditions

$$\mu_1 R < Q_1 < R, \quad (90a)$$

$$0 < Q_2 < \mu_2 R, \quad (90b)$$

$$\frac{1+\mu_2}{\mu_1} < \frac{c_2}{\gamma^M c_1}, \quad (91)$$

for positive numbers μ_1 and μ_2. Similar, letting $\delta = \sqrt{1+\mu_2}$, inequality (91) is equivalent to the following LMI:

$$\begin{pmatrix} \frac{c_2}{c_1\gamma^M} & \delta \\ \delta & \mu_1 \end{pmatrix} > 0. \quad (92)$$

6. Simulation Example

In this section, we discuss the numerical implementation of Theorem 8 with static feedback and Theorem 12 regarding the output feedback design. The group of agents consists of four following agents and one leader; that is $N = 4$. The leader agent and following agents are modeled by the linear dynamics (4) and (3), respectively, with the following system matrices:

$$A = \begin{pmatrix} -0.7 & -0.49 & 0.3 \\ 1 & 0 & 0.4 \\ 0.5 & 0 & -1.19 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0.49 & 1.19 \\ 0 & 0.49 & 1 \end{pmatrix}. \quad (93)$$

The interconnection topologies are assumed to be arbitrarily switched among three graphs $\overline{\mathcal{G}}_i$ $(i = 1, 2, 3)$, which is shown in Figure 1.

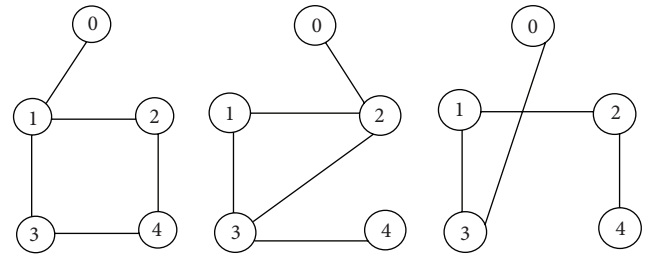


Figure 1: Three interconnection topology graphs.

The Laplacian matrices L_i $(i = 1, 2, 3)$ for subgraphs $\mathcal{G}_i$ are

$$L_1 = \begin{pmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & 0 & -1 \\ -1 & 0 & 2 & -1 \\ 0 & -1 & -1 & 2 \end{pmatrix}, \quad L_2 = \begin{pmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}, \quad L_3 = \begin{pmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix}. \quad (94)$$

The diagonal matrices for the relationship between the leader and the followers are

$$D_1 = \operatorname{diag}(1\ 0\ 0\ 0), \quad D_2 = \operatorname{diag}(0\ 1\ 0\ 0), \quad D_3 = \operatorname{diag}(0\ 0\ 1\ 0). \quad (95)$$

By simple calculations, we can obtain $\widehat{\lambda} = 4.3429$ and $\overline{\lambda} = 0.1206$.

Take $c_1 = 1$, $c_2 = 3$, and $R = I$. Our goal is to find a dynamical feedback controller by which the multiagent system (3)-(4) is FTC with respect to (c_1, c_2, T, R).

(1) Choosing $\alpha = 0$ and $T = 3$, with the aid of the LMI Control Toolbox, we can obtain control gain matrix

$$K = \begin{pmatrix} 3.9835 & 5.2242 & 1.0326 \\ 1.0326 & 1.9331 & 1.4431 \end{pmatrix} \quad (96)$$

and the gain matrix

$$G = \begin{pmatrix} 0.0008 & -0.0002 \\ -0.0002 & 0.0010 \\ 0.0004 & 0.0000 \end{pmatrix}. \quad (97)$$

Figure 2 shows that the leader-following multiagent system is asymptotic consensus, but not finite-time consensus with respect to (c_1, c_2, T, R).

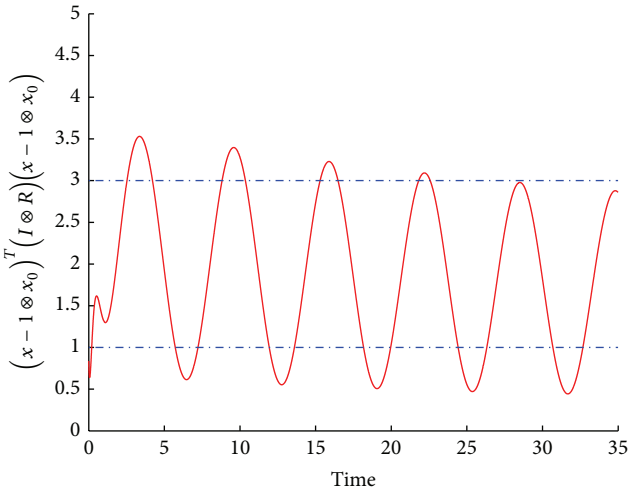


Figure 2: The error trajectories between the leader and each agent.

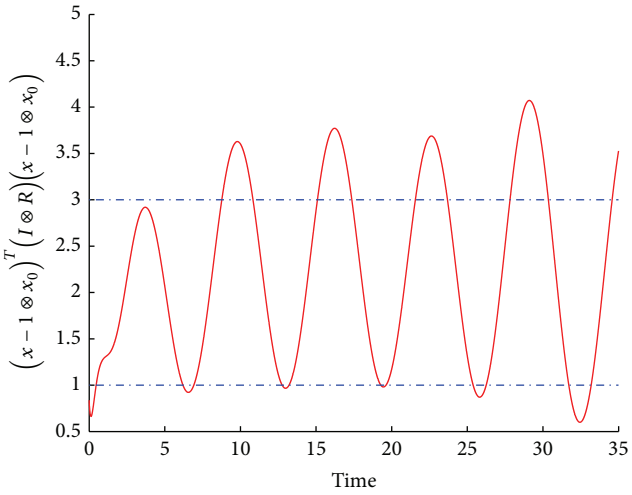


Figure 3: The error trajectories between the leader and each agent.

(2) Take $\alpha = 0.3$ and $T = 3$. With the control gain matrix $K = \begin{pmatrix} 2.5797 & 1.7563 & -0.2964 \\ -0.2964 & 0.1658 & 1.6599 \end{pmatrix}$ and the gain matrix $G = \begin{pmatrix} 0.0018 & -0.0014 \\ -0.0005 & 0.0031 \\ 0.0025 & 0.0018 \end{pmatrix}$, the trajectories of tracking errors are depicted in Figure 3, which show that the multiagent system is finite-time consensus but not asymptotic consensus.

(3) Choose $\alpha = 0$ and with the control gain matrix

$$K = \begin{pmatrix} 0.3341 & 0.1807 & 0.0365 \\ 0.0365 & 0.0591 & 0.2353 \end{pmatrix} \quad (98)$$

and the gain matrix

$$G = \begin{pmatrix} 0.2652 & -0.2133 \\ -0.0446 & 0.5563 \\ 0.4584 & 0.3170 \end{pmatrix}. \quad (99)$$

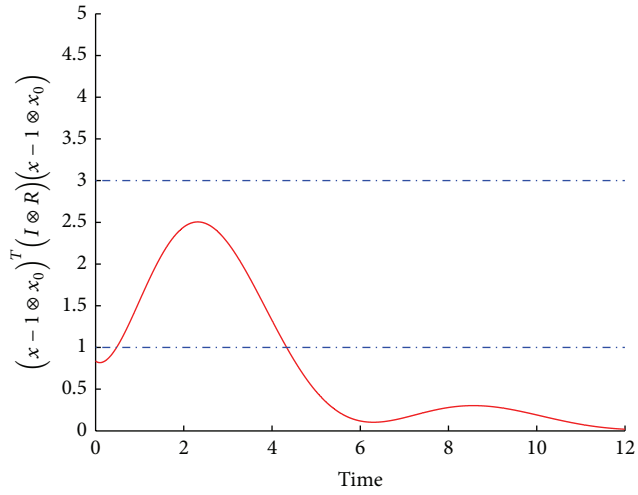


Figure 4: The error trajectories between the leader and each agent.

Figure 4 shows that the leader-following multiagent system is not only finite-time consensus with respect to (c_1, c_2, T, R) but also asymptotic consensus. From the above example, we know that when $a = 0$, the multiagent system can achieve asymptotic consensus, but maybe not finite-time consensus. Only when conditions (10) and (15) are both satisfied for $a = 0$, the multiagent system is both asymptotic consensus and finite-time consensus.

7. Conclusion

In this paper, we have discussed the finite-time consensus problem for leader-following multiagent systems with variable topology. Motivated by the concept of finite-time stability, the concept of finite-time consensus is proposed. The graph-theoretic notion is used to represent dynamical undirected interaction topologies. Two distributed consensus protocols based on its state and its output, respectively, are proposed to solve finite-time consensus. In light of linear matrix inequalities, some sufficient conditions are established to ensure that the multiagent system achieves finite-time consensus. Furthermore, we discuss the discrete-time counterpart along the similar lines. The simulation example also shows the effectiveness of the obtained theoretical results. There are some other observer/controller architectures that have been proposed to solve multiagent consensus problem. Our proposed design method can be also applied to solve the finite-time consensus problem under those architectures. Future extensions will focus on switching directed interaction topology, disturbance rejection, and robustness properties of the proposed consensus protocols.

Conflict of Interests

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

Acknowledgments

This work was supported by the National Nature Science Function of China under Granst nos. 61201074 and 61325019, the Zhejiang Provincial Natural Science Function of China under Grants nos. LY13F030048 and R1110679, and Scientific Research Fund of Zhejiang Provincial Educational Department under Grant no. Y201328776.

References

[1] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[2] R. M. Murray, "Recent research in cooperative control of multivehicle systems," *Journal of Dynamic Systems, Measurement and Control*, vol. 129, no. 5, pp. 571–583, 2007.

[3] D. Jin and L. Gao, "Stability analysis of a double integrator swarm model related to position and velocity," *Transactions of the Institute of Measurement and Control*, vol. 30, no. 3-4, pp. 275–293, 2008.

[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. R. Karimi, "Robust delay-dependent H_∞ control of uncertain time-delay systems with mixed neutral, discrete, and distributed time-delays and Markovian switching parameters," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 8, pp. 1910–1923, 2011.

[6] Y. Hong, J. Hu, and L. Gao, "Tracking control for multi-agent consensus with an active leader and variable topology," *Automatica*, vol. 42, no. 7, pp. 1177–1182, 2006.

[7] W. Ren and E. Atkins, "Distributed multi-vehicle coordinated control via local information exchange," *International Journal of Robust and Nonlinear Control*, vol. 17, no. 10-11, pp. 1002–1033, 2007.

[8] H. Du, S. Li, and P. Shi, "Robust consensus algorithm for second-order multi-agent systems with external disturbances," *International Journal of Control*, vol. 85, no. 12, pp. 1913–1928, 2012.

[9] L. Gao, X. Zhu, and W. Chen, "Leader-following consensus problem with an accelerated motion leader," *International Journal of Control, Automation, and Systems*, vol. 10, no. 5, pp. 931–939, 2012.

[10] F. Xiao and L. Wang, "Consensus problems for high-dimensional multi-agent systems," *IET Control Theory and Applications*, vol. 1, no. 3, pp. 830–837, 2007.

[11] W. He and J. Cao, "Consensus control for high-order multi-agent systems," *IET Control Theory and Applications*, vol. 5, no. 1, pp. 231–238, 2011.

[12] B. Yang, "Stability switches of arbitrary high-order consensus in multiagent networks with time delays," *The Scientific World Journal*, vol. 2013, Article ID 514823, 7 pages, 2013.

[13] W. Ni and D. Cheng, "Leader-following consensus of multi-agent systems under fixed and switching topologies," *Systems & Control Letters*, vol. 59, no. 3-4, pp. 209–217, 2010.

[14] Z. Li, Z. Duan, G. Chen, and L. Huang, "Consensus of multiagent systems and synchronization of complex networks: a unified viewpoint," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 1, pp. 213–224, 2010.

[15] Z. Li, X. Liu, P. Lin, and W. Ren, "Consensus of linear multi-agent systems with reduced-order observer-based protocols," *Systems & Control Letters*, vol. 60, no. 7, pp. 510–516, 2011.

[16] H. Zhang, F. L. Lewis, and A. Das, "Optimal design for synchronization of cooperative systems: state feedback, observer and output feedback," *IEEE Transactions on Automatic Control*, vol. 56, no. 8, pp. 1948–1952, 2011.

[17] X. Xu, S. Chen, W. Huang, and L. Gao, "Leader-following consensus of discrete-time multi-agent systems with observer-based protocols," *Neurocomputing*, vol. 118, pp. 334–341, 2013.

[18] Y. Zhang, L. Gao, and C. Tong, "On distributed reduced-order observer-based protocol for linear multi-agent consensus under switching topology," *Abstract and Applied Analysis*, vol. 2013, Article ID 793276, 13 pages, 2013.

[19] B. Xu, L. Gao, Y. Zhang, and X. Xu, "Leader-following consensus stability of discrete-time linear multiagent systems with observer-based protocols," *Abstract and Applied Analysis*, vol. 2013, Article ID 357971, 10 pages, 2013.

[20] S. P. Bhat and D. S. Bernstein, "Finite-time stability of continuous autonomous systems," *SIAM Journal on Control and Optimization*, vol. 38, no. 3, pp. 751–766, 2000.

[21] Y. Hong, "Finite-time stabilization and stabilizability of a class of controllable systems," *Systems & Control Letters*, vol. 46, no. 4, pp. 231–236, 2002.

[22] Y. Hong, J. Huang, and Y. Xu, "On an output feedback finite-time stabilization problem," *IEEE Transactions on Automatic Control*, vol. 46, no. 2, pp. 305–309, 2001.

[23] S. G. Nersesov, W. M. Haddad, and Q. Hui, "Finite-time stabilization of nonlinear dynamical systems via control vector Lyapunov functions," *Journal of the Franklin Institute*, vol. 345, no. 7, pp. 819–837, 2008.

[24] 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.

[25] F. Amato, M. Ariola, and C. Cosentino, "Finite-time stabilization via dynamic output feedback," *Automatica*, vol. 42, no. 2, pp. 337–342, 2006.

[26] F. Amato, M. Ariola, and C. Cosentino, "Finite-time control of discrete-time linear systems: analysis and design conditions," *Automatica*, vol. 46, no. 5, pp. 919–924, 2010.

[27] F. Amato, R. Ambrosino, C. Cosentino, and G. de Tommasi, "Input-output finite time stabilization of linear systems," *Automatica*, vol. 46, no. 9, pp. 1558–1562, 2010.

[28] F. Amato, G. de Tommasi, and A. Pironti, "Necessary and sufficient conditions for finite-time stability of impulsive dynamical linear system," *Automatica*, vol. 49, no. 8, pp. 2546–2550, 2013.

[29] F. Sun, J. Chen, Z.-H. Guan, L. Ding, and T. Li, "Leader-following finite-time consensus for multi-agent systems with jointly-reachable leader," *Nonlinear Analysis: Real World Applications*, vol. 13, no. 5, pp. 2271–2284, 2012.

[30] J. Mei, M. Jiang, W. Xu, and B. Wang, "Finite-time synchronization control of complex dynamicaly networks with time delay," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 9, pp. 2462–2478, 2013.

[31] Y. Zhang and Y. Yang, "Finite-time consensus of second-order leader-following multi-agent systems without velocity measurements," *Physics Letters A*, vol. 377, no. 3-4, pp. 243–249, 2013.

[32] P. Dorato, "Short time stability in linear time-varying systems," in *Proceeding of the IRE Internation Convention Record*, part 4, pp. 83–87, 1961.

[33] S. P. Boyd, L. EI Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, SIAM, Philadelphia, Pa, USA, 1994.

[34] R. Horn and C. Johnson, *Matrix Analysis*, Cambbridge University Press, New York, NY, USA, 1985.

[35] W. M. Wonham, *Linear Multivariable Contol*, Springer, New York, NY, USA, 1985.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 716838, 12 pages
http://dx.doi.org/10.1155/2014/716838

Research Article

Towards an Optimal Energy Consumption for Unattended Mobile Sensor Networks through Autonomous Sensor Redeployment

Jian Chen,[1,2] Jie Jia,[2] Yingyou Wen,[1,2] and Dazhe Zhao[1,2]

[1] *Key Laboratory of Medical Image Computing of Ministry of Education, Northeastern University, Shenyang 110819, China*
[2] *School of Information Science & Engineering, Northeastern University, Shenyang 110819, China*

Correspondence should be addressed to Jie Jia; jiajie@ise.neu.edu.cn

Received 9 December 2013; Accepted 12 January 2014; Published 22 April 2014

Academic Editors: H. R. Karimi, Z. Yu, and W. Zhang

Energy hole is an inherent problem caused by heavier traffic loads of sensor nodes nearer the sink because of more frequent data transmission, which is strongly dependent on the topology induced by the sensor deployment. In this paper, we propose an autonomous sensor redeployment algorithm to balance energy consumption and mitigate energy hole for unattended mobile sensor networks. First, with the target area divided into several equal width coronas, we present a mathematical problem modeling sensor node layout as well as transmission pattern to maximize network coverage and reduce communication cost. And then, by calculating the optimal node density for each corona to avoid energy hole, a fully distributed movement algorithm is proposed, which can achieve an optimal distribution quickly only by pushing or pulling its one-hop neighbors. The simulation results demonstrate that our algorithm achieves a much smaller average moving distance and a much longer network lifetime than existing algorithms and can eliminate the energy hole problem effectively.

1. Introduction

A typical wireless sensor network (WSN) is composed of hundreds of sensor nodes reporting their data to the information collector, referred to as the sink node. Sensor nodes are usually of low cost and low power, having limited sensing, computing, and communication capabilities. In recent years, with the rapid progress in advanced VLSI and radio frequency (RF) technologies, WSNs have attracted lots of interest due to their potential use in various applications such as military surveillance, target tracking, emergency navigation, and large scale systems [1, 2].

Sensor deployment is an important issue in designing a WSN since it affects the communication cost, detection capability, coverage, and connectivity [3]. In general, the resource-limited sensor nodes are dropped from airborne vehicles for remote surveying of unattended environment without a preconfigured infrastructure. Those sensor nodes are typically left unattended and remain static after initial deployment. Sometimes, establishing such a fixed sensor network over a hostile or dangerous area to provide complete coverage could be a daunting task. Thus, there exists an urgent need for exploiting sensor mobility in WSNs to improve network performance.

As sensor nodes are usually battery driven, they can survive for only a limited lifetime with nonrenewable batteries. Taking it one step further, the limited constraints of the sensor nodes restrict the use of high complexity algorithms and protocols. How to balance energy consumption is one of the fundamental issues arising in WSN. To address this issue, much work has been done during recent years. Among them, taking advantage of sensor mobility to enhance network lifetime has attracted extensive attention [4–11]. Typically, most of these traditional approaches aimed at having uniform deployment to achieve full coverage using a minimum number of sensor nodes. However, these approaches do not consider the issue of unbalanced energy depletion with distance to a predetermined sink. As the source nodes sent their sensing messages to the sink via a multihop relay tree in

WSN, this many-to-one communication pattern could lead to traffic imbalance among sensor nodes. It has been observed that the closer a sensor is to the sink, the faster its battery exhausts, as time evolves, which will cause an energy hole in the region near the sink [12]. If this happens, no more data could be transmitted to the sink and the network is inactive soon leaving much energy unused, regardless of how many sensor nodes are deployed. Experimental results in [13] show that when the network lifetime ends, nearly 90% of the total initial energy is unused with uniform distribution. Therefore, managing sensor mobility to achieve both optimal energy consumption and full coverage is important to prolong the network lifetime for WSN.

In this paper, we try to solve the energy hole problem by proposing an autonomous coverage-driven sensor redeployment scheme. We first develop the energy hole problem with nonuniform node distribution in WSN theoretically. By importing the energy-aware transmission mechanism and the accessibility condition of energy-balanced depletion in our pervious approach [14], we further propose a fully distributed density control scheme in different coronas to balance energy depletion for the entire network. The rest of the paper is organized as follows. Section 2 reviews the related literature. Section 3 describes the network model, assumptions, and energy-aware transmission mechanism. A novel sensor redeployment strategy is proposed in Section 4. Section 5 presents simulation results for our algorithms, and Section 6 concludes our paper.

2. Related Work

More recently, there has been growing interest in optimizing the sensor movement to maintain full coverage and prolong the network lifetime for mobile sensor networks. In [6], the authors proposed a potential field-based deployment algorithm, in which all the nodes explore from a compact region and fill the maximum working area in a way similar to the particles in the microworld [15]. In [7], the authors assumed that there were virtually attractive and repulsive forces among sensors. Using these virtual forces, mobile sensor nodes spread throughout the target area with a uniform distribution such that the network coverage rate is maximized. In [8], the authors proposed a Voronoi diagram-based distribution model, in which each sensor iteratively calculates its Voronoi polygon to detect the coverage holes and moves to a better position to improve the coverage rate. In [9], the authors proposed three independent algorithms (VEC, VOR, and MiniMax), by pushing or pulling nodes to cover the gaps based on virtual forces. These three algorithms have comparable performance in a bounded area, whereas only VEC algorithm can be used in both unbounded and bounded areas. In [10], the authors investigate how to move sensors to some locations while still preserving the degree of coverage under partially controlled placement. In [11], the sensing field was split into grids, and the sensors moved from high-density grids to low-density ones such that the densities remain constant. Overall, most of these traditional algorithms intended to redistribute sensor nodes uniformly and thus to maximize coverage rate, minimize coverage overlap or gap, and reduce the network cost for WSN. However, as the uniform distribution has unbalanced communication traffic, these approaches will cause the network lifetime to end prematurely with a great amount of energy unused.

In [12], the authors investigated the problem of uneven energy consumption in many-to-one WSNs for the first time. Further, they proposed several approaches to mitigate this problem and inferred that simply increasing the number of nodes under a uniform distribution could not prolong the network lifetime [16]. In [17], the authors focused on the nonuniform energy distribution among sensors. It was concluded that the higher the workload was, the higher the initial level of energy would be set. Although such a strategy seems to be promising, its application may be so difficult that it is inconvenient in producing and deploying sensor nodes. Sink mobility was also introduced to tackle the energy hole problem [18–20]. The authors in [18] proposed an autonomous sink movement strategy in which mobile sinks move toward half-quadrant zones with abundant energy. Similarly, the authors in [19] drew a conclusion that utilizing mobile sink could prolong network lifetime 3.48 times at most compared to the case with static sinks (when the mobile sink moves around a hexagonal network perimeter and stops at the six corners). They also proposed a sink movement algorithm to keep network connectivity so as to prolong network lifetime. The authors in [20] tried to incorporate both static and mobile sinks to improve the network performance of WSN, in which the static sinks are located at the center of the target area and the mobile sink moves around the network perimeter. Each time when the mobile sink stops at a new location, it only needs to broadcast the location updates messages to a subset of sensor nodes. Despite the fact that mobile sinks would bring some advantages to WSN, some new issues appeared such as the movement control policy, the energy-aware routing protocols, and the efficient path planning. Unfortunately, these issues are very complicated and not readily solvable [21, 22].

Energy-aware sensor redistribution was also proposed to mitigate or avoid the energy hole problem in WSNs. In [23], the authors proposed an efficient node placement, topology control, and a scheduling protocol of the MAC layer to prolong the network lifetime for a grid-based WSN. In [24], the authors developed a mixed-integer linear programming model for determining the locations of sensors and sinks, schedules, and sensor-to-sink data flow routes. Further, they proposed a heuristic algorithm to maximize the network lifetime. In [25], the authors tried to solve the energy provisioning and relay node placement problems simultaneously in a two-tiered WSN. At first, the low-cost sensors sense the surrounding area and forward the sensing data to their cluster head, and then the cluster head forwards the data to the sink. Further, they proposed a heuristic approach to solve the mixed-integer nonlinear programming problem. In [26], the authors investigated sensor self-deployment problem, by constructing focused coverage around a point of interest. The authors in [27] explored variable node distribution density to mitigate the effects of the uneven energy depletion. The authors in [28] proposed a nonuniform distribution

algorithm to solve the energy hole problem in a corona-based WSN. With their theoretical analysis, when all the sensors have a constant data acquisition rate, the balanced energy depletion among the whole network is impossible. However, their traffic pattern may not be true for highly dense WSN and the uneven energy depletion still exists between the outermost corona and the inner coronas. In fact, we can prove that balanced energy depletion is achievable with the extra help of energy-aware transmission mechanism in this paper. In [29], the authors investigated the sink-hole problem in duty-cycled connected k-covered WSNs, where each point is covered by at least k sensor. In [30], the ring-based relays are also used to eliminate energy hole of a WSN; however, how to form such a relay is not discussed in their approach. In our pervious approach [14], we have proposed the centralized algorithm to form an energy-balanced distribution for WSN; however, the exchange of global location information during dynamic sensor movement would put a heavy traffic burden on the network. In [31], the authors proposed a transmission range adjustment approach to tackle the unbalanced energy depletion. However, searching for the optimal transmission ranges among all the coronas is a NP-complete problem.

Overall, though most of the algorithms discussed above intended to maximize coverage rate, minimize deployment density, and eliminate the energy hole, they did not answer a fundamental question in sensor redeployment: what type of node layout and communication pattern that could provide the maximum coverage with the smallest overlap and gap and guarantee that all the working sensors die simultaneously with nearly zero residual energy? We will deal with this issue in the next section.

3. Network Model and Assumptions

In this section, we will present our network model and basic assumptions. Assume that a set of N homogeneous sensors with the same initial energy ε is deployed in a circular area with radius d to monitor some physical phenomena and an unlimited amount of energy is set for the sink node. We refer to the set of deployed sensors as $S = \{s_1, s_2, \ldots, s_N\}$, each of which has an ID, a fixed transmission range R_c, a fixed sensing range R_s, and is aware of its location. The only sink node is placed at the center of the circle. We divide the area into n adjacent coronas with the equivalent width of R_c, and the ith corona is denoted by C_i. Obviously, the corona C_i is composed of nodes whose distances to the sink node are between $(i-1) * R_c$ and $i * R_c$.

In this paper, periodic data gathering monitoring is considered, where the network is working in rounds. Each round is further divided into two phases: the node redistribution phase and stability monitoring phase. We will provide a detailed description of the first stage issues in the next section. During the second phase, each working node should send their sensing messages to the sink node per unit time via multihop communication. For the theoretical analysis, we use a simplified power consumption model and an ideal MAC layer with no collisions and retransmissions. The initial energy of each sensor is set as $\varepsilon > 0$, and the sink has no energy limitation. We further assume that each sensor consumes e_1 units of energy when sending one bit, while it depletes e_2 units of energy when receiving one bit, where $e_1 > e_2 > 0$.

When using traditional transmission mechanism, the redundant sensing messages for the same area will be retransmitted by more than one sensor, consuming a considerable amount of energy [26]. In order to save energy, an energy aware data transmission mechanism proposed in our pervious approach [14] is imported. In this mechanism, we first need to build Voronoi graph for each sensor node to determine its own sensing range. After the establishment of the Voronoi graph, the sensing message for each pixel is sent only once. Based on the network model, we can conclude that sensors belonging to corona $\{C_i \mid i \neq n\}$ will forward both the data generated by themselves and the data produced by coronas $\{C_j \mid (i+1) \leq j \leq n\}$. Obviously, the sensors located in the outermost corona C_n do not need to forward any data. Assume that the sensors in each corona are distributed uniformly and there is no data aggregation at any forwarding nodes. Define the number of sensors deployed in corona C_i as N_i and the number of pixels in C_i as A_i. When the energy-aware data transmission mechanism is applied, the number of messages for corona C_i to receive and forward is $(A_{i+1} + A_{i+2} + \cdots + A_n)$ and $(A_i + A_{i+1} + \cdots + A_n)$. As sensing messages are transmitted per working round, the average energy consumption per working round of sensors in corona C_i is

$$\overline{E}_i = \frac{e_1}{\rho_i} + \frac{A_{i+1} + A_{i+2} + \cdots + A_n}{A_i \cdot \rho_i}\left(e_1 + e_2\right), \quad 1 \leq i \leq n-1, \tag{1}$$

where ρ_i is the node density of corona C_i.

As the sensors in corona C_n only need to send their own sensing messages, the energy depletion of sensors in corona C_n can be calculated as follows:

$$\overline{E}_n = \frac{A_n \cdot e_1}{N_n} = \frac{e_1}{\rho_n}. \tag{2}$$

Definition 1. The optimal energy consumption state means that all the sensors in the network deplete their energy in the constant ratio; namely, all the sensor nodes have the same lifetime which is identical with the corresponding network lifetime. In particular, if the optimal energy depletion is achieved, there is no energy wasted and the network lifetime can be given by

$$\frac{\varepsilon}{\overline{E}_1} = \frac{\varepsilon}{\overline{E}_2} = \cdots = \frac{\varepsilon}{\overline{E}_i} = \cdots = \frac{\varepsilon}{\overline{E}_n}. \tag{3}$$

Theorem 2. *Optimal energy depletion is possible, if all the working nodes take the energy-aware transmission mechanism, and the node density ρ_i in corona C_i satisfies [14]*

$$\rho_i = \rho_n \cdot \left[1 + \frac{\left(n^2 - i^2\right) \cdot \left(e_1 + e_2\right)}{(2i-1) \cdot e_1}\right], \quad \rho_1 \geq \rho_2 \geq \cdots \geq \rho_n. \tag{4}$$

Proof. Without loss of generality, suppose that (4) is true: so (1) can be described as follows:

$$\begin{aligned}\overline{E}_i &= \frac{A_i \cdot e_1 + \sum_{k=i+1}^{n} A_k \cdot (e_1 + e_2)}{A_i \cdot \rho_i} \\ &= \frac{[A_i \cdot e_1 + \sum_{k=i+1}^{n} A_k \cdot (e_1 + e_2)] \cdot (2i-1) \cdot e_1}{A_i \cdot \rho_n \cdot [(2i-1) \cdot e_1 + (n^2 - i^2) \cdot (e_1 + e_2)]}.\end{aligned} \tag{5}$$

As $A_i = \pi {R_c}^2 \cdot (2i-1)$, we have

$$\overline{E}_i = \frac{e_1}{\rho_n} = \overline{E}_n. \tag{6}$$

Since $d\rho_i/di = (-2i/(2i-1)) - (2(n^2 - i^2)/(2i-1)^2) < 0$ is a permanent establishment, we can draw a conclusion that $\rho_1 \geq \rho_2 \geq \cdots \geq \rho_n$. This completes the proof of Theorem 2. □

Theorem 2 shows that, in a circular monitored area, based on the energy-aware transmission mechanism, if the sensors in each corona obey a uniform distribution and the node density meets a certain condition, the optimal energy depletion of the whole network can be achieved. Besides, the node density ρ_i of corona C_i only depends on ρ_n and the corona number i.

Theorem 3. *The lifetime comparison of energy–balanced redeployment with traditional uniform approaches is ρ_1/ρ_N.*

Proof. Suppose that these two schemes run with the same initial conditions. The node density of each corona obeys (4) in our proposed scheme, while, in the uniform distribution, the density ρ_i is equal to ρ_n. As the innermost corona C_1 needs to forward all the sensing messages in the whole network, it consumes the most energy. Therefore, the maximum network lifetime under uniform distribution is determined by the survival time C_1. The network lifetime can be calculated as

$$\frac{\varepsilon}{\overline{E}_1'} = \frac{\rho_n \cdot \varepsilon}{e_1 + (n^2 - 1) \cdot (e_1 + e_2)}, \tag{7}$$

where $\overline{E}_1'$ is the average energy depletion of C_1 per unit time under uniform distribution. Further, we can get the average energy depletion in C_1 under optimal energy consumption conditions as

$$\overline{E}_1 = \cdots \overline{E}_i = \cdots = \overline{E}_n = \frac{e_1}{\rho_n}. \tag{8}$$

Thus the lifetime enhancement is

$$\frac{\varepsilon/\overline{E}_1}{\varepsilon/\overline{E}_1'} = \frac{\left(e_1 + \left(n^2 - 1\right) \cdot (e_1 + e_2)\right)/\rho_n}{e_1/\rho_n} = \frac{\rho_1}{\rho_n} > 1. \tag{9}$$

This completes the proof of Theorem 3. □

Therefore, compared with traditional uniform distribution strategy, the network lifetime can raise as much as ρ_1/ρ_n times by using optimal energy depletion distribution.

4. Optimization of Sensor Redeployment

In this section, we first introduce equivalent sensing radius. Then we develop the uniform sensor distribution for corona-shaped area. Further, we propose a novel autonomous sensor redeployment approach to balance energy depletion.

4.1. Redistribution under Equivalent Sensing Radius

Definition 4. Equivalent sensing radius is defined as the sensing radius when the given distribution density ρ_i is the lowest one to maintain the full coverage for the target area with size S.

According to [9], the equivalent sensing radius and the distribution density ρ_i satisfy the following:

$$R_i = \sqrt{\frac{2}{\sqrt{27} \cdot \rho_i}}. \tag{10}$$

Theorem 5. *Optimal energy depletion can be achieved, where R_i satisfies the following:*

$$\begin{aligned}R_i &= R_s \sqrt{\frac{(2i-1) \cdot e_1}{(2i-1) \cdot e_1 + (n^2 - i^2) \cdot (e_1 + e_2)}}, \\ R_1 &\leq R_2 \leq \cdots \leq R_n.\end{aligned} \tag{11}$$

Proof. According to the definition of equivalent sensing radius, we can combine it with the energy balance condition. Thus we have

$$\rho_i = \frac{2}{\sqrt{27} \cdot {R_s}^2} \cdot \left[1 + \frac{\left(n^2 - i^2\right) \cdot (e_1 + e_2)}{(2i-1) \cdot e_1}\right]. \tag{12}$$

After simple transformation, we can get

$$R_i = R_s \cdot \sqrt{\frac{(2i-1) \cdot e_1}{(2i-1) \cdot e_1 + (n^2 - i^2) \cdot (e_1 + e_2)}}. \tag{13}$$

This concludes the proof of Theorem 5. □

Since the equivalent sensing radius is only determined by corona number i, the redeployment of sensors in corona C_i is similar to the traditional uniform distribution problem. Define the equivalent sensing radius of corona C_i as R_i and the desired number of deployed sensor for C_i as $\rho_i \cdot S_i$. Note that the boundary effects cannot be ignored simply. In order to cover corona C_i uniformly, the optimal sensor distribution should satisfy the following conditions.

(1) If $R_i \geq R_c/2$, the optimal deployment can be achieved if all the sensors are uniformly lying on the middle line of corona C_i, and the angle between any two adjacent nodes is $\alpha_{C_i} = 2\pi/(\rho_i \cdot S_i)$.

(2) If $R_i < R_c/2$, as the single ring distribution will cause coverage gaps, the optimal sensor redeployment is equivalent to a multiring distribution. Define the number of rings for C_i as N_i, calculated as $N_i = \lceil R_c/2R_i \rceil$. When multiring uniform distribution is

achieved for corona C_i, the number of sensors distributed in k_{th} ring C_i^k is $\rho_i \cdot S_{C_i^k}$, and the angle between any two neighbours in C_i^k is $\alpha_{C_i^k} = 2\pi/(\rho_i \cdot S_{C_i^k})$, where $S_{C_i^k}$ is the area of ring C_i^k. Assuming that all the rings have the same width, $d = R_c/N_i$, such that $S_{C_i^k}$ can be calculated as

$$S_{C_i^k} = \pi\left[(i-1)\cdot R_c + k\cdot d\right]^2 - \pi\left[(i-1)\cdot R_c + (k-1)\cdot d\right]^2. \tag{14}$$

Figure 1 shows the optimal sensor redeployment with different equivalent sensing radii R_i. When $R_i = R_c/2$, the sensor nodes in C_i obey a single-ring uniform distribution, and the angle between any two neighbours is $\alpha = 2\pi/(\rho_i \cdot S_i)$, as shown in Figure 1(a). When $\lceil R_c/2R_i \rceil = 2$, the sensor nodes in corona C_i obey a two-ring uniform distribution; the adjacent angles of ring C_i^1 and C_i^2 are $\alpha_{C_i^1} = 2\pi/(\rho_i \cdot S_{C_i^1})$ and $\alpha_{C_i^2} = 2\pi/(\rho_i \cdot S_{C_i^2})$, respectively.

Therefore, the optimal sensor redeployment to balance energy depletion can be transformed into a uniform distribution problem with given deployment density and equivalent sensing radius. The novel sensor redeployment algorithm mainly contains two parts: (1) sensor redeployment control among the coronas to regulate the number of nodes for each corona and (2) sensor redeployment control inside coronas to guarantee that each corona achieves the given node density.

4.2. Distributed Sensor Redeployment among Coronas. As sensors are randomly deployed in the target area, the deployment uncertainty may cause the number of deployed nodes to be more or fewer than the corona really needs. Movement control of sensor nodes will satisfy the desired deployment density for each corona and its rings. To avoid consuming much energy during the moving process, the nodes are only allowed to move to the adjacent coronas. After the needed number of sensors is achieved in each corona, a certain number of sensors should move between different rings to achieve uniform distribution. Define the number of sensors deployed in corona C_i as D_i and the desired number of sensors in C_i as ED_i. Sensor redeployment control among coronas can be conducted in Algorithm 1.

4.3. Distributed Sensor Redeployment inside Coronas. Sensor redeployment inside coronas mainly focuses on how to redistribute the nodes locating on the median line of these rings in each corona to a perfect layout. Here, it is executed simply with its neighbour nodes. The straightforward idea is to adjust the angle between any two neighbours to α. However, as each node may have numerous neighbours, the nearest neighbour is taken in the current working round. Taking Figure 2 as an example, assume that node m is the nearest neighbour of node k, to form an optimal layout and node k should move to m_{opt1} or m_{opt2}, both of which are the best locations of node m. In our algorithm, node k moves straight to the nearer location m_{opt1}, thus saving energy. If some other sensors q existed in this best location, to minimize the total movement among all the sensors, q should move with a small deviation ε along the same orientation to node m. The whole process of sensor redeployment inside coronas is shown in Algorithm 2.

Theorem 6. *The sensor redeployment algorithm ends in finite rounds and can achieve uniform layout in the ring.*

Proof. As long as the angle between any two nearest neighbours is not equal to α, the total covered area with uniform distribution will keep on growing. It suffices to prove that, once the algorithm has reached the stable state when the angle between any two neighbours is equal to α, the sensors locating on the ring will be fixed.

Let $S = (x_1, x_2, \ldots, x_n)$ denote the set of sensors distributed in the ring, whereas $\theta_i \in (0, 2\pi]$ is azimuth between the lines ox_i and ox. After reordering these sensors according to θ_i, we rename S as $(\theta_1, \theta_2, \ldots, \theta_n)$. Define β_i as the angle of two adjacent nodes in S. To ensure that β_i does not exceed α, it needs to satisfy the following:

$$\beta_i = \begin{cases} \theta_{i+1} - \theta_i & \text{if } (\theta_{i+1} - \theta_i) \le \alpha \\ 0 & (\theta_{i+1} - \theta_i) > \alpha. \end{cases} \tag{15}$$

Thus the total covered angle of S can be calculated as

$$f(S) = \sum_{i=1}^{n-1} \beta_i. \tag{16}$$

If all the sensor nodes are distributed in an optimal layout, we can get the upper bound for $f(S)$ as $2\pi - \alpha$.

Denote the angle between θ_i and θ_{i+1} as β_i. We now consider the changes of β_i in one working round. If $\beta_i < \alpha$, the following three cases may occur.

(1) If β_i increases to α with no sensors coinciding, the new achieved distribution can be expressed as $S' = (\theta_1, \theta_2, \ldots, \theta_i, \theta'_{i+1}, \ldots, \theta_n)$, as shown in Figure 3(a). Since $\beta'_i > \beta_i$, we have $f(S') > f(S)$.

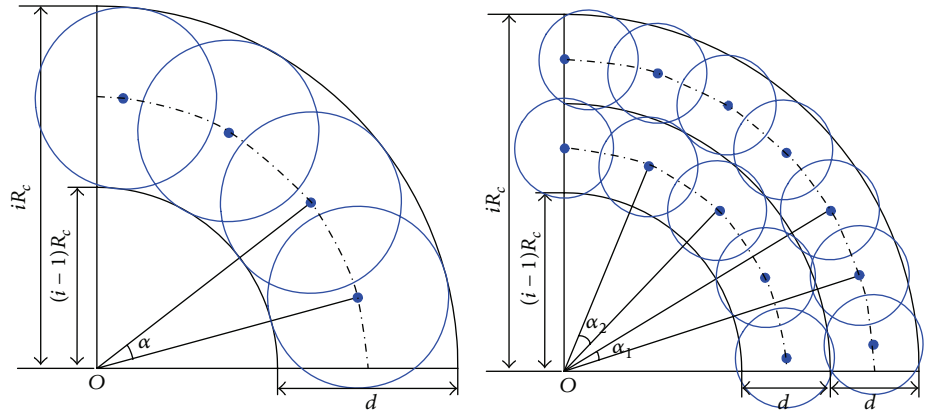
(2) If β_i increases to α with some sensors coinciding, the coincided sensors would move with a small deviation ε in our algorithm. Assume that the new achieved distribution is $S' = (\theta_1, \theta_2, \ldots, \theta_i, \theta'_{i+1}, \theta'_{i+2}, \ldots, \theta_n)$, as shown in Figure 3(b). Therefore, $f(S')$ can be calculated as

$$f(S') = \sum_{j=1}^{i-1}\beta_j + (\theta'_{i+1} - \theta_i) + (\theta'_{i+2} - \theta'_{i+1}) + \sum_{j=i+2}^{n}\beta_j = \sum_{j=1}^{i-1}\beta_j + \alpha + \varepsilon + \sum_{j=i+2}^{n}\beta_j, \tag{17}$$

while

$$f(S) = \sum_{j=1}^{i-1}\beta_j + (\theta_{i+1} - \theta_i) + (\theta_{i+2} - \theta_{i+1}) + \sum_{j=i+2}^{n}\beta_j = \sum_{j=1}^{i-1}\beta_j + \alpha + \sum_{j=i+2}^{n}\beta_j. \tag{18}$$

Thus we have $f(S') > f(S)$.



(a) (b)

Figure 1: Optimal sensor redeployment with different equivalent sensing radii R_i. (a) $R_i = R_c/2$: uniform distribution in single ring; (b) $\lceil R_c/2R_i \rceil = 2$: uniform sensor distribution in two rings.

For each corona C_i, $i \in (n, n-1, \ldots, 1)$
  If $D_i > ED_i$
    Select $D_i - ED_i$ nodes close to C_{i-1} from C_i;
    Move these nodes straight to C_{i-1} and update $D_{i-1} = D_{i-1} + (D_i - ED_i)$;
    For each ring C_i^k, $k \in (1, 2, \ldots, N_i)$
      Move $\rho_i \cdot S_{C_i^k}$ nodes straight to the middle line of C_i^k
  Elseif $D_i \leq ED_i$
    Select $ED_i - D_i$ nodes close to C_i from C_{i-1};
    Move these nodes straight to C_i and update $D_{i-1} = D_{i-1} - (ED_i - D_i)$;
    For each ring C_i^k, $k \in (1, 2, \ldots, N_i)$
      Move $\rho_i \cdot S_{C_i^k}$ nodes straight to the middle line of C_i^k.

Algorithm 1

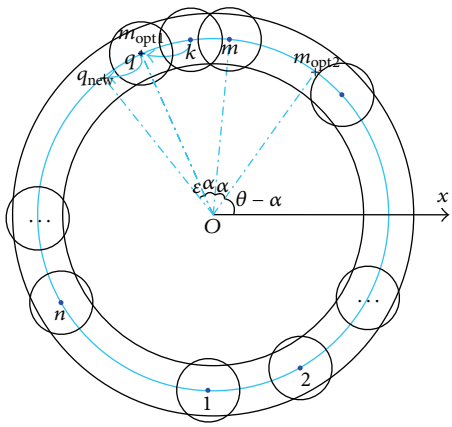


Figure 2: Sensor redeployment in the ring.

(3) If β_i increases to α with some sensors crossing, the new achieved distribution can be denoted as $S' = (\theta_1, \theta_2, \ldots, \theta'_{i-1}, \theta'_i, \theta'_{i+1}, \ldots, \theta_n)$, as shown in Figure 3(c). Since this new generated distribution is

For each round
For each node k in ring C_i^j of corona C_i
  Find its nearest neighbor m;
  Calculate the angle θ between the line om and ox;
  Calculate node m's best location m_{opt1} and m_{opt2}
    Let $\theta_1 = \theta + \alpha; \theta_2 = \theta - \alpha$;
      $x_{m_{opt1}} = \cos\theta_1 \cdot d$; $y_{m_{opt1}} = \sin\theta_1 \cdot d$
      $x_{m_{opt2}} = \cos\theta_2 \cdot d$; $y_{m_{opt2}} = \sin\theta_2 \cdot d$
  If distance (m_{opt1}, k) < distance (m_{opt2}, k)
    Move node k to location $(x_{m_{opt1}}, y_{m_{opt1}})$;
    While there exists node q with coordinate $(x_{m_{opt1}}, y_{m_{opt1}})$
      $x_{new} = \cos(\theta_1 - \varepsilon) \cdot d$
      $y_{new} = \sin(\theta_1 - \varepsilon) \cdot d$
      Move node q to location q_{new} (x_{new}, y_{new});
  Else move node k to location $(x_{m_{opt2}}, y_{m_{opt2}})$;
    While there exists node q with coordinate $(x_{m_{opt2}}, y_{m_{opt2}})$
      $x_{new} = \cos(\theta_2 + \varepsilon) \cdot d$
      $y_{new} = \sin(\theta_2 + \varepsilon) \cdot d$
      Move node q to location q_{new} (x_{new}, y_{new});

Algorithm 2

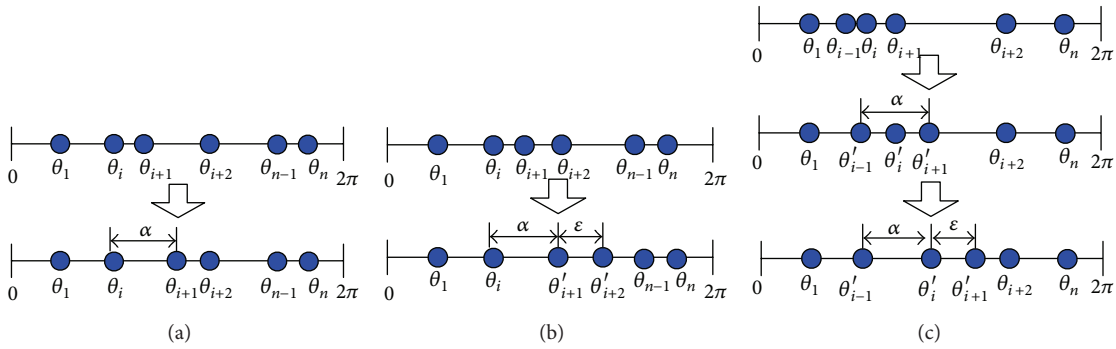


Figure 3: Illustration of sensor redeployment in one working round.

similar to case (2), we can draw the same conclusion of $f(S') > f(S)$, when θ'_i becomes the next operating node.

Clearly, in all the cases, $f(S)$ is an increasing function on rounds $(1, \ldots, n)$. After a finite number of rounds, $f(S)$ would achieve its upper bound. As a result, all the sensors will be redeployed uniformly on the middle line of the ring with the given density. This completes the proof of Theorem 6. □

5. Performance Evaluation

In this section, we will present the simulation results of our algorithm for both random and Gaussian deployment models. Three metrics, including coverage rate, the average moving distance, and the network survival lifetime, are imported to evaluate the performance of the algorithm.

First, a random deployment is considered in the circular area of radius 100 with 627 potential sensors. We assume that all the sensors are homogeneous, that is, having the same initial energy reserve ε of 10000 J, the same transmission radius R_c of 25 m, and the same sensing radius R_s of 9 m, respectively. The values of e_1 and e_2 are given as follows: $e_1 = 0.5/10^3$ J/bit and $e_2 = 0.25/10^3$ J/bit. The total number of working rounds for sensor node i is determined by $\varepsilon/(L \cdot M_i \cdot e_1 + L \cdot N_i \cdot (e_1 + e_2))$, where M_i is the total number of pixels covered by i, N_i is the total number of messages forwarded by i, and L is the length of each sensing pixel, $L = 1000$ bits.

The sensing data forwarding strategy is similar to [27]. As the nodes obey an approximate uniform distribution in the corona after sensor redeployment, any node in corona C_i can directly communicate with almost $\rho_{i-1} \cdot A_{i-1}/\rho_i \cdot A_i$ nodes in C_{i-1}. Among these $\rho_{i-1} \cdot A_{i-1}/\rho_i \cdot A_i$ candidate nodes, the node with most residual energy will be selected as the forwarding one.

Figure 4 depicts the sensor distribution gained in different rounds using random deployment model. The target area is divided into four coronas marked from C_1 to C_4, and the equivalent sensing radius from C_1 to C_4 can be calculated as 1.86, 3.4, 5.11, and 9, respectively, as shown in Figure 4(a). From the simulation results, we can make the observation that our proposed algorithm can converge to the global optimal distribution effectively. What is more, the layout shown in Figure 4(d) is the optimal uniform sensor distribution with the given density, which achieves the coverage rate of 97.6%. And the number of sensors redeployed from corona C_1 to C_4 is 220, 196, 145, and 66, which approaches to the accessibility condition of optimal energy depletion.

The second assumption examined is the Gaussian distribution, and the simulation environment is the same as the first experiment. In this distribution, each sensor's coordinate (x, y) obeys a two-dimensional normal Gaussian function:

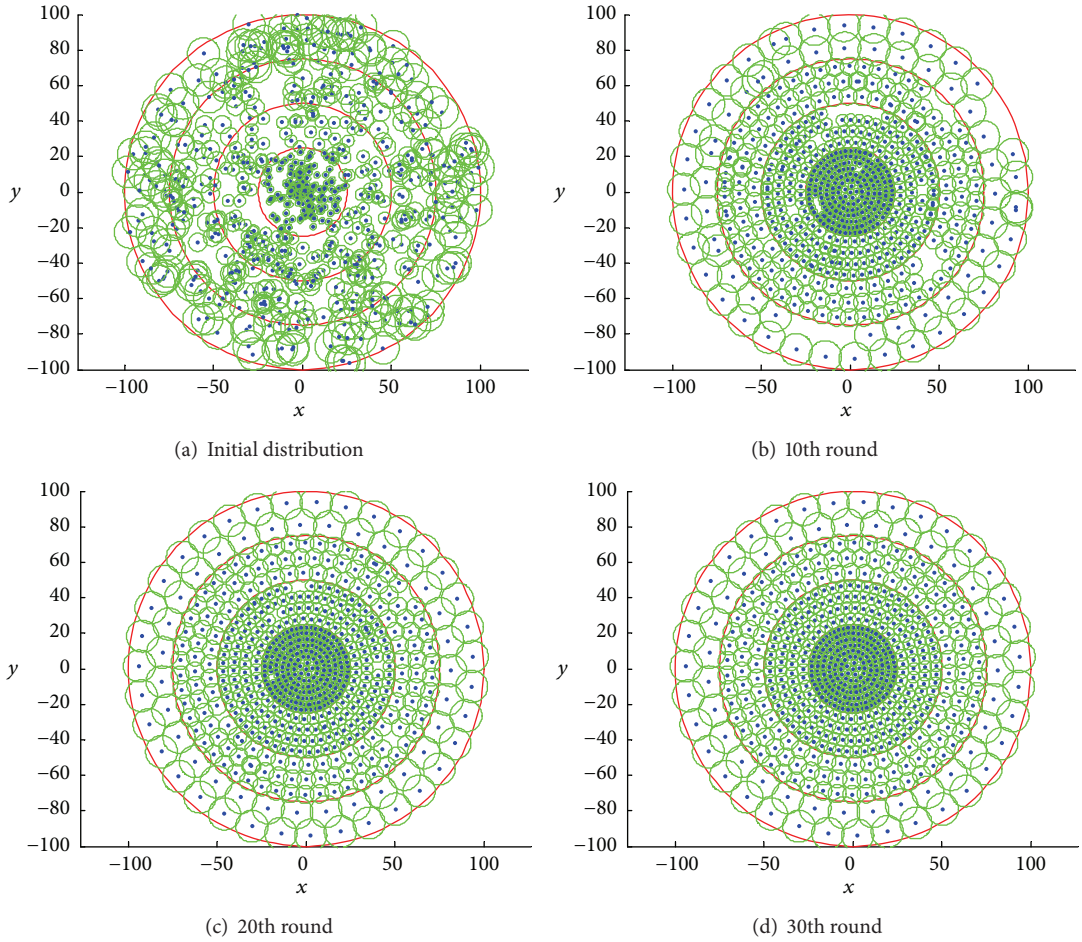
$$f(x, y) = \frac{1}{2\pi\sigma^2} e^{-[x^2+y^2]/2\sigma^2}, \tag{19}$$

where σ represents the standard deviation of Gaussian distribution. We set σ equal to the communication radius R_c to ensure that all the sensors will be deployed in the target area.

Figure 5(a) shows the sensor distribution obtained in different rounds using Gaussian deployment model. From the simulation results, we can draw a conclusion that the total coverage enlarges round by round, as observed from Figure 5(b) to Figure 5(d). At the 28th round, the nodes are uniformly distributed with no sensors moving anymore, which indicates that the algorithm can converge to the optimal solution quickly.

Further, we compare the performance of our algorithm with VEC in terms of coverage rate and average moving. Figure 6 shows the average moving distance and coverage rate for VEC [10] and our approach in different numbers of rounds. From the simulation results in Figure 6, we can conclude that (1) the average moving distance under Gaussian deployment model is shorter than under random model, which is mainly due to having more sensors deployed in the central region under Gaussian deployment model; (2) to achieve the desired coverage rate, many more sensors are needed in VEC than in our proposed algorithm, which shows our algorithm has a shorter average moving distance than VEC under the same condition; (3) our algorithm can get a higher coverage rate than VEC, which means that our algorithm is free of the boundary influence and can achieve the optimal sensor distribution effectively.

We further compare the performance of our algorithm with traditional uniform and nonuniform sensor deployment in terms of network lifetime and energy efficiency. In addition, the performance of our energy-aware transmission



(a) Initial distribution

(b) 10th round

(c) 20th round

(d) 30th round

Figure 4: Illustration of network distribution in different rounds using random deployment model.

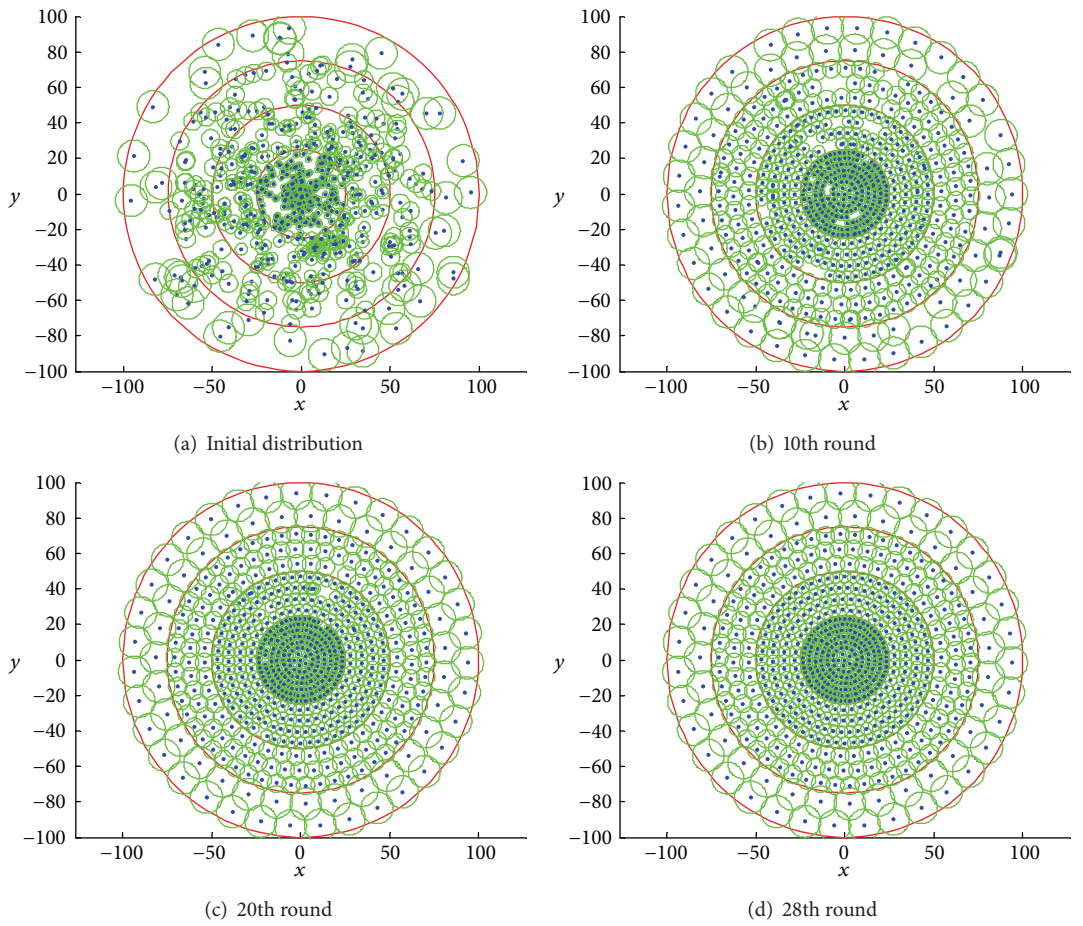
mechanism (E_t) and traditional transmission mechanism (T_t) combined these algorithms is also tested.

Since the node densities for all the coronas are the same in uniform deployment, the number of sensors distributed from C_4 to C_1 is 274, 196, 118, and 39. While fewer nodes are used to cover the outmost corona in the nonuniform deployment, the ratio q is set to 2.12. Therefore, the number of sensors uniformly distributed from corona C_4 to C_1 is 66, 73, 156, and 332, respectively. For our algorithm, to ensure the optimal energy consumption, the number of sensors distributed from corona C_4 to C_1 is 66, 145, 196, and 220.

Figure 7 shows the comparisons of average energy depletion per working round. From the simulation results in Figure 7, we can draw the conclusions that (1) the average energy consumed in a round in E_t is much smaller than in T_t. Because the energy-aware transmission can avoid retransmitting the same sensing data, the total messages transferred in each round are much smaller; (2) since the uniform deployment ignores the traffic imbalance, the innermost corona has the largest traffic load and most energy consumption, as shown in Figure 7(a); (3) the nonuniform deployment combined with P_t scheme can achieve a suboptimal distribution; (4) the energy consumption of the whole working set is almost equal in our approach. Although the nodes in inner coronas do not behave as source but router, their sensing pixels are much smaller than those of the outer coronas.

Figure 8 compares the energy unused ratio of each node when the network terminates. Here, the energy unused ratio refers to the ratio of the residual energy to the initial energy at the end of the network lifetime. From Figure 8, clearly, the uniform scheme leaves an abundant amount of nodal energy in both transmission mechanisms as expected, especially for the sensor nodes in the outermost corona. Although the nonuniform deployment combined with T_t scheme achieves much better energy efficiency, it leaves a considerable amount of unused energy in the outermost corona (almost 50% or so of the initial nodal energy). For the nonuniform deployment combined with P_t, as corona C_3 has the most transmission load and corona C_1 has the largest node density, no doubt, C_1 has the largest energy unused ratio. For our algorithm, as the energy consumption of the whole working set is almost equal, all the sensors exhaust their energy simultaneously when the network terminates, as shown in Figure 7. In fact, the energy unused ratio for most sensors is lower than 1%.

Table 1 compares these algorithms in network lifetime. We can see that the algorithm combined with energy-aware transmission mechanism has a longer network lifetime as



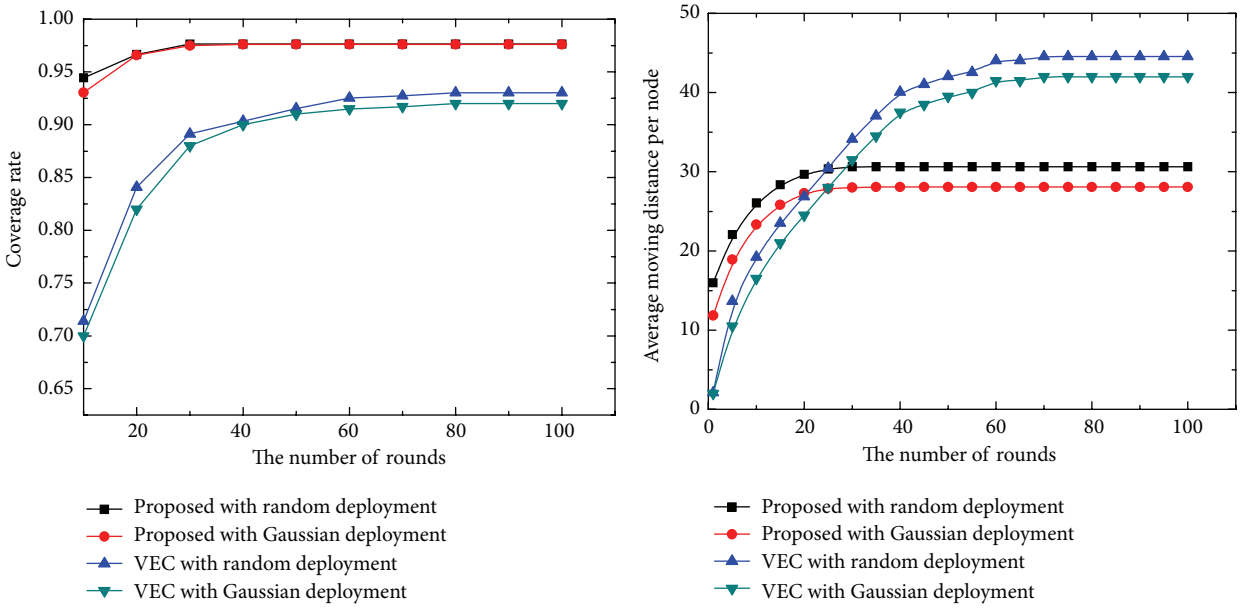
(a) Initial distribution

(b) 10th round

(c) 20th round

(d) 28th round

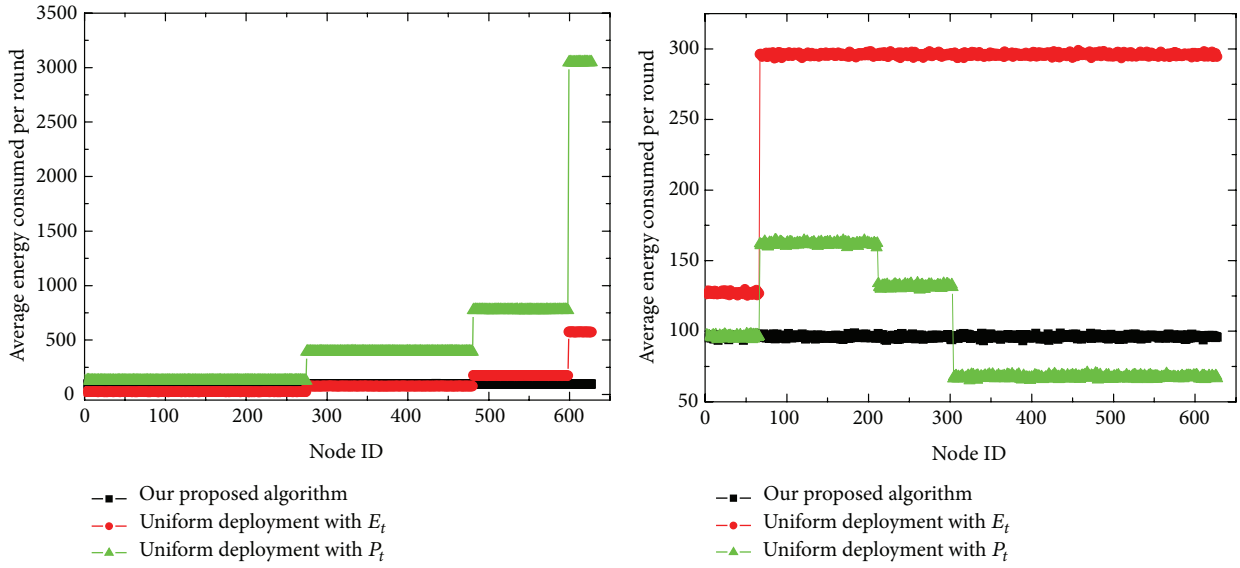
Figure 5: Illustration of network distribution in different rounds using Gaussian deployment model.



(a) Comparisons of coverage rate

(b) Comparisons of moving distance

Figure 6: Comparisons of network performance in different rounds.



(a) Comparison with uniform deployment

(b) Comparison with the nonuniform deployment

Figure 7: Comparison of average energy consumption in a sampling period.

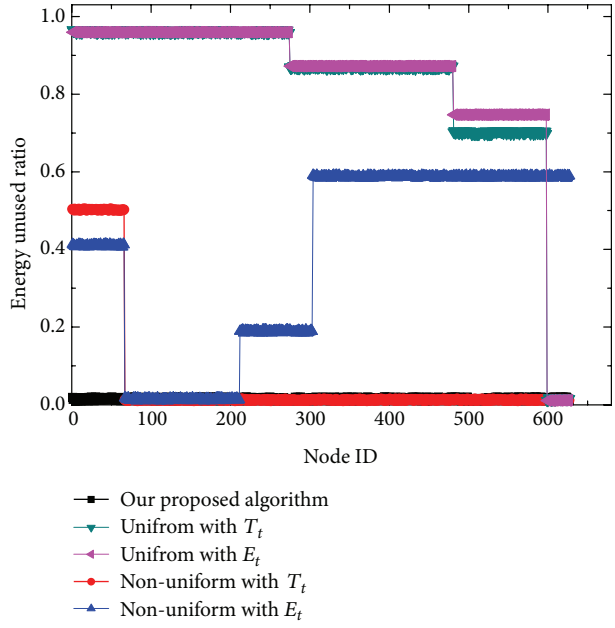


Figure 8: Comparison of energy unused ratio for each node.

expected. Because of the imbalance consumption of energy near the sink, the uniform distribution has the shortest network lifetime. The nonuniform distribution combined with T_t achieved a shorter network lifetime than with E_t, because its average energy consumption is larger than in E_t scheme. Our algorithm can achieve the optimal energy consumption among the whole working nodes, thus having the longest network lifetime.

Table 1: Comparison of the number of working rounds.

Algorithm	The number of working rounds
Uniform deployment with T_t	3
Uniform deployment with E_t	17
Nonuniform deployment with T_t	33
Nonuniform deployment with E_t	61
Our proposed algorithm	104

6. Conclusion and Future Work

In this paper, we have investigated the problem of sensor redeployment to achieve optimal energy depletion and minimize sensor movement. We have given a theoretical analysis on energy consumption using nonuniform distribution strategy. Formally, we have proved that, the optimal energy consumption can be achieved through calculating the node densities for different regions of the target area. As a contribution, we have proposed an autonomous coverage-driven sensor redeployment algorithm to produce an optimal solution, which can maximize the network lifetime and minimize total movement of sensors. In addition, extensive simulation results have been presented to demonstrate the effectiveness of our proposed techniques.It also should be noted that we only consider the two dimensional (2D) case in this paper. As part of our future work, we will design new algorithms for the sensor redeployment problem in 3D space.

Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant no. 61173153 and no. 60903159, the National Science Foundation for Distinguished Young Scholars of China under Grant no. 61225012 and no. 71325002, the Fundamental Research Funds for the Central Universities under Grant nos. N130504007, N110318001, N110204003, N100218001, and N120104001, the China Postdoctoral Science Foundation funded project under Grant no. 20110491508 and no. 2012T50248, the Specialized Research Fund of the Doctoral Program of Higher Education for the Priority Development Areas under Grant no. 20120042130003, and the Specialized Research Fund for the Doctoral Program of Higher Education under Grant no. 20110042110024.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[2] S. Yin, S. Ding, A. Haghani, and H. Hao, "Data-driven monitoring for stochastic systems and its application on batch process," *International Journal of Systems Science*, vol. 44, no. 7, pp. 1366–1376, 2013.

[3] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: a survey," *Ad Hoc Networks*, vol. 6, no. 4, pp. 621–655, 2008.

[4] N. Heo and P. K. Varshney, "Energy-efficient deployment of intelligent mobile sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics A*, vol. 35, no. 1, pp. 78–92, 2005.

[5] G. Tuna, V. C. Gungor, and K. Gulez, "An autonomous wireless sensor network deployment system using mobile robots for human existence detection in case of disasters," *Ad Hoc Networks*, vol. 13, pp. 54–68, 2014.

[6] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: a distributed scalable solution to the area coverage problem," in *Proceedings of the 6th International Conference Distributed Autonomous Robotic Systems (DARS '02)*, pp. 299–308, 2002.

[7] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization based on virtual forces," in *Proceedings of the 22nd Annual Joint Conference on the IEEE Computer and Communications Societies*, pp. 1293–1303, April 2003.

[8] G. Wang, G. Cao, T. La Porta, and W. Zhang, "Sensor relocation in mobile sensor networks," in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05)*, pp. 2302–2312, Miami, Fla, USA, March 2005.

[9] G. Wang, G. Cao, and T. F. La Porta, "Movement-assisted sensor deployment," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 640–652, 2006.

[10] M. Leoncini, G. Resta, and P. Santi, "Partially controlled deployment strategies for wireless sensors," *Ad Hoc Networks*, vol. 7, no. 1, pp. 1–23, 2009.

[11] J. Wu and S. Yang, "SMART: a scan-based movement-assisted sensor deployment method in wireless sensor networks," in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05)*, pp. 2313–2324, Miami, Fla, USA, March 2005.

[12] J. Li and P. Mohapatra, "An analytical model for the energy hole problem in many-to-one sensor networks," in *Proceedings of the 62nd Vehicular Technology Conference Fall (VTC '09)*, pp. 2721–2725, 2005.

[13] J. Lian, K. Naik, and G. B. Agnew, "Data capacity improvement of wireless sensor networks using non-uniform sensor distribution," *International Journal of Distributed Sensor Networks*, vol. 2, no. 2, pp. 121–145, 2006.

[14] J. Jia, X. Wu, J. Chen, and X. Wang, "Exploiting sensor redistribution for eliminating the energy hole problem in mobile sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, article 68, 2012.

[15] G. D. Coughlan, J. E. Dodd, and B. M. Gripaios, *The Ideas of Particle Physics: An Introduction for Scientists*, Cambridge University Press, 1991.

[16] J. Li and P. Mohapatra, "Analytical modeling and mitigation techniques for the energy hole problem in sensor networks," *Pervasive and Mobile Computing*, vol. 3, no. 3, pp. 233–254, 2007.

[17] M. Esseghir, N. Bouabdallah, and G. Pujolle, "Energy provisioning model for maximizing wireless sensor network lifetime," in *Proceedings of the 1st International Global Information Infrastructure Symposium (GIIS '07)*, pp. 80–84, July 2007.

[18] Y. Bi, L. Sun, J. Ma, N. Li, I. A. Khan, and C. Chen, "HUMS: an autonomous moving strategy for mobile sinks in data-gathering sensor networks," *Eurasip Journal on Wireless Communications and Networking*, vol. 2007, Article ID 64574, 2007.

[19] M. Marta and M. Cardei, "Using sink mobility to increase wireless sensor networks lifetime," in *Proceedings of the 9th IEEE International Symposium on Wireless, Mobile and Multimedia Networks (WoWMoM '08)*, pp. 1–10, June 2008.

[20] W. Xiaobing and C. Guihai, "Dual-Sink: using mobile and static sinks for lifetime improvement in wireless sensor networks," in *Proceedings of the 16th International Conference on Computer Communications and Networks (ICCCN '07)*, pp. 1297–1302, August 2007.

[21] J. Heo, J. Hong, and Y. Cho, "EARQ: energy aware routing for real-time and reliable communication in wireless industrial sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 1, pp. 3–11, 2009.

[22] M. Perillo and W. Heinzelman, "An integrated approach to sensor role selection," *IEEE Transactions on Mobile Computing*, vol. 8, no. 5, pp. 709–720, 2009.

[23] C.-Y. Chang and H.-R. Chang, "Energy-aware node placement, topology control and MAC scheduling for wireless sensor networks," *Computer Networks*, vol. 52, no. 11, pp. 2189–2204, 2008.

[24] Y. B. Türkoğullari, N. Aras, I. Kuban Altinel, and C. Ersoy, "An efficient heuristic for placement, scheduling and routing in wireless sensor networks," *Ad Hoc Networks*, vol. 8, no. 6, pp. 654–667, 2010.

[25] Y. T. Hou, Y. Shi, H. D. Sherali, and S. F. Midkiff, "On energy provisioning and relay node placement for wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 4, no. 5, pp. 2579–2590, 2005.

[26] X. Li, H. Frey, N. Santoro, and I. Stojmenovic, "Strictly localized sensor self-deployment for optimal focused coverage," *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1520–1533, 2011.

[27] S. Olariu and I. Stojmenović, "Design guidelines for maximizing lifetime and avoiding energy holes in sensor networks with uniform distribution and uniform reporting," in *Proceedings of*

the 25th IEEE International Conference on Computer Communications (INFOCOM '06), pp. 1–12, April 2006.

[28] X. Wu, G. Chen, and S. K. Das, "Avoiding energy holes in wireless sensor networks with nonuniform node distribution," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 5, pp. 710–720, 2008.

[29] H. Ammari, "Investigating the energy sink-hole problem in connected k-covered wireless sensor networks," *IEEE Transactions on Computers*, vol. 99, 2013.

[30] J. Jia, G. Zhang, X. Wu, J. Chen, X. Wang, and X. Yan, "On the problem of energy balanced relay sensor placement in wireless sensor networks," *International Journal of Distributed Sensor Networks*, Article ID 342904, 2013.

[31] C. Song, M. Liu, J. Cao, Y. Zheng, H. Gong, and G. Chen, "Maximizing network lifetime based on transmission range adjustment in wireless sensor networks," *Computer Communications*, vol. 32, no. 11, pp. 1316–1325, 2009.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 408613, 8 pages
http://dx.doi.org/10.1155/2014/408613

Research Article

In-Band Asymmetry Compensation for Accurate Time/Phase Transport over Optical Transport Network

Sammy Siu, Wen-Hung Tseng, Hsiu-fang Hu, Shinn-Yan Lin, Chia-Shu Liao, and Yi-Liang Lai

Telecommunication Laboratories, Chunghwa Telecom Co., Ltd., No. 99, Dianyan Road, Taoyuan County, Yangmei City 32601, Taiwan

Correspondence should be addressed to Wen-Hung Tseng; whtseng@cht.com.tw

Received 2 January 2014; Accepted 19 March 2014; Published 10 April 2014

Academic Editors: H. R. Karimi, X. Yang, and Z. Yu

The demands of precise time/phase synchronization have been increasing recently due to the next generation of telecommunication synchronization. This paper studies the issues that are relevant to distributing accurate time/phase over optical transport network (OTN). Each node and link can introduce asymmetry, which affects the adequate time/phase accuracy over the networks. In order to achieve better accuracy, protocol level full timing support is used (e.g., Telecom-Boundary clock). Due to chromatic dispersion, the use of different wavelengths consequently causes fiber link delay asymmetry. The analytical result indicates that it introduces significant time error (i.e., phase offset) within 0.3397 ns/km in *C*-band or 0.3943 ns/km in *L*-band depending on the wavelength spacing. With the proposed scheme in this paper, the fiber link delay asymmetry can be compensated relying on the estimated mean fiber link delay by the Telecom-Boundary clock, while the OTN control plane is responsible for processing the fiber link delay asymmetry to determine the asymmetry compensation in the timing chain.

1. Introduction

Precise synchronization of clocks has become an important technique not only for the scientific researches but also for the modern daily life. For many industrial infrastructures, the demands for precise time/phase synchronization have greatly increased recently, for example, communication networks, the smart grid of electric power distribution systems [1], and the practice of providing time stamps for financial networks [2]. Traditional communication network synchronization has relied on the accurate distribution of frequency [3]; evolving wireless networks require the distribution of accurate time/phase based on IEEE1588v2 for long term evolution (LTE) and accurate quality-of-service/service-level-agreement (QoS/SLA) measurements to determine the network health [4, 5].

The primary reference time clocks (PRTCs) location depends on the network that IEEE1588v2 support. Currently, the PRTCs are closer to the end application than than the primary reference clocks (PRCs) for traditional frequency distribution, in order to limit and control time/phase degradation [6]. The core networks will incorporate the accurate time/phase distribution capability into optical transport network (OTN), as addressed in ITU-T Recommendation G.709 [7]. The OTN provides new packet based time/phase distribution service; thus the PRTCs can colocate with the PRCs as shown in Figure 1. This architecture is compatible with PRTC redundancy (e.g., in order to secure the global navigation satellite system (GNSS) failures) and also requires a small number of GNSS receivers. The integrity of transferring accurate time/phase synchronization distribution over OTN in the core network and packet transport network with synchronous ethernet (PTN with Sync-E) in the backhaul networks [8, 9] can simplify network architecture, reduce operational expenditure (OPEX), and make the network easy to maintain.

For accurate time/phase transport over OTN, two options are considered: (1) the use of OTN optical channel data unit-k (ODUk) reserved overhead bytes to transport IEEE1588v2 Sync packets as shown in Figure 2 and (2) the use of optical supervisory channel (OSC) to transport IEEE1588v2 Sync

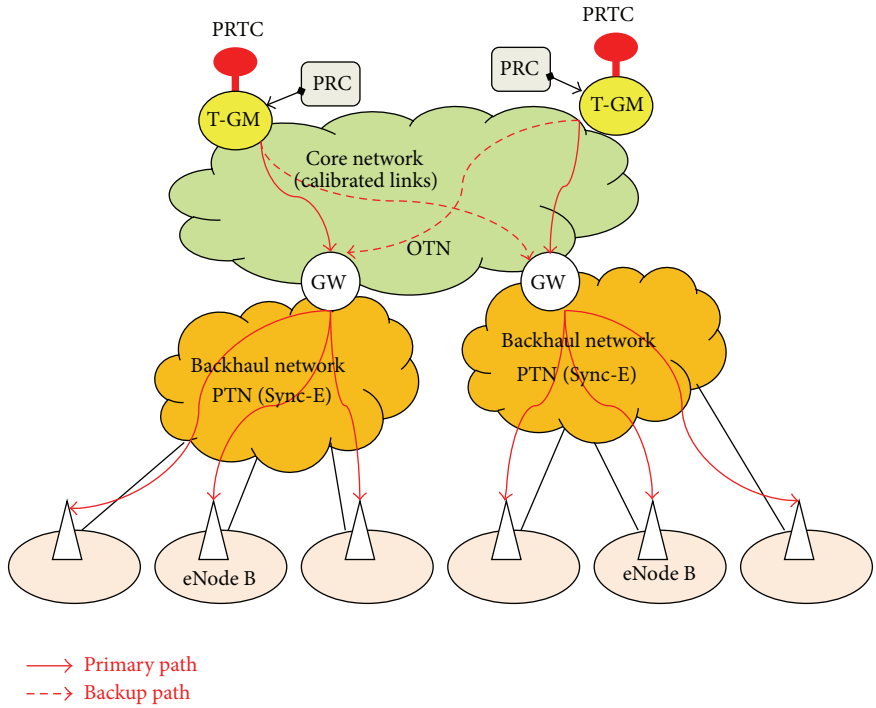


Figure 1: Architecture with centralized PRTC functions colocated with PRC. T-GM: telecom-grand master; GW: gate way.

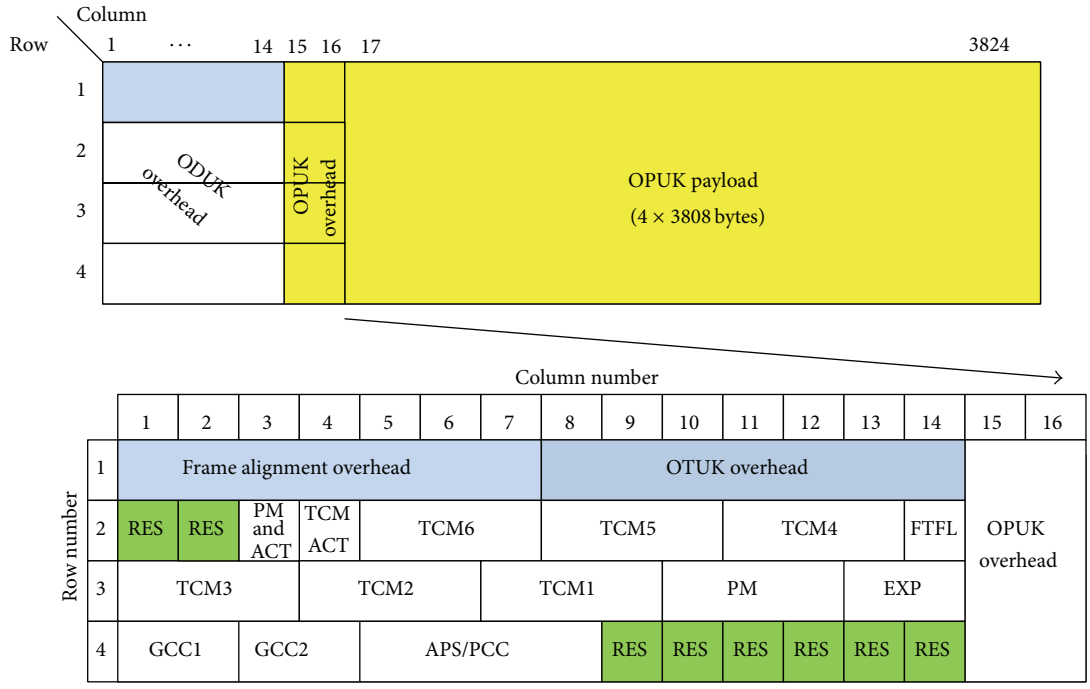


Figure 2: Optical transport network (OTN) frame format (ITU-T G.709).

packets [10]. The former belongs to In-band and the latter belongs to Out-of-band [7].

Nevertheless, each node and link can introduce asymmetry, which affects the adequate time/phase accuracy over the networks. Removal of packet delay variation (PDV) and asymmetry in the OTN nodes by means of IEEE1588v2 support (e.g., T-BC in every node [11, 12]) is analogue to backhaul network (ITU-T G.8275.1). In OTN, the forward and backward paths may not be the same wavelength depending on network configuration or wavelength switching; this will result in fiber link delay asymmetry and should be taken into account.

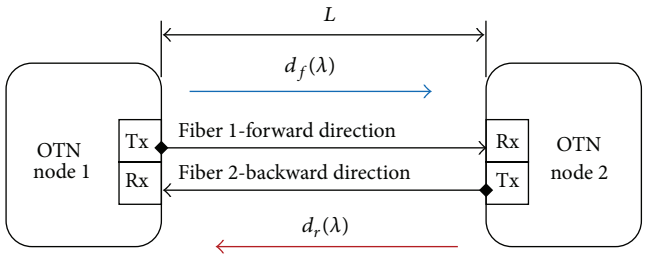


Figure 3: Packet signal transmission over optical fiber links.

In this paper, we focus on the use of OTN overhead to transport Sync packets (In-band). The link delay asymmetry formation is given in Section 2, removal link and node asymmetry based on T-BC mode is given in Section 3, and the link delay asymmetry analysis is given in Section 4.

2. Use of Different Wavelengths

In wavelength division multiplexing (WDM), multiple channels of information carried over the same fiber each using an individual wavelength to increase the transmission capacity as shown in Figure 3. Due to chromatic dispersion, the use of different wavelengths consequently causes fiber link delay asymmetry [13]. Group velocity is given by $v = c/\eta$, where c is speed of light and η is group refractive index depending on wavelength (λ). The fiber link delay asymmetry is given by

$$A(\lambda) = d_f(\lambda) - d_r(\lambda) = L \times \left(\frac{\eta_f(\lambda)}{c} - \frac{\eta_r(\lambda)}{c} \right), \quad (1)$$

where L denotes the transmission distance (fiber link length), d_f, d_r are forward and backward propagation delays, and η_f, η_r are the related refractive indexes. The mean fiber link delay (D) can be represented as

$$D = \frac{L}{2} \times \left(\frac{\eta_f(\lambda)}{c} + \frac{\eta_r(\lambda)}{c} \right). \quad (2)$$

Then,

$$L = \frac{2D}{\eta_f(\lambda)/c + \eta_r(\lambda)/c}. \quad (3)$$

Substituting L in (3) into (1) and simplifying, one obtains the fiber link delay asymmetry in terms of network mean fiber link delay as follows:

$$A(\lambda) = d_f(\lambda) - d_r(\lambda) = 2D \times \left(\frac{\eta_f(\lambda) - \eta_r(\lambda)}{\eta_f(\lambda) + \eta_r(\lambda)} \right). \quad (4)$$

Half of the delay asymmetry (i.e., $A(\lambda)/2$) will contribute to the time error, where $A(\lambda)$ depends on the wavelength spacing.

3. Scheme to Remove Asymmetry Error of Node and Link

In IEEE1588v2 distribution, assume that the fiber link delay in each direction is symmetric, whereas in WDM systems the delay may not be symmetric. Fortunately, if a T-BC is implemented in every node in OTN, the mean fiber link delay D can be estimated by the T-BC mode, which would know the difference $(d_f(\lambda) - d_r(\lambda))$ to compensate the phase offset (θ) as shown in (4). The compensation scheme is proposed as follows.

3.1. Telecom-Boundary Clock Mode. Each node and link in a network can introduce asymmetry. In Telecom-Boundary clock (T-BC) mode [14], ingress/egress buffers are bypassed, and nodes asymmetry is avoided as shown in Figure 4. The time transfer model as shown in Figure 5 can be written as

$$\begin{aligned} T_2 &= T_1 + \theta + \delta_{\text{link},M\to S}, \\ T_4 &= T_3 - \theta + \delta_{\text{link},S\to M}, \end{aligned} \quad (5)$$

where δ_{node}, δ_{link} denote node and link delay, respectively, and assume $\delta_{\text{node},M\to S} = \delta_{\text{node},S\to M}$.

Based on the time transfer model in (5), the estimated mean fiber link delay $\widehat{D}$ and estimated phase offset $\widehat{\theta}$ can be derived as

$$\widehat{D} = \frac{\delta_{\text{link},M\to S} + \delta_{\text{link},S\to M}}{2} = \frac{(T_2 - T_1) + (T_4 - T_3)}{2}, \quad (6)$$

$$\widehat{\theta} = \frac{(T_2 - T_1) - (T_4 - T_3)}{2} - \frac{(\delta_{\text{link},M\to S} - \delta_{\text{link},S\to M})}{2}. \quad (7)$$

Equation (7) shows that any asymmetry will contribute with half of that to the error in the phase offset calculation. The second term in (7) is the link asymmetry compensation. The link asymmetry consists of mainly fiber link length asymmetry and fiber link delay asymmetry for use of different wavelengths. Substituting $\widehat{D}$ in (6) into (4), assume that the fiber link length is symmetric; one obtains the fiber link delay asymmetry as

$$A(\lambda) = ((T_2 - T_1) + (T_4 - T_3)) \times \left(\frac{\eta_f(\lambda) - \eta_r(\lambda)}{\eta_f(\lambda) + \eta_r(\lambda)} \right). \quad (8)$$

Substituting (8) into (7), one obtains the estimated phase offset $\widehat{\theta}$ as

$$\widehat{\theta} = \left(\frac{(T_2 - T_1) - (T_4 - T_3)}{2} \right) - \frac{1}{2} ((T_2 - T_1) + (T_4 - T_3)) \times \left(\frac{\eta_f(\lambda) - \eta_r(\lambda)}{\eta_f(\lambda) + \eta_r(\lambda)} \right). \quad (9)$$

The second term in (9) represents the fiber link delay asymmetry compensation. If the same wavelength is used both on forward and backward paths (i.e., $\eta_f(\lambda) = \eta_r(\lambda)$), then (9) becomes

$$\widehat{\theta} = \left(\frac{(T_2 - T_1) - (T_4 - T_3)}{2} \right). \quad (10)$$

If there is a fiber length difference β between forward and backward paths, this will cause $\beta \cdot \eta(\lambda)/2c$ error in

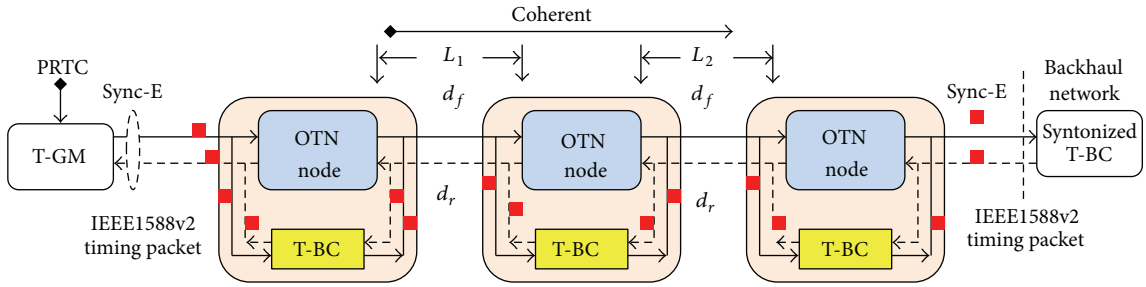


Figure 4: Telecom-Boundary clock mode in OTN.

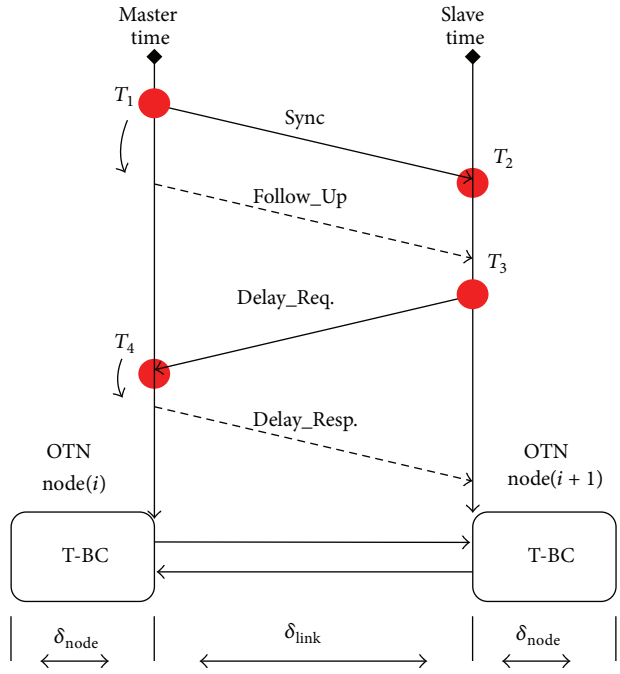


Figure 5: Time transfer mode based on T-BC mode

the estimation of phase offset $\widehat{\theta}$. For example, when η is 1.4682 at $\lambda = 1550$ nm, the estimated phase offset will have about 2.449 ns of error per meter of length asymmetry, which is related to the group delay (about 4.897 ns per meter).

3.2. Reducing Link Length Asymmetry. In a practical communication network, the link length asymmetry could be diminished to a tolerable extent if the fiber links are well designed at the beginning. An illustration of bidirectional and unidirectional protection switches in existent network fault management is shown in Figure 6. Bidirectional protection switch can minimize link length asymmetry (β) because two-way time transfer (TWTT) takes place within one cable. The cable asymmetry β should be within two meters ($\beta < 2$ m); this requires good cabling control. However, unidirectional protection switch TWTT takes place in separate cables, where the working and protection cables may not be in equal link length (i.e., $L_w \neq L_p$). In the current field trials, some budget is allocated for link length asymmetry unless the accurate link length asymmetry is manually measured and compensated.

4. In-Band Link Delay Asymmetry Analysis

The dispersion of single-mode optical fiber (e.g., SMF-28 that meets the requirements of ITU-T Recommendation G. 652) is

$$\bar{D}(\lambda) = \frac{\lambda S_o}{8}\left(1 - \frac{\lambda_o^4}{\lambda^4}\right), \tag{11}$$

where S_o (≤ 0.092 ps/(nm^2·km)) is the zero dispersion slope, λ_o (1302 nm $\leq \lambda_o \leq$ 1322 nm) is the zero dispersion wavelength ($\lambda_0 = 1310$ nm in the following calculation), and λ (1200 nm $\leq \lambda \leq$ 1600 nm) is the operating wavelength [15]. The index of refraction η and $\bar{D}(\lambda)$ are related by $(1/c)(\partial\eta/\partial\lambda) = \bar{D}(\lambda)$, which is then written as

$$\frac{\eta(\lambda)}{c} = \frac{\eta(\lambda_o)}{c} + \int_{\lambda_o}^{\lambda} \bar{D}(\lambda)\, d\lambda. \tag{12}$$

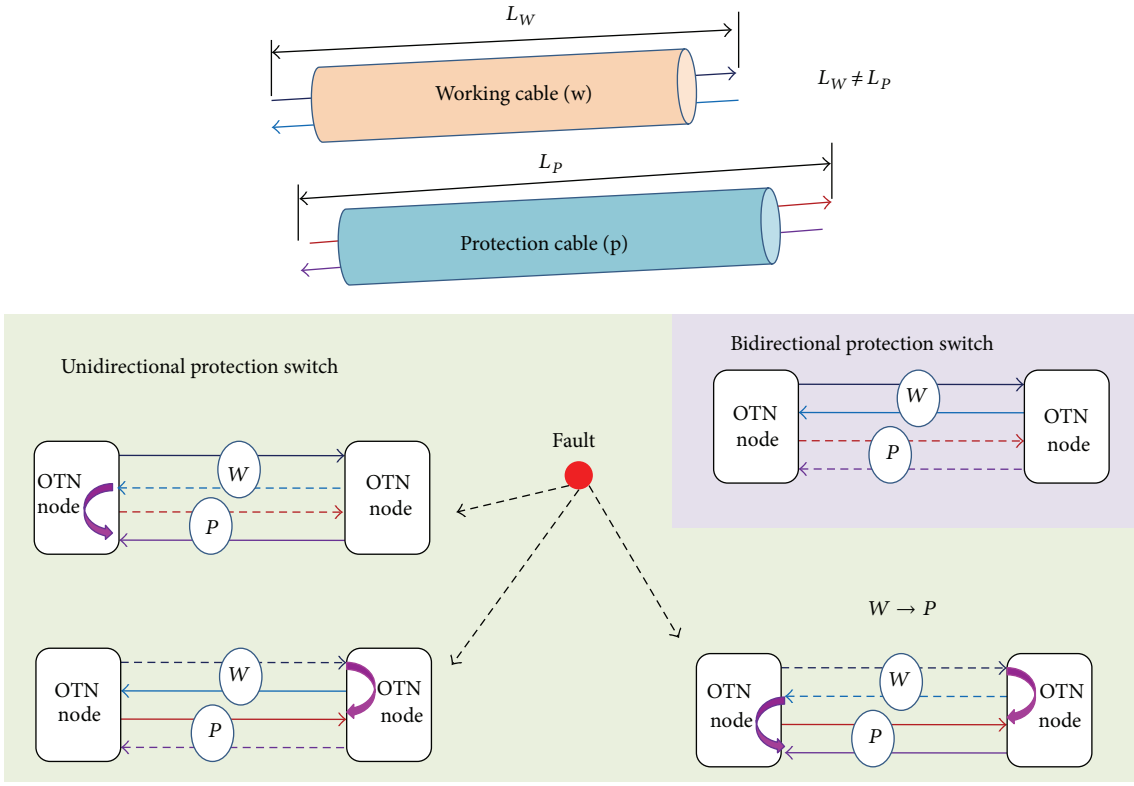


Figure 6: Schematic illustration of bidirectional/unidirectional network protection switches.

After integrating, we find that

$$\frac{\eta(\lambda)}{c} - \frac{\eta(\lambda_o)}{c} = \frac{S_o}{8}\lambda^2\left(1 - \frac{\lambda_o^2}{\lambda^2}\right)^2. \tag{13}$$

Substituting (13) into (1), the fiber link delay asymmetry per km (A/L) is

$$\begin{aligned}\frac{A}{L} &= \frac{\eta_f(\lambda_f)}{c} - \frac{\eta_r(\lambda_r)}{c} \\ &= \frac{S_o}{8}\left\{\lambda_f^2\left(1 - \frac{\lambda_o^2}{\lambda_f^2}\right)^2 - \lambda_r^2\left(1 - \frac{\lambda_o^2}{\lambda_r^2}\right)^2\right\},\end{aligned} \tag{14}$$

where λ_f and λ_r are the wavelengths in the forward and backward directions and are defined based on ITU wavelength grid specification. The fiber link delay asymmetry $A(\lambda)$ depends on the wavelength spacing $\{\lambda_f, \lambda_r\}$ and also fiber link length (L) as shown in (14). The calculated values of $A(\lambda)$ versus L(km) for $\{\lambda_f = 1569\,\text{nm}, \lambda_r = 1530\,\text{nm}\}$ in C-band and $\{\lambda_f = 1610\,\text{nm}, \lambda_r = 1570\,\text{nm}\}$ in L-band are depicted in Figure 7.

Based on (14), the maximum fiber link delay asymmetry for the two extreme wavelengths is about $A/L = 0.6795$ ns/km in C-band (i.e., $1530\,\text{nm} \leq \lambda \leq 1569\,\text{nm}$) and $A/L = 0.78854$ ns/km in L-band (i.e., $1570\,\text{nm} \leq \lambda \leq 1610\,\text{nm}$). This link delay asymmetry introduces significant time error (i.e., phase offset) within 0.3397 ns/km in C-band (e.g., $L = 100$ km, phase offset ≤ 33.97 ns) or 0.3943 ns/km in L-band (e.g., $L = 100$ km, phase offset ≤ 39.43 ns). The above results are summarized in Table 1.

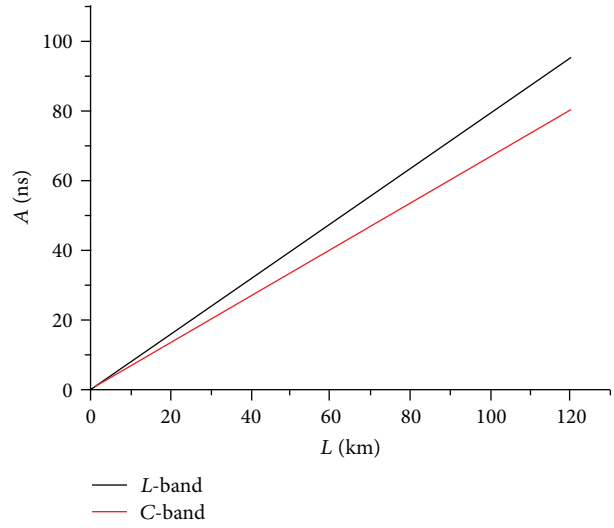


Figure 7: Fiber link delay asymmetry $A(\lambda)$ versus transmission distance L(km), where $\{\lambda_f = 1569\,\text{nm}, \lambda_r = 1530\,\text{nm}\}$ in C-band and $\{\lambda_f = 1610\,\text{nm}, \lambda_r = 1570\,\text{nm}\}$ in L-band.

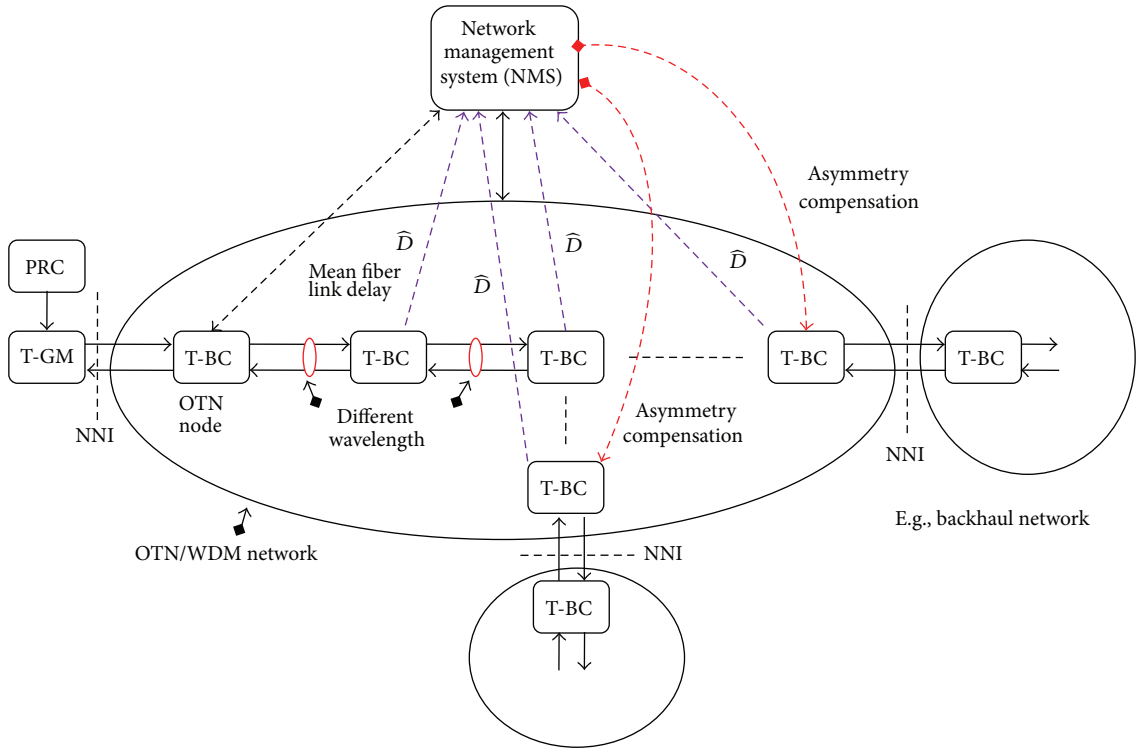


Figure 8: Asymmetry compensation support from control plane.

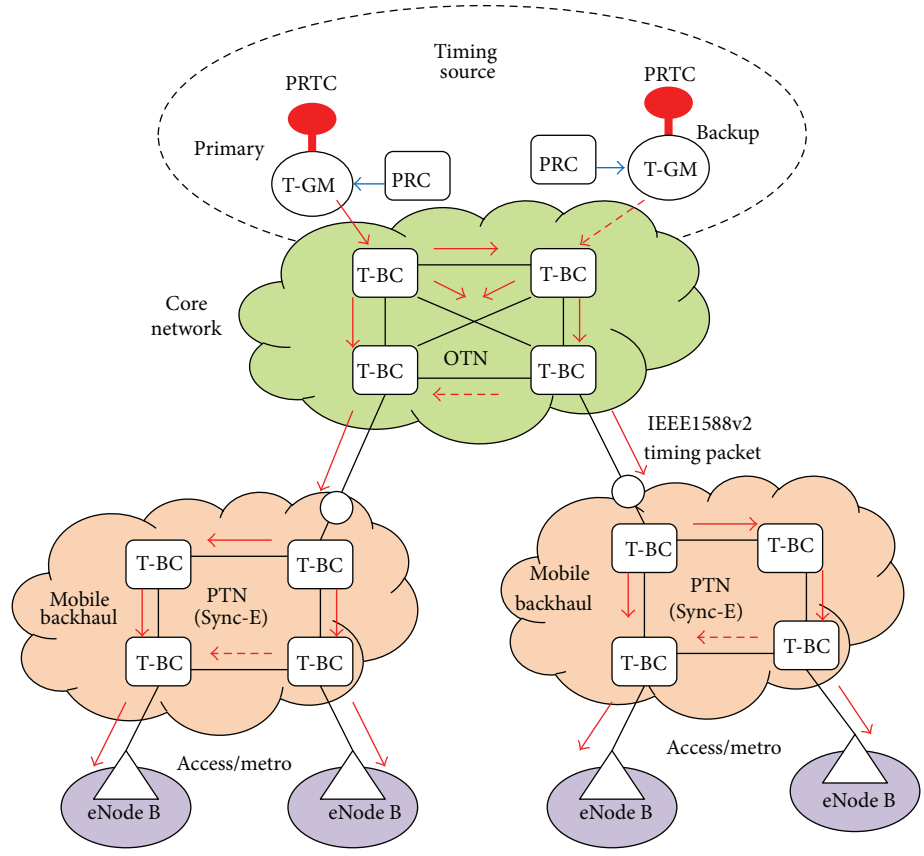


Figure 9: Time/phase over OTN networks with T-BC mode.

Table 1: Maximum fiber link delay asymmetry for the two extreme wavelengths.

Link delay asymmetry	C-band $1530\text{ nm} \le \lambda \le 1569\text{ nm}$	L-band $1570\text{ nm} \le \lambda \le 1610\text{ nm}$
$\frac{A(\lambda)}{L}$	≤0.6795 ns/km	≤0.78854 ns/km

For accurate time/phase transport, we have to take care of the fiber link delay asymmetry $A(\lambda)$, especially for long haul transmission. Nevertheless, this error may be canceled out to some extent *relying* on the estimate $\widehat{D}$ by T-BC (6). The OTN control plane contains global route information, which may play an important role in the asymmetry calibration process [10]. The network management system (NMS) is responsible for configuring the network including the wavelength (λ) assignment, collecting the mean fiber link delay ($\widehat{D}$) by the T-BC, and processing the fiber link delay asymmetry ($A(\lambda)$) in (8) to determine the asymmetry compensation in the timing chain. The sum of $A(\lambda)$ in the timing chain can be written as

$$A_T\left(\lambda_f, \lambda_r\right) = A_1\left(\lambda_{f1}, \lambda_{r1}\right) + A_2\left(\lambda_{f2}, \lambda_{r2}\right) + \cdots + A_N\left(\lambda_{fN}, \lambda_{rN}\right), \tag{15}$$

where $A_i(\lambda_{fi}, \lambda_{ri}) = 0$ for $\lambda_{fi} = \lambda_{ri}$. An illustration of asymmetry compensation support from OTN control plane is shown in Figure 8.

The integrity of transferring accurate time/phase synchronization over OTN and PTN with Sync-E networks is shown in Figure 9. Figure 9 is based on the full timing support (e.g., T-BC) from the network architecture as described in ITU-T G.8275.1, with the addition of frequency support (e.g., syntonized T-BC) being considered to improve time/phase recovery accuracy [16, 17]. The timing chain normally would be 11 hops (e.g., 10 T-BCs) and can extend to 15 hops (e.g., 14 T-BCs); this requires tight time error components control [18]. T-BC corrects the time/phase in the various network nodes and also provides a set of performance metrics including mean path delay and current offset from master [14]. As the PRTC cooperate with PRC (shown in Figure 9), the coherence between the frequency and time/phase planes can be realized, and this allows extending the time/phase holdover period during GNSS failures. Furthermore, a unified IEEE1588v2 management approach offers a compelling set of operational advantages including the ability to perform end-to-end performance analysis and troubleshooting.

5. Conclusion

The In-band fiber link delay asymmetry due to the use of different wavelengths in the two directions should be taken into account, especially for long haul transmission. This introduces significant time error (i.e., phase offset) within 0.3397 ns/km in C-band or 0.3943 ns/km in L-band depending on the wavelength spacing.

With the proposed scheme in this paper, the fiber link delay asymmetry can be compensated relying on the estimated mean fiber link delay by the T-BC mode and the NMS to compute the delay asymmetry in the timing chain. To deploy IEEE1588v2, bidirectional protection switch can minimize link length asymmetry in contrast to unidirectional protection switch.

It is an essential prerequisite to shorten the number of a T-BC chain, which can limit the impact of asymmetries. Furthermore, the integrity of transferring accurate time/phase synchronization over OTN and PTN with Sync-E networks can simplify network architecture, reduce OPEX, and make the network easy to maintain.

Conflict of Interests

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

Acknowledgment

This work is supported in part by the Bureau of Standards, Metrology and Inspection, Ministry of Economic Affairs, Taiwan (Grant no. BSMI 103-1403-05-05-01).

References

[1] A. Bari, J. Jiang, W. Saad, and A. Jaekel, "Challenges in the smart grid applications: an overview," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 974682, 11 pages, 2014.

[2] D. Schneider, "The microsecond market," *IEEE Spectrum*, vol. 49, no. 6, pp. 66–81, 2012.

[3] P. Kartaschoff, "Synchronization in digital communications networks," *Proceedings of the IEEE*, vol. 79, no. 7, pp. 1019–1028, 1991.

[4] A. Magee, "Synchronization in next-generation mobile backhaul networks," *IEEE Communications Magazine*, vol. 48, no. 10, pp. 110–116, 2010.

[5] J.-L. Ferrant, M. Gilson, S. Jobert et al., *Synchronous Ethernet and IEEE 1588 in Telecoms: Next Generation Synchronization Networks*, Wiley-ISTE, 2013.

[6] ITU-T Recommendation G.8275, *Architecture and Requirements for Packet-Based Time and Phase Delivery*, 2013.

[7] ITU-T Recommendation G.709/Y.1331, Interfaces for the Optical Transport Network (OTN), December 2009.

[8] K. Hann, S. Jobert, and S. Rodrigues, "Synchronous ethernet to transport frequency and phase/time," *IEEE Communications Magazine*, vol. 50, no. 8, pp. 152–160, 2012.

[9] J.-L. Ferrant, M. Gilson, S. Jobert et al., "Development of the first IEEE 1588 telecom profile to address mobile backhaul needs," *IEEE Communications Magazine*, vol. 48, no. 10, pp. 118–126, 2010.

[10] S. Ruffini, "Time sync distribution via PTP challenges, asymmetries, solutions," in *Proceedings of the International Telecom Sync Forum (ITSF '11)*, pp. 3–37, 2011.

[11] M. Ouellette, K. Ji, S. Liu, and H. Li, "Using IEEE 1588 and boundary clocks for clock synchronization in telecom networks," *IEEE Communications Magazine*, vol. 49, no. 2, pp. 164–171, 2011.

[12] M. Ouellette, G. Garner, and S. Jobert, "Simulations of a chain of telecom boundary clocks combined with synchronous ethernet for phase/time transfer," in *Proceedings of the International IEEE Symposium on Precision Clock Synchronization for Measurement, Control, and Communication (ISPCS '11)*, pp. 105–113, Munich, Germany, September 2011.

[13] K. Cochrane, J. E. Bailey, P. Lake, and A. Carlson, "Wavelength-dependent measurements of optical-fiber transit time, material dispersion, and attenuation," *Applied Optics*, vol. 40, no. 1, pp. 150–156, 2001.

[14] "IEEE standard for a precision clock synchronization protocol for networked measurement and control systems," IEEE Std 1588-2008, Revision of IEEE Std 1588-2002, July 2008.

[15] ITU-T Recommendation G.984.3, Amendment 2: Time-of-day distribution and maintenance and clarifications, 2008.

[16] ITU-T Recommendation G.8251, "The Control of Jitter and Wander within the Optical Transport Network (OTN)," November 2010.

[17] J.-L. Ferrant, G. M. Garner, M. Mayer, J. Rahn, S. Rodrigues, and S. Ruffini, "OTN timing aspects," *IEEE Communications Magazine*, vol. 48, no. 9, pp. 62–69, 2010.

[18] ITU-T, Network Limits for Time Synchronization in Packets Networks, ITU-T Recommendation G.8271.1, 2013.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 105428, 11 pages
http://dx.doi.org/10.1155/2014/105428

Research Article

Content Patterns in Topic-Based Overlapping Communities

Sebastián A. Ríos and Ricardo Muñoz

Industrial Engineering Department, University of Chile, Av. República 701, 8370439 Santiago, Chile

Correspondence should be addressed to Ricardo Muñoz; rimunoz@ing.uchile.cl

Received 13 January 2014; Accepted 19 February 2014; Published 9 April 2014

Academic Editors: H. R. Karimi, X. Yang, and Z. Yu

Understanding the underlying community structure is an important challenge in social network analysis. Most state-of-the-art algorithms only consider structural properties to detect disjoint subcommunities and do not include the fact that people can belong to more than one community and also ignore the information contained in posts that users have made. To tackle this problem, we developed a novel methodology to detect overlapping subcommunities in online social networks and a method to analyze the content patterns for each subcommunities using topic models. This paper presents our main contribution, a hybrid algorithm which combines two different overlapping sub-community detection approaches: the first one considers the graph structure of the network (topology-based subcommunities detection approach) and the second one takes the textual information of the network nodes into consideration (topic-based subcommunities detection approach). Additionally we provide a method to analyze and compare the content generated. Tests on real-world virtual communities show that our algorithm outperforms other methods.

1. Introduction

The community finding problem [1] is key to understanding several different problems in social networks and other areas, for example, how these evolve through time [2], how information spreads in (online) human networks, and spam detection [3], among others.

Most community finding algorithms perform hard clustering on human networks based only on structural criteria, such as finding clique structures. Therefore, every member of a social network is assigned to only one community. However, we know that people can belong to more than one community (or subset of people with the same interests). Another shortcoming of these algorithms is that they ignore the content generated by the members of the community, where every post can let us improve the detection of the communities to which a member belongs. Methodologies showing that topics can be used to understand the dynamics of community structures are presented in Ding [4] and Yan et al. [5]. However, these methodologies are not comparable with our work. Thus, we decided to develop an algorithm to detect such communities allowing a member to participate in more than one community including semantic information. This is known as the overlapping community discovery problem. We benchmarked several algorithms, and we selected the best two algorithms for evaluating our algorithm, which are COPRA [6] and speaker-listener propagation algorithm (SLPA) [7]. We also used the speaker-listener topic propagation algorithm (SLTA), created by our group, which also performs well, as reported in [8].

Since 2009 we have stated the need for adding the semantic information of written comments by community members in order to perform online social network analysis (SNA) in a better (closer to reality) manner. This means extracting meaningful information leaving aside interactions which do not contribute to the main topics of human communication that are called online social network noise. We have proposed semantic filters to tackle this problem, reducing noise and showing good results in identifying key-members (central nodes) [9, 10], detecting preferences [11], and detecting a community (without overlapping) [12], among others.

This time, our main contribution is a new community finding algorithm (TPA) which allows members to belong to more than one community. Our algorithm (TPA) considers both structural properties of posted messages and the semantic information of posted content. We compared TPA to the best state-of-the-art algorithms, discovering that TPA outperforms them. In addition, we incorporate a method

to interpret the content generated in each of the identified subcommunities, which is very useful when the activity in a social network is described.

This paper is structured as follows. In Section 2, previous work on detecting context-dependent communities in social networks is introduced. In Section 3, the proposed methodology and TPA algorithm are presented. In Sections 4 and 5 we present an experimental setup for an online community, and its results are discussed. Finally in Section 6, we establish the main conclusions of this work and indicate directions for future research.

2. Related Work

A great number of algorithms have been developed using a variety of methods [1, 13–15]; these vary in their effectiveness and time performance for different types of networks. This section summarizes three algorithms for overlapping community detection. Then, algorithms are benchmarked in unsupervised manner; therefore, we used link-based modularity [16] as a clustering quality measure for these algorithms.

2.1. Algorithms

2.1.1. Community Overlap PRopagation Algorithm (COPRA). COPRA (http://www.cs.bris.ac.uk/~steve/networks/software/copra.html) [6] is an algorithm based on the label propagation technique of Raghavan, Albert, and Kumara, but it is able to detect communities that overlap. Like the original algorithm, vertices have labels that propagate between neighboring vertices so that community members reach a consensus on their community membership, each node updating the coefficients that belong to it by averaging the coefficients from all its neighbors at each time step in a synchronous fashion.

2.1.2. Speaker-Listener Propagation Algorithm (SLPA). SLPA (https://sites.google.com/site/communitydetectionslpa/) [7] is a general speaker-listener based information propagation process. It spreads labels between nodes according to pairwise interaction rules. Unlike other algorithms, where a node forgets information gained in previous iterations, SLPA provides each node with a memory for storing received information (i.e., labels). The membership strength is interpreted as the probability of observing a label in a node's memory. In SLPA we need to determine the following: (1) how to spread node information to other nodes and (2) how to process the information received from other nodes. The critical issue related to both questions is how information should be maintained.

2.1.3. Speaker-Listener Topic Propagation Algorithm (SLTA). SLTA [8, 17] is a modification of the speaker-listener propagation algorithm (SLPA) [7]. In SLPA, the memory of each node is initialized with the node's id. SLTA follows this idea but applies a different initialization process. SLTA mimics human pairwise communication behavior. At each communication step, each node serves as both a speaker (information provider) and a listener (information consumer). Specifically, each node broadcasts a topic of interest to neighbors and at the same time receives an indication of interest from each neighbor.

2.2. Evaluation Criteria: Link-Based Modularity. To measure the quality of a cover (a cover of network is defined as a set of clusters such that each node is assigned to one or more clusters and no cluster is a proper subset of any other cluster) produced by overlapping detection algorithms on real-world social networks, where the ground truth is usually unknown, most measures extend the framework of modularity Q for a disjoint partition [18], which is given as

$$Q = \frac{1}{2m}\sum_{c}\sum_{i,j\in c}\left[A_{ij} - \frac{k_i k_j}{2m}\right], \quad (1)$$

where c is a community, A_{ij} is the element of the adjacency matrix for nodes i and j, $m = (1/2)\sum_{ij} A_{ij}$ is the total number of edges, and k_i is the degree of node i (each network node corresponds to an individual in the community).

In this paper we will use an extension of modularity based on the belonging coefficients of links proposed by Nicosia et al. [16]; this extension of modularity is used to evaluate the goodness of overlapped community decomposition. The Nicosia modularity is defined as

$$Q_{ov}^{Ni} = \frac{1}{m}\sum_{c}\sum_{i,j\in V}\left[\beta_{l(i,j),c}A_{ij} - \beta^{\text{out}}_{l(i,j),c}\beta^{\text{in}}_{l(i,j),c}\frac{k_i^{\text{out}}k_j^{\text{in}}}{m}\right], \quad (2)$$

where $\beta_{l(i,j),c}$ is the community belonging coefficient c of an edge $l = (i, j)$ which starts at node i and ends at node j. $\beta^{\text{out}}_{l(i,j),c}$ is the expected belonging coefficient of any possible link $l(i, j)$ from node i to node j in community c and $\beta^{\text{in}}_{l(i,j),c}$ is the expected belonging coefficient of any link $l(i, j)$ pointing to node j in community c. k_i^{out} and k_i^{in} are the out-degree and the in-degree of node i, respectively, and V is the set of nodes. Note that if the modularity of an algorithm in a community equals zero, it implies that the algorithm did not find a community structure on this community and its output is a cluster containing all the community members.

3. Proposed Model

This section is organized as follows: first, basic notation and representation of documents are introduced; then, probabilistic models, network configuration, topic-based network filtering, and a modified community detection algorithm are presented.

3.1. Basic Notation and Concepts. Let us introduce some concepts. In the following, let $\mathscr{V}$ be a vector of words that defines the vocabulary to be used. We will refer to a word w, as a basic unit of discrete data, indexed by $\{1, \ldots, |\mathscr{V}|\}$. A posted message is a sequence of S words defined by $\mathbf{w} = (w^1, \ldots, w^S)$, where w^s represents the sth word in the message. Finally, a corpus is defined by a collection of $\mathscr{P}$ posted messages denoted by $\mathscr{C} = (\mathbf{w}_1, ..., \mathbf{w}_{|\mathscr{P}|})$.

Social networks and communities have been studied by sociologists for many decades. They have proposed the following types of communities: communities of interest, communities of purpose, and communities of practice. In our previous work, we showed that our methodology enhances community detection in communities of practice (CoP). Thus, to determine whether these results can be replicated with other communities we focused on communities of interest (CoI) which have been studied by many researchers such as Kosonen [19] and Porter [20]. The dark web corresponds to virtual communities of interests (VCoI) [19, 20]. It gathers together groups of members whose interests are shared on different levels by the community users [10].

3.2. Topic Modeling. A topic model, for example, latent Dirichlet allocation (LDA) [21], can be considered to be a probabilistic model that relates documents and words through variables which represent the main topics inferred from the text itself. In this context, a document can be considered as a mixture of topics, represented by probability distributions which can generate the words in a document given these topics. The inferring process of the latent variables, or topics, is the key component of this model, whose main objective is to learn from text data the distribution of the underlying topics in a given corpus of text documents.

With LDA, given the smoothing parameters β and α and a joint distribution of a topic mixture θ, the idea is to determine the probability distribution to generate—from a set of topics $\mathscr{T}$—a message composed by a set of S words w ($\mathbf{w} = (w^1, \ldots, w^S)$),

$$p(\theta, z, \mathbf{w} \mid \alpha, \beta) = p(\theta \mid \alpha) \prod_{s=1}^{S} p(z_s \mid \theta)\, p(w^s \mid z_s, \beta), \quad (3)$$

where $p(z_s \mid \theta)$ can be represented by the random variable θ_i; this topic z_s is present in document i ($z_s^i = 1$). A final expression can be deduced by integrating (3) over the random variable θ and summing topics $z \in \mathscr{T}$.

3.3. Network Configuration. To build the social network graph, the members' interaction must be taken into consideration. In general, member activity is followed according to participation in the forum. Likewise, participation appears when a member posts a comment in the community. The network will be configured according to the following: *nodes* will represent VCoI members and *arcs* will represent interactions among them. How to link the members and how to measure their interactions to complete the network is our main concern.

In this work we used an *all-previous-reply network* [9, 10, 22] to represent the VCoI network. This means when a member creates a post in a thread, every reply following it will be relayed to all the people who replied before on the thread. In other words, we assume that the last reply is a broadcast to all members who posted a comment before in that specific thread. This type of network representation is the densest and with the greatest number of interactions, and also the one with the most noisy arcs. It is therefore the hardest on which to apply data mining or social network analysis (SNA), and from which to extract useful information.

3.4. Topic-Based Network Filtering. The main idea of semantic filters is to compare semantic information of two members' posts with Euclidean distance. The semantic is extracted or represented by topics, which are not keywords. If the similarity is over a certain threshold θ, an interaction will be considered between them. We support the idea that this will help avoid irrelevant interactions. For example, in a VCoI with k topics, let TB_j be a post of user j that is a reply to post of user i (TB_i). The cosine similarity between them will be calculated with

$$d_m\left(\mathrm{TB}_i, \mathrm{TB}_j\right) = \frac{\sum_k g_{ik} g_{jk}}{\sqrt{\sum_k g_{ik}^2 \sum_k g_{jk}^2}}, \quad (4)$$

where g_{ik} is the score of topic k in the post of user i (the topic-post distribution obtained with LDA). It is clear that the similarity exists only if TB_j is a reply to TB_i. After that, the weight of arc $a_{i,j}$ is calculated according to

$$a_{i,j} = \sum_{\substack{i,j \\ d_m\left(\mathrm{TB}_i, \mathrm{TB}_j\right) \geq \theta}} d\left(\mathrm{TB}_i, \mathrm{TB}_j\right). \quad (5)$$

Considering all posts $\mathscr{P}$, the network is built following the structure described in Section 3.3. In other words, for each post of user i in a thread, the arc $a_{i,j}$ is added for each user j who posted a comment on that thread. But we only consider the arcs if the similarity of their messages is greater than or equal to the threshold θ in (5). This way, we are able to filter arcs by topic similarity to a specific thread's topics.

3.5. Community Detection in Topic-Based Networks, Topic Propagation Algorithm (TPA). This algorithm extends the idea presented by Ríos and Muñoz [8] where the topic used most for a node is propagated in the network. In this algorithm, nodes interact among themselves following a certain interaction rule, which updates the membership vector of each node in an asynchronous process. The membership vector for each node is initialized with its topic score's vector (see (6)). In summary, the proposed algorithm consists of the following three stages.

(1) The membership vector of each node is initialized with its topic score's vector. The topic score's vector is computed using text mining techniques, specifically LDA, which was applied in the topic modeling step (see Section 3.2).

(2) Then, the following steps are repeated until the stop criterion is satisfied.

   (a) One node is selected as a candidate.
   (b) The average membership vector of all neighbors of the selected node is calculated.
   (c) The candidate updates its membership vector following a certain interaction rule between its

membership vector and the average membership vector of its neighbors. Then, the candidate's membership vector is normalized.

(3) Finally, the postprocessing based on the belonging vectors of nodes is applied to output the communities.

In the initialization process, the membership vector of a node i is initialized with its topic score's vector Ψ^i; Ψ^i is a vector where every component is the average score over all post messages from user i. Mathematically,

$$\Psi_k^i = \frac{1}{|\mathrm{TB}|} \sum_{l \in \mathrm{TB}} q_{ikl} \tag{6}$$

$$\Psi^i = \frac{1}{\left\|\Psi^i\right\|} \Psi^i, \tag{7}$$

where TB is the set of users' posted messages; q_{ikl} is the score of topic k in post l of user i.

Let Ψ_n^i be the normalized average membership vector of node i's neighborhood. Then, the interaction rule between Ψ^i and Ψ_n^i at iteration t updates the belonging vector of node i as follows:

$$\begin{aligned} \Psi^{i,t} &= \Psi^{i,t-1} + \varphi\left(\Psi^{i,t-1}, \Psi_n^{i,t-1}\right)\left[\Psi_n^{i,t-1} - \Psi^{i,t-1}\right] \\ \Psi^{i,t} &= \frac{1}{\left\|\Psi^{i,t}\right\|} \Psi^{i,t}. \end{aligned} \tag{8}$$

After a modularity optimization process, we estimated the function $\varphi(\Psi^i, \Psi_n^i)$ as

$$\varphi\left(\Psi^i, \Psi_n^i\right) = \exp\left(-\frac{t}{4\,\log(k)}\right). \tag{9}$$

(a) Stopping Criterion. We can stop at any time as long as we collect sufficient information for postprocessing. In the current implementation we simply stop when the predefined maximum number of iterations T is reached. Although TPA is nondeterministic due to the random selection, it performs well on average as shown in later sections. This algorithm produces relatively stable outputs, independent of network size or structure, when T is greater than 20.

(b) Postprocessing and Community Detection. Given the membership vector of a node/member, a simple thresholding procedure is performed to produce an overlapped assignment of members to different communities. If the topic score of certain component is greater than a given threshold $r \in [0, 1]$, this node belongs to those communities. Thus, that node is called an overlapping node.

3.6. Community Characterization. The aim of community characterization is to find out what the community is talking about. For this purpose, we propose the following methodology. Let $T = \mathrm{VC}(V, E) \in \mathrm{TVC}$ be a virtual community and $C_1, C_2, \ldots, C_{|C|}$ a set of cover for virtual community T. For each cover C_j we define the content community pattern CV_j, a vector of length K, where K is number of topics extracted for corpus $\mathscr{C}$. The component CV_j^k can be expressed as follows:

$$CV_j^k = \frac{\left|cv_j^k\right|}{\sum_{j \in C}\left|cv_j^k\right|}, \tag{10}$$

where $cv_j^k = \{P_{il} \in \mathrm{TB} \mid q_{ikl} \geq \kappa \text{ and } i \in C_j\}$ and cv_j^k is the set of posts written by members of community j who has a topic score q_{ikl} for topic k greater than a threshold $\kappa \in [0, 1]$.

4. Experimental Setup and Results

The method presented in Section 3.5 was evaluated over a dark web portal dataset. The dark web forum portal [23] is a web-based knowledge portal which was created based on a general framework for web forum data integration. The portal incorporates the data collected from different international Jihadist forums. These online discussion sites are dedicated to topics relating primarily to Islamic ideology and theology. The dark web can be considered to be a virtual community of interests (VCoI) whose members are extremists who share and comment on their feelings and interests with others who support their cause. Our proposed methodology for overlapping community detection was applied to the `IslamicAwakening` English language based forum, available on ISI-KDD 2012 Website (http://www.ischool.drexel.edu/isi-kdd2012/challenge.html).

Next, an analysis of topics extracted using LDA (described in Section 3.2) is presented. Then, the network topology construction is described by using *all-previous-reply* oriented structures for the whole period. Finally, overlapping community detection algorithms were applied, and their results were compared with different LDA-filtered networks.

In order to validate the proposed method (described in Section 3.5), we applied this method using five different LDA-filtered networks. To better understand the performance of the proposed algorithm, we compared TPA with two well-known algorithms, COPRA [6] and SLPA [7]. We also compared TPA with SLTA [8, 17], a modified version of SLPA which includes semantic information.

We used default parameter settings for most algorithms where applicable. For TPA the maximum number of iterations T was set to 30, and for SLPA and SLTA, this parameter was set to 100. The threshold r for SLPA and SLTA takes values in set $\{0.01, 0.05, 0.1, 0.2, 0.3, 0.4\}$ and for TPA, parameter r varies from $1/k$ to $5/k$ with an interval $1/2\,k$, where k is the number of extracted topics. For the extracted communities we measured and reported the maximum performance over ten repetitions for TPA, SLPA, SLTA, and COPRA.

We selected overlapping link-based modularity Q_{ov}^{Ni} in (2) as a quality measure. Although modularity is a function of a cover and a network, we sometimes refer to the *modularity of algorithm A on network N*, referring to the modularity of the cover produced by algorithm A when run on network N.

4.1. Topic Extraction. There are 7 years (2004–2010) of data available. Posts were created by 2,792 members and extracted topics were realized over 127,216 posts $\mathscr{P}$ and 244, 200 words

Table 1: Ten most relevant words with their respective conditional probabilities for the five most relevant topics for all data from the IslamicAwakening Forum.

Topic 19 "War"	Topic 15 "Terrorist attacks (general)"	Topic 33 "Political/terrorist trials"	Topic 54 "Religion (Allah)"	Topic 51 "Islamic religion"
Government (0.0140)	Kill (0.0255)	Court (0.0149)	Allah (0.0141)	Islamic (0.0254)
War (0.0113)	Police (0.0180)	Guantanamo (0.0099)	Prophet (0.0163)	Muslim (0.0176)
Military (0.0104)	Soldier (0.0168)	Trail (0.0080)	Messenger (0.0141)	Islam (0.0159)
Country (0.0091)	Attack (0.0139)	Prison (0.0073)	Peace (0.0118)	World (0.0126)
United (0.0089)	Force (0.0137)	Judge (0.0066)	People (0.0113)	Religious (0.0107)
Security (0.0083)	Military (0.0109)	Torture (0.0065)	Lord (0.0107)	Society (0.0104)
Force (0.0079)	Official (0.0105)	Rights (0.0061)	Day (0.0099)	People (0.0086)
International (0.0059)	Security (0.0089)	Charges (0.0057)	Believer (0.0096)	Law (0.0083)
Official (0.0059)	Report (0.0086)	Government (0.0056)	Bless (0.0084)	Political (0.0078)
American (0.0056)	Army (0.0075)	Arrested (0.0056)	Quran (0.0077)	Western (0.0070)

in the vocabulary $\mathscr{V}$ by using a Java Gibbs sampling-based implementation of LDA (http://jgibblda.sourceforge.net/) previously described in Section 3.2.

The application of LDA over text content resulted in $\{10, 50, 100\}$ topics with 20 words and their respective probabilities. The most popular topics extracted from the IslamicAwakening forum are presented in Table 1. These topics represent the most popular ideas posted in the forum when 100 topics are extracted.

4.2. Topic-Based Social Network Visualization. The graph has many variables which modify its configuration [22].

(1) *Time.* One dimension that was not mentioned before is time. The time period to be analyzed could be a month, annually, or whole history of posts of the network.

(2) *Graph Filtering.* Including the traditional nonfiltered graph, the other four configurations correspond to graphs filtered by topics, as presented in Section 3.4.

(3) *Interaction Topology.* According to the assumption of *who* is replying, the all-previous-reply network is considered. In this configuration every reply of a thread will be a response to all posts which are already in a specific thread.

To configure the network and have the graph representation, all of these three variables have to be decided. In this work, three temporal virtual communities (TVC) of interest are used. They are extracted from the same forum data, but they have different time frames. We used 20 four-month time frames for the period between January 2004 and August 2010.

Networks were built from 2004 to 2010 using three different topic sets, and for comparisons, the threshold θ takes values in set $\{0.0, 0.1, 0.2, 0.3, 0.4\}$ for LDA-Filter, as explained in Section 3.4. Then, for each interaction representation, the result is a graph with the members who posted in a specific period of time and has an interaction greater than or equal to the filter threshold. We chose the LDA threshold to eliminate a large number of irrelevant interactions but without excluding many members from the network. For example, after applying the highest threshold over the one-year TVC, the networks have, on average, 25% fewer interactions but we excluded only 5% of the members.

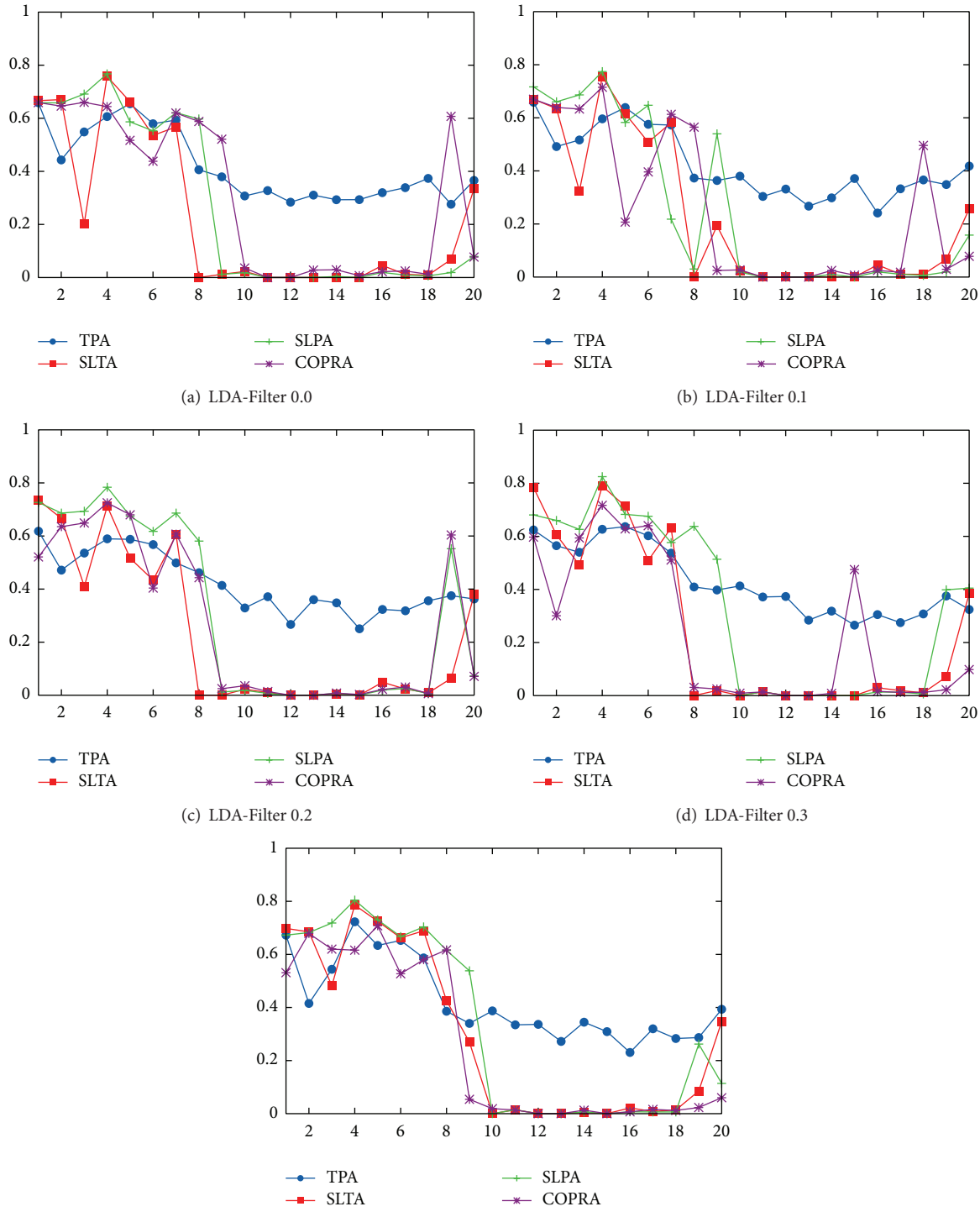
5. Overlapping Community Detection

Our results are shown in Figure 1. Figure 1 shows results obtained after applying the methodology described in Section 3 for all four-month TVCs available. This figure shows the maximum link-based modularity (Q_{ov}^{Ni}, see Section 2.2) over ten repetitions for algorithms considered in this work. Despite using three different topic sets to detect overlapping communities, we only present those results with the highest average modularity which were obtained using 50 topics.

Results show that TPA achieves, on average, a modularity measure of 0.33 while the best state-of-the-art algorithm achieves only 0.043 when it is applied over a VCoI. Nonetheless, there are no different results when the semantic filter applied over the networks is increased. We excluded results with a modularity measure that equals zero from this analysis because of the distortion that occurs when the average is calculated.

According to sociological theory, it is expected that VCoI's members are related to one another because they share the same interests. Therefore, all users should belong to the same community. Most algorithms capture this; they find only a cover which contains all members. Nonetheless, TPA is able to detect several subcommunity structures. We will see below that detected community structures define groups of members that share almost all the content generated within them, and they differ only in a few topics while most state-of-the-art algorithms cannot detect these groups because they do not include semantic information as TPA.

In Figure 1 we do not observe a clear advantage of one algorithm over the other in the period between 2004 and 2006 (see abscissa points from 1 to 9). However, TPA achieves a clear advantage for the period between 2007 and 2010 (see abscissa points from 10 to 20), because during this period the VCs become denser, making it more difficult for traditional algorithms to find a community structure using only structural properties, and discovered subcommunities



(a) LDA-Filter 0.0

(b) LDA-Filter 0.1

(c) LDA-Filter 0.2

(d) LDA-Filter 0.3

(e) LDA-Filter 0.4

Figure 1: Algorithm modularity comparison for four-month TVC. (Figures include the period from Jan. 2004 to Aug. 2010.)

are only explained due to subtle changes in the content spoken in each one of them, which is captured by TPA. We obtained similar TPA behavior regardless of the semantic filter (θ) applied. Remember that if the modularity of some algorithm on a VC equals zero, it implies that the algorithm did not find a community structure on this VC and its output is a cluster containing all the VC's nodes. This means that our algorithm systematically discovers overlapping communities, while other algorithms do not discover any community structures.

After analyzing the effect of different semantic filters on the quality of overlapping community detection on VCoI, it

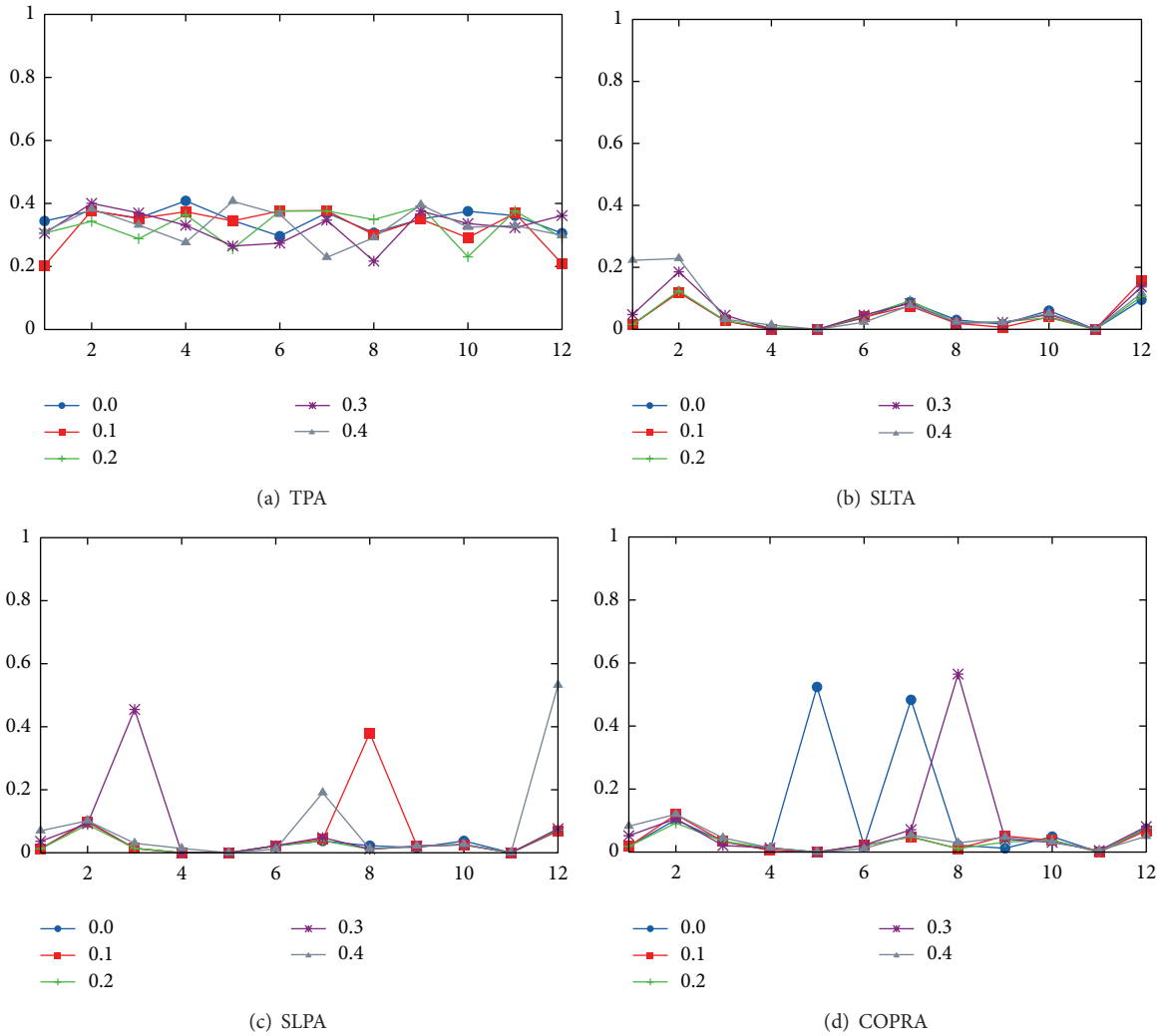


Figure 2: Effects of different semantic LDA-Filter on algorithms using one-month TVC (we include 2009).

is possible to see that there are no significant differences. In Figure 2 we illustrate the effect of LDA-Filters on overlapped community detection performance using one-month TVC for the year 2009.

Findings. The former is very important since in a previous work using virtual communities of practice (VCoP) [17] we discovered that semantic filters do affect overlapping community detection performance. In other words, the modularity of SLTA and SLPA is higher when the semantic threshold is higher; that is, irrelevant interactions are deleted. Therefore, less noise is processed and more communities are found. This means that the type of virtual community being processed affects the outcome of the algorithms.

We can state that we have changed the actual idea of community, requiring that at least two nodes have an edge connecting each other to belong to the same community. In our case, we extended that notion since—in addition to a connected subgraph—we also used the members' interaction semantic information, which means that two nodes belong to the same community if they are also interested in the same topics. This allows us to identify better groups and find much better communities, where every community is characterized according to the methodology developed in this work (see Section 3). In Figure 3, we show communities detected through TPA for a 2005 data network (LDA-Filter 0.1). Multicolor nodes indicate overlap nodes/members. This figure illustrates the new concept of community where two nodes do not need to be connected to belong to the same community, while they share the same interest. Our object is that this new definition of community will allow obtaining results that are closer to reality while gathering better information for analysts. This new information can be used from recommender systems to community management or even community moderation.

5.1. Community Characterization. In this section, detected overlapping communities are characterized according to the methodology proposed in Section 3.6.

Table 2: Three largest communities for 2005-VC (0.1).

Community	Topic	Top 10 words
Largest community members = 50	Topic 12 (0.043)	Ibn, hadith, imam, abu, book, narration, ahmad, al, weak, Muhammad
	Topic 11 (0.043)	People, time, feel, talk, person, bad, life, understand, live, start
	Topic 27 (0.043)	Allah, people, lord, heart, love, believer, day, person, life, prophet
	Topic 47 (0.043)	Quote, posted, originally, bro, akhi, abumuwahid, ahmed, brothermujahid, waziri, wild
	Topic 40 (0.041)	Sheikh, al, ibn, Muhammad, scholar, bin, Abdullah, book, lecture, anwar
Second community members = 21	Topic 47 (0.055)	Quote, posted, originally, bro, akhi, abumuwahid, ahmed, brothermujahid, Waziri, wild
	Topic 11 (0.045)	People, time, feel, talk, person, bad, life, understand, live, start
	Topic 27 (0.045)	Allah, people, lord, heart, love, believer, day, person, life, prophet
	Topic 10 (0.038)	Salafi, people, call, sunnah, dawah, issue, aqeedah, scholar, qutb, manhaj
	Topic 12 (0.033)	Ibn, hadith, imam, abu, book, narration, ahmad, al, weak, Muhammad
Third community members = 17	Topic 27 (0.067)	Allah, people, lord, heart, love, believer, day, person, life, prophet
	Topic 47 (0.054)	Quote, posted, originally, bro, akhi, Abumuwahid, ahmed, brothermujahid, waziri, wild
	Topic 11 (0.054)	People, time, feel, talk, person, bad, life, understand, live, start
	Topic 34 (0.047)	Sufi, music, sheikh, tasawwuf, love, listen, people, sound, call, singing
	Topic 29 (0.040)	Alaykum, assalam, wa, salam, inshallah, assalamu, rahmatullah, hope, forum, false

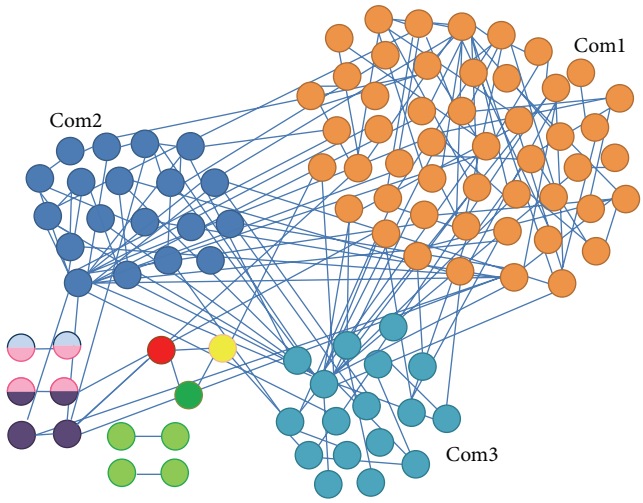


Figure 3: Dark web portal 2005 LDA-filtered network, with LDA-Filter 0.1.

Figure 4 shows the content pattern for the three largest communities detected in filtered 2005-VC with LDA-Filter $\theta = 0.1$ and $\theta = 0.2$. Each content pattern shows the topic distribution over each community. In this case, it is possible to see how different content patterns are when communities for the same filtered VC are analyzed; this is likely to happen because the communities were formed towards different interest topics. On the contrary, when communities of similar size are analyzed across different filtered 2005-VC it is possible to see that the content for each does not change when we apply different semantic filters. These properties are expected of any semantic-based community description. First, it is expected that detected communities in the same VC are described by different topic distributions. Finally, the content, that is, topic distribution within a community, is independent of irrelevant interactions. Therefore, topic distribution for a community does not change when a semantic filter is applied, since its topics are always its topics.

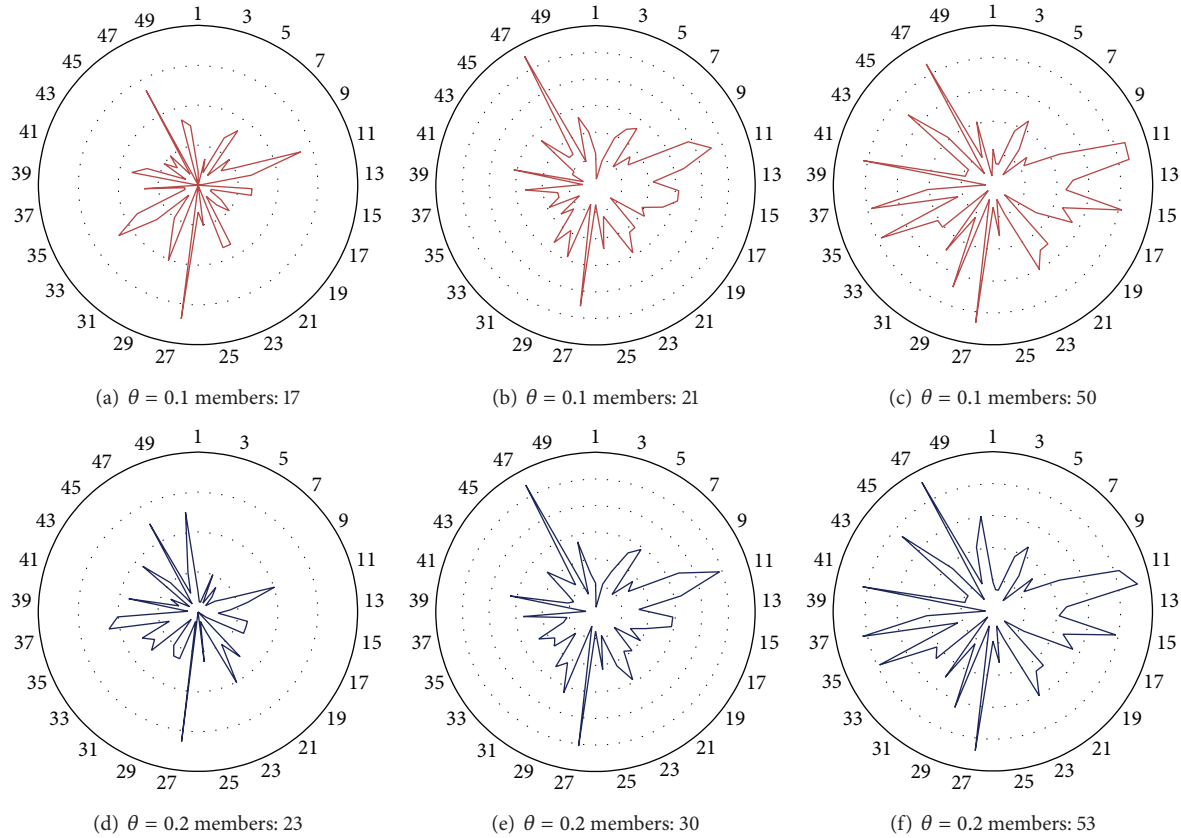
After applying TPA algorithm for each LDA-filtered network, the content community pattern for the largest community detected remains the same (see Figure 5). Despite the fact we applied different semantic filters, it is possible to see that there is no change in community content pattern. This example corroborates the idea explained above; the community content pattern is constant even if a semantic filter is applied. This can be explained for the methodology used to characterize a community, where we only included components over a threshold in our analysis. This lets us avoid the inclusion of irrelevant topics within a community, leaving only the core meaning of every post.

As an example we applied the methodology described in Section 3.6 to 2005-VC with a semantic filter $\theta = 0.1$. We characterized the TPA's output which obtained the highest modularity after ten iterations.

Table 2 shows the three largest communities detected based on the 2005-VC with LDA-Filter 0.1 by the TPA community detection approach. For each community, the five topics with the highest score are shown. In this example, we can see that, within the five most relevant topics for each community, three of them are shared by all communities ((1) *live life quotes,* (2) *profiles (forum), and* (3) *Islam in general*) and just two of them let us characterize the content generated by community members.

6. Conclusions

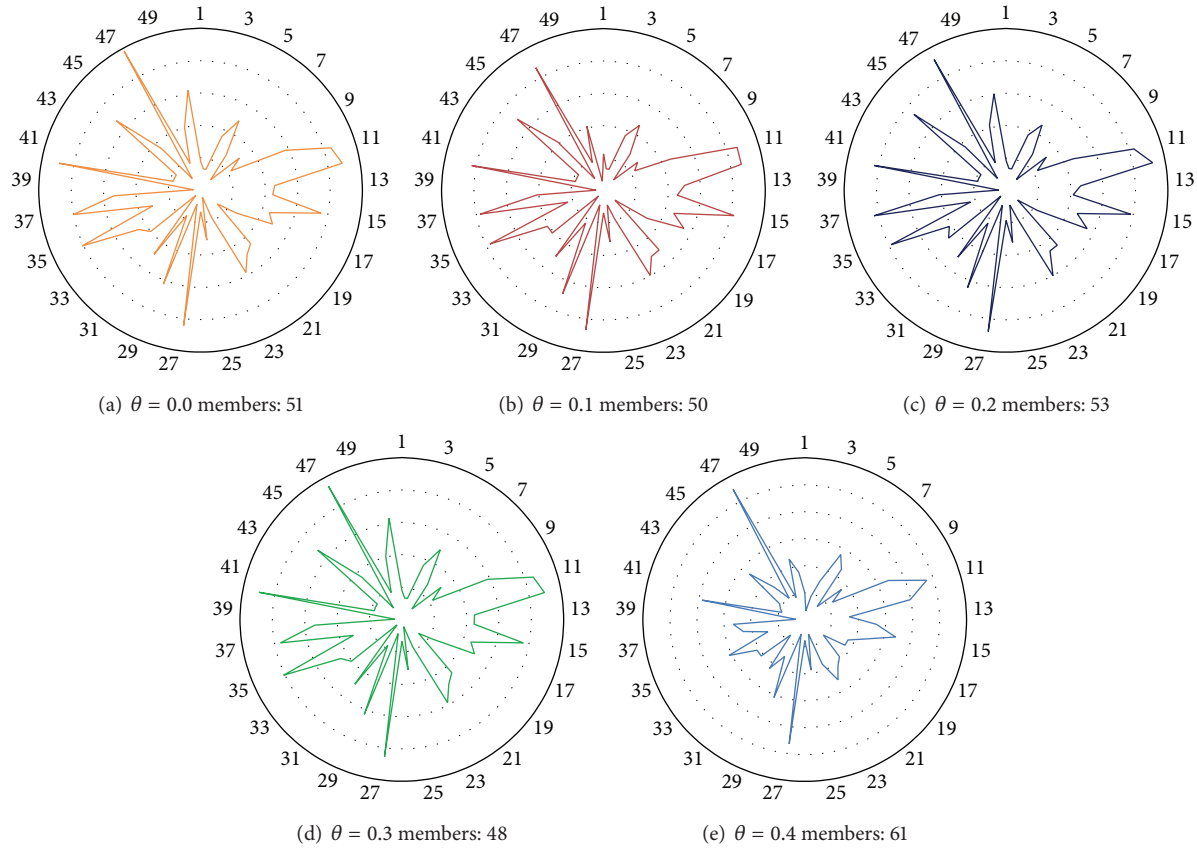
Community discovery on social networks is a hot topic which affects many different areas from recommender systems to community management or even community moderation. However, most algorithms detect hard communities, which means that every member belongs to only one discovered



(a) $\theta = 0.1$ members: 17 (b) $\theta = 0.1$ members: 21 (c) $\theta = 0.1$ members: 50

(d) $\theta = 0.2$ members: 23 (e) $\theta = 0.2$ members: 30 (f) $\theta = 0.2$ members: 53

Figure 4: Content community pattern of the three largest communities detected through TPA on 2005-VC.



(a) $\theta = 0.0$ members: 51 (b) $\theta = 0.1$ members: 50 (c) $\theta = 0.2$ members: 53

(d) $\theta = 0.3$ members: 48 (e) $\theta = 0.4$ members: 61

Figure 5: Content community patterns of the largest communities detected through TPA on 2005-VC with several semantic filters.

community. This is far from reality since most people have more than one interest; therefore, they usually belong to more than one community. This is called the overlapping community detection problem and only a few algorithms exist.

In this paper, we report our work in which we developed a new overlapping community discovery algorithm called the *topic propagation algorithm* (TPA) to tackle the problem of overlapping community detection. Our algorithm incorporates the sociological point of view that people have multiple interests and that communities reflect those interests. Thus, our algorithm takes advantage of a topic model, semantic filters, and social network representation.

We benchmarked our algorithm with COPRA, SLPA, and SLTA, which are state-of-the-art overlapping community detection algorithms. We tested all algorithms using a real-world dataset corresponding to a virtual community of interest (VCoI).

Experiments show that modularity results for TPA discovering overlapping communities outperform those of SLPA, SLTA, and COPRA. We also systematically obtained better results when modifying the time frame analyzed or the semantic filter threshold.

When our methodology is applied in order to detect overlapping communities over a VCoI, the expectation is that it will not detect subcommunity structure because all VCoI's members only share ideas and thoughts on a common interest or passion but they do not share their knowledge and expertise to learn more about a specific topic. After applying our methodology we validated the underlying theory, finding that our algorithms do not detect a better subcommunity structure when the semantic filter is increased over a VCoI. Moreover, only TPA is capable of finding a subcommunity structure on a VCoI because TPA includes semantic information to detect overlapping communities and it is able to capture groups which differ only in a few topics.

Conflict of Interests

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

References

[1] M. Mitchell, "Complex systems: network thinking," *Artificial Intelligence*, vol. 170, no. 18, pp. 1194–1212, 2006.

[2] B. Prieto, F. Tricas, J. J. Merelo, A. Mora, and A. Prieto, "Visualizing the evolution of a web-based social network," *Journal of Network and Computer Applications*, vol. 31, no. 4, pp. 677–698, 2008.

[3] Z. Sadan and D. G. Schwartz, "Social network analysis of web links to eliminate false positives in collaborative anti-spam systems," *Journal of Network and Computer Applications*, vol. 34, no. 5, pp. 1717–1723, 2011.

[4] Y. Ding, "Community detection: topological vs. topical," *Journal of Informetrics*, vol. 5, no. 4, pp. 498–514, 2011.

[5] E. Yan, Y. Ding, S. Milojević, and C. R. Sugimoto, "Topics in dynamic research communities: an exploratory study for the field of information retrieval," *Journal of Informetrics*, vol. 6, no. 1, pp. 140–153, 2012.

[6] S. Gregory, "Finding overlapping communities in networks by label propagation," *New Journal of Physics*, vol. 12, no. 10, Article ID 103018, 26 pages, 2010.

[7] J. Xie, B. K. Szymanski, and X. Liu, "SLPA: uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process," in *Proceedings of the 11th IEEE International Conference on Data Mining Workshops (ICDMW '11)*, M. Spiliopoulou, H. Wang, D. J. Cook et al., Eds., pp. 344–349, December 2011.

[8] S. A. Ríos and R. Muñoz, "Dark web portal overlapping community detection based on topic models," in *Proceedings of the ACM SIGKDD Workshop on Intelligence and Security Informatics (ISI-KDD '12)*, pp. 2:1–2:7, ACM, Beijing, China, August 2012.

[9] H. Alvarez, S. A. Ríos, F. Aguilera, E. Merlo, and L. Guerrero, "Enhancing social network analysis with a conceptbased text mining approach to discover key members on a virtual community of practice," in *Knowledge-Based and Intelligent Information and Engineering Systems*, vol. 6277 of *Lecture Notes in Computer Science*, pp. 591–600, Springer, Berlin, Germany, 2010.

[10] G. L'Huillier, H. Alvarez, F. Aguilera, and S. A. Ríos, "Topic-based social network analysis for virtual communities of interests in the dark web," in *Proceedings of the ACM SIGKDD Workshop on Intelligence and Security Informatics (ISI-KDD '10)*, pp. 9:1–9:9, ACM, July 2010.

[11] S. A. Ríos, F. Aguilera, and L. Guerrero, "Virtual communities of practice's purpose evolution analysis using a concept-based mining approach," in *Knowledge-Based Intelligent Information and Engineering Systems-Part II*, vol. 5712 of *Lecture Notes in Computer Science*, pp. 480–489, Springer, 2009.

[12] L. Cuadra, S. A. Ríos, and G. L'Huillier, "Enhancing community discovery and characterization in vcop using topic models," in *Web Intelligence/IAT Workshops*, J. F. Hübner, J. M. Petit, and E. Suzuki, Eds., pp. 326–329, IEEE Computer Society, 2011.

[13] D. Li, Y. Ding, X. Shuai et al., "Adding community and dynamic to topic models," *Journal of Informetrics*, vol. 6, no. 2, pp. 237–253, 2012.

[14] H. Shen, X. Cheng, K. Cai, and M. Hu, "Detect overlapping and hierarchical community structure in networks," *Physica A: Statistical Mechanics and its Applications*, vol. 388, no. 8, pp. 1706–1712, 2009.

[15] Z. Wu, Y. Lin, H. Wan, S. Tian, and K. Hu, "Efficient overlapping community detection in huge real-world networks," *Physica A: Statistical Mechanics and its Applications*, vol. 391, no. 7, pp. 2475–2490, 2012.

[16] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri, "Extending the definition of modularity to directed graphs with overlapping communities," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2009, no. 3, Article ID P03024, 2009.

[17] R. Muñoz and S. A. Ríos, "Overlapping community detection in vcop using topic models," in *KES*, M. Grana, C. Toro, J. Posada, R. J. Howlett, and L. C. Jain, Eds., vol. 243 of *Frontiers in Artificial Intelligence and Applications*, pp. 736–745, IOS Press, 2012.

[18] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, vol. 69, no. 6, Article ID 066133, 2004.

[19] M. Kosonen, "Knowledge sharing in virtual communities—a review of the empirical research," *International Journal of Web Based Communities*, vol. 5, no. 2, pp. 144–163, 2009.

[20] C. E. Porter, "A typology of virtual communities: a multi-disciplinary foundation for future research," *Journal of Computer-Mediated Communication*, vol. 10, no. 1, 2004.

[21] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, no. 4-5, pp. 993–1022, 2003.

[22] H. Alvarez, *Deteccion de miembros clave en una comunidad virtual de pratica mediante analisis de redes sociales y minerıa de datos avanzada (key-members detection in a virtual community of practice using social network analysis and advanced data mining) [M.S. thesis]*, University of Chile, 2010.

[23] Y. Zhang, S. Zeng, L. Fan, Y. Dang, C. A. Larson, and H. Chen, "Dark web forums portal: searching and analyzing jihadist forums," in *Proceedings of the IEEE International Conference on Intelligence and Security Informatics (ISI '09)*, pp. 71–76, IEEE Press, Dallas, Tex, USA, June 2009.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 412479, 15 pages
http://dx.doi.org/10.1155/2014/412479

Research Article

Modeling and Simulation of Complex Network Attributes on Coordinating Large Multiagent System

Yang Xu, Xiang Li, and Ming Liu

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 14 December 2013; Accepted 12 January 2014; Published 3 April 2014

Academic Editors: H. R. Karimi, Z. Yu, and W. Zhang

With the expansion of distributed multiagent systems, traditional coordination strategy becomes a severe bottleneck when the system scales up to hundreds of agents. The key challenge is that in typical large multiagent systems, sparsely distributed agents can only communicate directly with very few others and the network is typically modeled as an adaptive complex network. In this paper, we present simulation testbed *CoordSim* built to model the coordination of network centric multiagent systems. Based on the token-based strategy, the coordination can be built as a communication decision problem that agents make decisions to target communications and pass them over to the capable agents who will potentially benefit the team most. We have theoretically analyzed that the characters of complex network make a significant difference with both random and intelligent coordination strategies, which may contribute to future multiagent algorithm design.

1. Introduction

Multiagent system as a paradigm of distributed artificial intelligence has been popular in domains such as military, aeronautics, and disaster rescue. For example, large heterogeneous multirobot teams respond to large-scale natural disasters [1], performing future constructions and managing large civic infrastructures [2], deploying large groups of unmannered aerial vehicles and ground vehicles in military operations [3]. In those systems, the key is to build practical coordination algorithms to be able to control the heterogeneous agents toward their common objective. Although there have been many popular research works towards this issue [4], when state of the art applications require more and more agents, existing coordination approaches are not capable of addressing the new challenges of controlling hundreds of agents.

First of all, network structure has been a key attribute to the success of coordination algorithm design. For example, when a large team of robots are deployed, they are always sparsely distributed. With the restrictions of wireless communication coverage, agents may only be able to communicate directly with very a few of the others. Therefore, similar to human society dynamic infrastructure, each time, agents only closely coordinate with few teammates via communication networks either logically or physically, while they communicate with all the others with multihops of retransmission.

Secondly, multiagent systems are required to react to dynamic and partially observable environment with agents' independent decisions. Because of the nature of the multiagent application domains, agents are always in a complex, dynamically changing, and, in some cases, hostile environment. To react to the critical domain events, they have to intellectually adapt to the dynamic changes and coordinate themselves with their own observation and knowledge. In this process, network is critical to either make agreement on agents' joint activities or receive information to support agents' decisions when it cannot be completely observed.

Thirdly, coordination is a complex process since several distributed algorithms are required to interact to produce agile, cohesive, and efficient coordinated behavior. Typical coordination requires different aspects of subtasks interacted between networked agents, such as sharing information on external environmental events [5], allocating individual tasks for joint actions [6], and sharing exclusive resources [7].

The communication network is a major bottleneck to an efficient and scalable coordination. Existing algorithms for small team coordination, that is, blackboard [8] or specific centralized agents [9], relying on excessive communication

protocols are either infeasible, undesirable, or too expensive in large-scale teams. To reduce communication and make the decentralized coordination possible, agents should attempt to target their communications. However, it is a chicken-and-the-egg puzzle where communication is required for sharing knowledge while rich localized knowledge is indispensable to target communications.

In summary, with the expansion of the multiagent system, network has been a key factor to model team coordination. It is indicated, in related works [11], that different topologies may significantly change the performances over the same controlling protocol. The key of simulating multiagent coordination is to solve the problem of how decentralized agents are interacting over communication network, so that different aspects of coordination tasks can be realized to reach their common goal. Although popular network simulation tools such as NS2 [12] and OPNET [13] are able to simulate the performances of typical distributed system applications, they are not able to simulate the multiagent coordination in large-scale systems. We summarize the reasons as follows.

(1) Agents use proactive decision model to interact with others and dynamically choose who to communicate with, instead of traditional predefined reactive communication protocol.

(2) Agents are required to build local knowledge base and dynamic reasoning model to make decentralized coordination decision, which are not possible to be encoded in those network simulation tools.

(3) Agents are required to communicate many complex semantic data such as domain information, tasks, and sharable resource with others. But these data cannot be uniformly abstracted as communication packages as most network simulators do.

Although there have been existing multiagent team coordination simulators, most of them cannot simulate the large-scale networked multiagent system well. For example, Machinetta [14] that simulated the multiagent coordination in real domain, ignored the complexity of network structure and only modelled agents in a small team with a complete network. Framework for task analysis, environment modeling, and simulation (TAEMS) [15] uses a partial global planning (PGP) framework [16] for domain-independent decision making in handling uncertainty in the environment. It is built based on a hierarchical network architecture. Other decentralized coordination simulators, such as LA-DCOP [17], are only capable of partial coordination subtasks. In this paper, we will analyze the characters of complex networks which will make a significant difference in the multiagent coordination, and based on these observations, we put forward a novel approach to model and simulate coordination in networked multiagent systems. The simulator is capable of simulating the major aspects of coordination including information sharing, task assignment, and resource allocation. The key is to simulate how agents interact over the network so that agents can carry out their joint activity toward their common objective. The most prominent advantage of the simulator is that it allows a large number of parameters critical to the coordination to be adjusted and also allows statistics to be recorded; especially, it is able to initiate the agents in different network topologies by taking the complex network effects into consideration. These variations and records are helpful to verify the efficiency of coordinating large-scale networked multiagent systems.

2. Coordination for Multiagent System

The coordination for a multiagent system $A = \{a_1, a_2, \ldots, a_n\}$ can be described as follows. Networked agents share a top level common goal Goal (as in [18]), and achieving Goal requires achieving a number of subgoals $\{g_1, g_2, \ldots, g_i, \ldots\}$. When subgoal g_i is satisfied, the team will receive a reward reward_i. For example, subgoals of a high level goal to respond to a disaster might be to extinguish fires and provide medical attention to injured civilians. To satisfy subgoals, the team follows a set of $\text{Plan} = \{\text{plan}_1, \text{plan}_2, \ldots, \text{plan}_i, \ldots\}$ represented in a library [19]. Each plan i includes four parts and is written as $\text{plan}_i = \langle g_i, \text{conditions}_i, \text{tasks}_i, \text{reward}_i \rangle$. The first element is the subgoal g_i; the second is the conditions under which it is applicable, $\text{conditions}_i = \text{event}_1 \cap \text{event}_2 \cap \cdots \cap \text{event}_l$; the third element is the individual tasks $\text{tasks}_i = \{r_1, r_2, \ldots, r_k\}$ which are required to achieve g_i and the last part, the reward_i, is to be received by the team on successful satisfaction of g_i. Each task $r_i = \langle g_i, \text{capability}_i, \text{res}_i \rangle$ is represented by its subgoal g_i carried out by a single agent with the required certain kind of capability_i and res_i (which would be introduced below), that is, a description of the actual things for the agent to do to realize the subgoal g_i.

For example, a fire fighting plan can be modeled as follows: $\text{plan}_{\text{fire}} = \langle (\text{Fight fire at location } X), (\text{Fire alarm at } X \cap \text{Smoke at } X), \{r_1, r_2, r_3\}, (100) \rangle$. This plan requires two conditions before it is initiated: a fire alarm and smoke. After this plan is initiated, three tasks, $\{r_1, r_2, r_3\}$ need to be assigned and a reward 100 will be credited to the team. The three tasks in the plan are driving the fire truck, fighting the fire, and searching for victims; that is, $r_1 = \langle (\text{driving the fire truck}), (\text{skillful in driving truck}), (\text{fire truck}) \rangle$, $r_2 = \langle (\text{fighting the fire}), (\text{have training in fire fighting}), (\text{hose, water}) \rangle$, and $r_3 = \langle (\text{searching for victims}), (\text{none}), (\text{breathing equipment}) \rangle$. To perform r_1, an agent is required to be able to drive and have access to a fire truck which is an exclusive resource.

In a heterogeneous multiagent system, each agent may have different capabilities to perform different tasks. For example, in the fire rescuing, the capable agents may be able to drive the fire truck, fight fire, or search for the wounded. However, when performing the same task, agents with different capabilities make different performances; for example, a fireman is more capable than an inexperienced citizen in a fire fighting. Therefore, we define a function $\text{capability}(a_i, r)$ to denote the capability of agent a_i to perform task r. If the agent's capability is more than r's threshold, it is able to perform the task. A reward will be received by the system when an agent finishes a task. It is a function of an agent' capability and task, as well as the resources the agent has. Specifically,

$$\text{reward}\left(a_i, r, \text{Holds}\left(a_i\right)\right) \longrightarrow \mathscr{R}. \tag{1}$$

The function Assigned$(a_i, r) = 1$ if agent a is assigned to task r; otherwise it is equal to 0. At any time, a task could only be assigned to at most one agent; that is, $\sum_{a_i \in A} \text{Assigned}(a_i, r) \leq 1$.

Networked agents always require sharable resources to perform tasks. These resources, $R = \{\text{res}_1, \ldots, \text{res}_m\}$, are discrete and nonconsumable. Agent a_i has exclusive access to resources $\text{Holds}(a_i) \subseteq R$. Only one agent may hold a resource at any point of time; that is, for all $a_i, a_j \in A, a_i \neq a_j$, $\text{Holds}(a_i) \cap \text{Holds}(a_j) = \emptyset$.

The coordination problem is to maximize the rewards to the team, while minimizing the costs of coordination. The overall reward is simply

$$\sum_{a_i \in A} \sum_{r_i \in \text{task}_i} \text{Assigned}(a_i, r_i)\, \text{Reward}(a_i, r_i, \text{Holds}(a_i)). \quad (2)$$

The costs of coordination can be very general and in some cases difficult to define. Here we are specifically concerned with only the volume of communication.

3. Networked Agents

According to the characters of networked agents that they can only directly interact or communicate with very few of others over the network, the organization of the multiagent system can be abstracted as an undirected network $G = (A, E)$ as shown in Figure 1 [20]. In this model, A is the agent set, and E is the set of edges where if $e_k = \langle a_i, a_j \rangle$, a_i and a_j can communicate with each other directly. Moreover, $n(a_i)$ is defined as the set of a_i's neighbors that it can directly communicate with.

G could be organized as different network topologies based on the different properties of social networks. In our simulation, we are mainly interested in four of them: random network, grid network, small world network, and scale-free network. Preliminary studies [21] found that each topology encodes the following different fundamental properties.

(1) Average degree: $\bar{d} = (1/N) \sum_{a_i \in A} |n(a_i)|$.

(2) Degree distribution: $p(k) = \Pr[d = k]$.

(3) Average distance: $d_{\text{ave}} = (1/N(N-1)) \sum_{\forall a_i, a_j \in A} \text{distance}(a_i, a_j)$. $\text{distance}(a_i, a_j)$ defines the least number of hops to communicate between a_i and a_j.

Different network topologies can be described according to the properties above. In our approach, we mainly take consideration of the effects of small world and scale-free network and use degree distribution to express the scale-free effect and their average distance to express the small world effect.

4. Networked Coordination Model

As the coordination problem modeled in Section 2, agents' interactions over network are to carry out different aspects of coordination subtasks to reach their common goal including information sharing, task allocation, and resource sharing. $\text{INF} = \{\text{event}^1_{1,1}, \text{event}^2_{1,1}, \ldots, \text{event}^k_{i,j}, \ldots\}$ is the set of all possible domain events, $\text{TASK} = \{\text{task}^1_{1,1}, \text{task}^2_{1,1}, \ldots, \text{task}^k_{i,j}, \ldots\}$ is the set of potential available joint activities, and $\text{RESOURCE} = \{\text{res}_1, \text{res}_2, \ldots, \text{res}_k, \ldots\}$ is all available exclusive resources in team A. Multiagent coordination is defined as $\Xi = \text{INF} \cup \text{TASK} \cup \text{RESOURCE}$. At each time, the coordination element interacted between a pair of agents over the network can be defined as $tc \in \Xi$, which is either a domain event, a joint activity, or an exclusive resource. By encapsulating tc, we can define the data structure of the interaction between agents in the network as Token.

Figure 1: An example of large-scale multiagent network.

If we assume that agents will handle their own jobs well, the objective of communicating token through networked agents is to find capable agents who can perform the desired task, use resource, or obtain the information, so that a subgoal g_i will be achieved. The structure of any token Δ_j is written as $\Delta_j = \langle \text{ID}, tc, \text{path}, \text{threshold} \rangle$. We assume that token cannot be duplicated. When an agent is holding Δ_j, it takes over control $\Delta_j.tc$ and will release $\Delta_j.tc$ if it is passed. $\Delta_j.\text{path}$ records the sequence of agents where Δ_j has been passed. $\Delta_j.\text{path}$ is also used as stop condition for information and task tokens if $|\Delta_j.\text{path}| > \text{Length}$. Length is predefined that Δ_j is allowed to be passed to others before being stopped.

Threshold generalizes a threshold for resource and task tokens that can be accepted by any agent. Only when the capability or calculated desire of an agent is higher than $\Delta_j.\text{threshold}$, it can be accepted. Threshold can be dynamically adjusted to balance the needs of the whole system and produce improved allocation performance. For example, if a resource is highly competitive, it will enhance the threshold to travel across the network to find a "better" accepter. Otherwise, if it has traveled a long way over the network, the threshold should be decreased to help to find anyone who can make use of it, even if it may not be the best.

5. Multiagent Communication Decision

As explained in Section 4, the objective of communicating token is to find capable agents who can perform the desired task, use resource, or obtain the information if themselves are not capable of making use of it. Similar to human society,

```
(1) ApplicableTasks = [], OwnTasks = [], Holds = [], Tokens = [];
(2) while (true) do
(3)     for (α ∈ agent a_i, α ∉ ApplicableTasks) do
(4)         if (Applicable(α)) then
(5)             ApplicableTasks.append(α);
(6)             Tokens.append(CreateTokens(α));
(7)         end if
(8)         Tokens.append(recvTokens());
(9)     end for
(10)    for (Δ ∈ Tokens) do
(11)        if (Δ is TaskTokens(α)) then
(12)            if (GetCap(α) > Δ.threshold) then
(13)                if (α ∉ OwnTasks) then
(14)                    OwnTasks.append(α);
(15)                end if
(16)            else
(17)                SendToNeighbour(Δ);
(18)            end if
(19)        else if (Δ is ResourceTokens(r)) then
(20)            Δ.threshold += δ;
(21)            if (GetNeed(r) > Δ.threshold) then
(22)                if (r ∉ Holds) then
(23)                    Holds.append(r);
(24)                end if
(25)            else
(26)                Δ.threshold −= δ;
(27)                SendToNeighbour(Tokens);
(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(CreateTokens(r));
(38)            end for
(39)        end if
(40)    end for
(41) end while
```

Algorithm 1: General process of agents' communication decision.

agents always forward the incapable tasks and resources across the network to find the best accepters. The key of the coordination is to optimize their interactions, so that the best capable agents could be reached as fast as possible with the least communication cost as well as assignment delay.

Algorithm 1 [22] describes the general decision process for decentralized networked agents. In this algorithm, agents first check whether new tasks have become applicable. If so, the agent will embed the task to a token and add it into its token list Tokens, which contains the tokens to be processed (lines 3–7). Next, the agent will receive all the tokens passed from other agents (line 8). It then processes all the tokens in the Tokens. If a token represents a task, the agent will accept the task if its capability to perform that task is higher than token's threshold (lines 11–14); otherwise, the agent will choose a neighbor to pass that token to (line 17). If the token is a resource token and the agent's need for that resource to perform his waiting tasks is higher than token's current threshold, this resource will be held; otherwise, it is passed to a neighbor (lines 19–27). Note that when a token is sent, it will be removed from that agent's list. Finally, the agent will check whether any task which is appended can now be executed (lines 30) and release any resources from completed tasks (lines 32–37).

The key of the decision process is the function of SendToNeighbour( ), which is to pass a token to a neighbor who either needs it or knows who does. In our simulation model, we primarily use two algorithms in which tokens are routed either intelligently or randomly. By building a local decision matrix of an agent a for a token Δ as $P_a[\Delta]$.

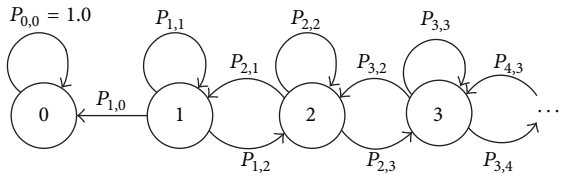


Figure 2: Markov chains model on tokens' movement between agents.

$P_a[\Delta, b] \rightarrow [0, 1], b \in n(a)$ represents a's decision of the probability of passing token Δ to neighbor b. In random routing algorithm, for each $b \in n(a)$, $P_a[\Delta, b]$ is uniform. However, when agent a has built a knowledge base for its received tokens, agent a can route Δ in a smart way. By taking the advantage of tokens that may be related in domain knowledge, previous received tokens can be used to update local probability model that receiving a token predicts that the sender may request a related token. For example, the sender of the information token "I am hungry" denotes that it may be interested in a resource token "pizza" to feed a hungry agent. Therefore, each token is used to improve the routing of other tokens which will lead to a dramatic performance improvement. The key algorithm has been detailed and described in [23] and has been encoded in our simulator.

6. Complex Network Effects in Coordination Efficiency

In this section, we briefly analyze how complex network effects will change the team coordination performance. We modeled the token's random walk as a finite Markov chain and it is time reversible [24]. For a specific token movement, we can define different states.

As it is shown in Figure 2, state s_i defined the state that the token moves to an agent with the shortest distance of i from the sink agent. Specifically, probability $P_{i,j}$ defined the probability of the token being passed from states i to j. Because the token can only move one step for each horizon, $P_{i,j} = 0$ except for $j \in \{i - 1, i, i + 1\}$. Therefore, for a state $s_i \neq s_0$, the token may move closer to the destination ($P_{i,i-1}$), stay on the same level ($P_{i,i}$), or move far away ($P_{i,i+1}$). When the token reaches state S_0, it will be stopped at the destination and the probability $P_{0,0} = 1$. For example, suppose that u is the initial probability distribution of the token being in state S. According to the theory of Markov chains, we can calculate the probability that the token reaches the sink agent after n steps as $P_S^n = u \times P^n$.

As the agents transmit the tokens randomly to anyone of their neighbors, there will be different distances between the source and destination. Figure 3 shows the relative rates of $P(s_i, s_{i-1})$ (marked "Close"), $P(s_i, s_i)$ (marked "Same"), and $P(s_i, s_{i+1})$ (marked "Further") for scale-free and random networks. Notice that we average $P(s_i, s_j)$ over each node at distance i, though this will vary from node to node. 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," different colors represent the corresponding proportions, and the sum of them is 1. Note that the closer the token is to the target, the more likely it will be randomly passed further from the target. Correspondingly, the further the token is from the target, the more likely it will be passed closer to the target in random movement. Moreover, since the figures show different distributions, their token movement characteristics are likely to be different.

Figure 3(a) shows a typical scale-free network. The state probability transition matrix P in this network topology is

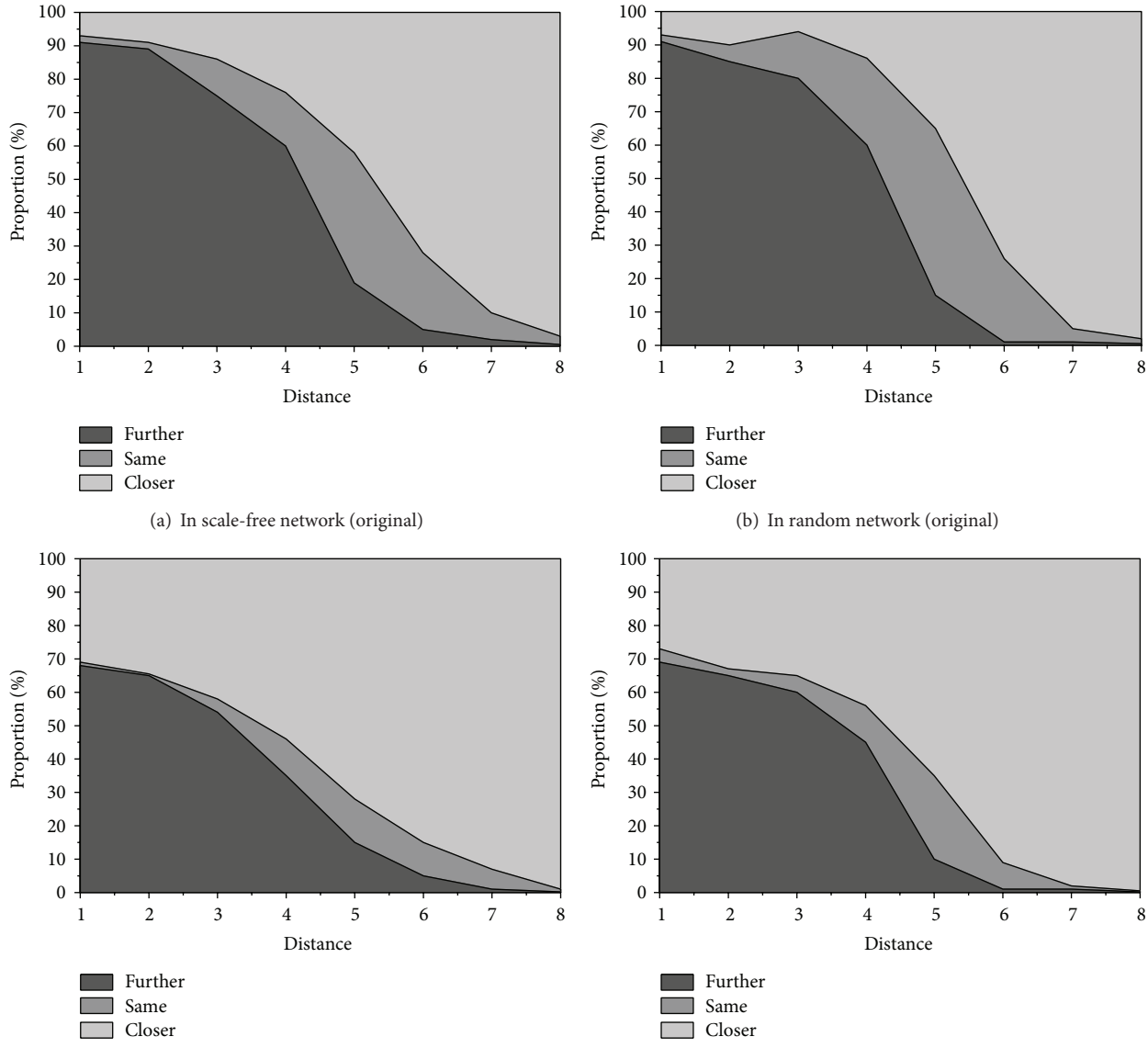
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0.01 & 0.89 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.15 & 0.1 & 0.75 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.25 & 0.25 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.45 & 0.35 & 0.2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.7 & 0.25 & 0.05 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.89 & 0.01 & 0.1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.98 & 0.02 \end{pmatrix}. \quad (3)$$

Figure 3(b) shows a typical random network. The state transition probability matrix P is

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0.02 & 0.88 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.15 & 0.05 & 0.8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.15 & 0.25 & 0.6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.45 & 0.4 & 0.15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.75 & 0.24 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.96 & 0.03 & 0.01 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.99 & 0.01 \end{pmatrix}. \quad (4)$$

Suppose that the same token's initial distribution is u = [0.25 0.25 0.15 0.15 0.1 0.05 0.04 0.01] and after a token random movement for 1000 steps, the state probability distribution for a scale-free network is [0.8287 0.0030 0.0198 0.0605 0.0675 0.0194 0.0011 0.0000], where, in 83% of cases, this token has reached the destination agent. On the other hand, the state probability distribution for a random network is [0.0171 0.0084 0.0575 0.3337 0.4904 0.0920 0.0009 0.0000], where, in only about 1.7% cases, this token has reached the destination agent. It proves what we observe in figures. The efficiency of information transmission in scale-free network is higher than in random network.

On the other hand, in teams, tokens do not simply move randomly from agent to agent when agents can build a better decision model [23]. Therefore, tokens are able to move faster to get accepted based on agents' local knowledge. To model that such movement is not completely random but is biased towards the target, we use a parameter β to make $P(s_i, s_{i-1})$ larger and $P(s_i, s_{i+1})$ smaller. However, this bias should be stronger as the token moves nearer to the target location because agents such as their neighbors are more likely to know the target. We model this by using $\beta(i) = 1/e^{\alpha_i}$. Informally, one can think of β as the total learning of the team about the team state and α as how much more agents "near"



(a) In scale-free network (original)

(b) In random network (original)

(c) In scale-free network (with biased intelligent movement)

(d) In random network (with biased intelligent movement)

Figure 3: The relative proportions of links that lead closer to, keep the same distance, or move further from the target agent, as the distance to it is varied [10].

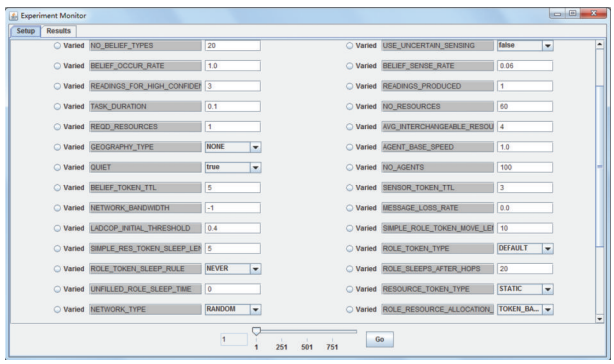


Figure 4: CoordSim allows a large number of parameters to be varied and also allows statistics to be recorded.

an agent know about it than agents "far" from it do. Using α and β, the Markov Chain state transitions can be rewritten as

$$\begin{aligned}\check{P}(s_i, s_{i-1}) &= P(s_i, s_{i-1}) + (1 - \beta(i))\, P(s_i, s_i) \\ &\quad + (1 - 2\beta(i))\, P(s_i, s_{i+1}), \\ \check{P}(s_i, s_i) &= P(s_i, s_i) - (1 - \beta(i))\, P(s_i, s_i), \\ \check{P}(s_i, s_{i+1}) &= P(s_i, s_{i+1}) - (1 - 2\beta(i))\, P(s_i, s_{i+1}).\end{aligned} \tag{5}$$

In addition, an α value of 0.97 is used to bias the links towards moving towards the target agent. Figures 3(c) and 3(d) show the effects on the scale-free and random distribution respectively. When tokens are especially close to the target,

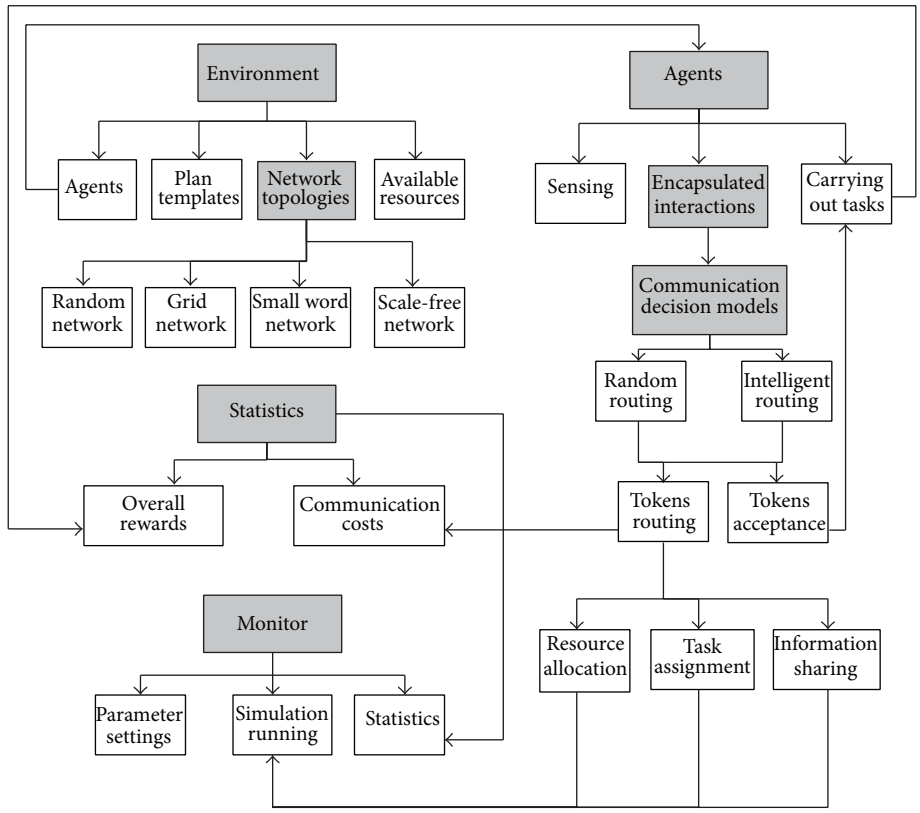


Figure 5: The architecture of CoordSim simulator.

they are much more easier to get closer to the target and the agents will be the more efficient to deliver tokens.

Compared with the scale-free and random networks in random walk, if a token's initial distribution is the same as in the last section (i.e., u = [0.25 0.25 0.15 0.15 0.1 0.05 0.04 0.01], after 1000 movements), we will find that the state probability distribution is [1.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000] and that the token will surely reach the destination. Moreover, after 100 moves, the state probability distribution is [0.9290 0.0029 0.0094 0.0242 0.0262 0.0078 0.0005 0.0000]. This means that, in 93% of cases, the token has reached the destination, which is better than token's random moves over 1000 steps in the same network topology. The theoretical analysis above verifies that the multiagent coordination will have a significant difference in different network topologies with both random and intelligent coordination strategies, so it will be an important factor we will consider in Simulator Design.

7. Simulator Design

In this section, we describe the design of our simulator called CoordSim, shown as Figure 4. This simulator is capable of simulating the major aspects of coordination, including information sharing, task assignment, and resource allocation. CoordSim abstracts the environment by simulating only its effects on the system. Agents cannot receive any domain knowledge unless they sense it themselves or are "told" by a teammate over network. The physical resources required for tasks are simulated and allow only one agent to access them at any given time. There is no cost for transferring resources, and resources cannot be consumed or lost. The tasks are distributed randomly in an open environment. All agents are allowed to "think" and "act" at each time step, although the effects of their "actions" are abstractly simulated and they only take one time step.

Reward is simulated as being received by the team when a task is allocated to one agent. The most prominent advantage of CoordSim is that it allows a large number of parameters to be adjusted and also allows statistics to be recorded, such as the number of rewards and token movements. There are more than 20 parameters that can be varied, covering the major aspects of large heterogeneous coordination, and these configurations and records help to verify the validation of our approach.

7.1. System Architecture. The system architecture of CoordSim is illustrated as in Figure 5. It mainly consists of five modules: the environment, agents, communication decision models, as well as the monitor, and statistic model with user interface, which have been described in dark boxes. In environment module, it initiates the agents deployed in the environment, plan templates that could be activated to model agents' joint activities, the network that agents are originally

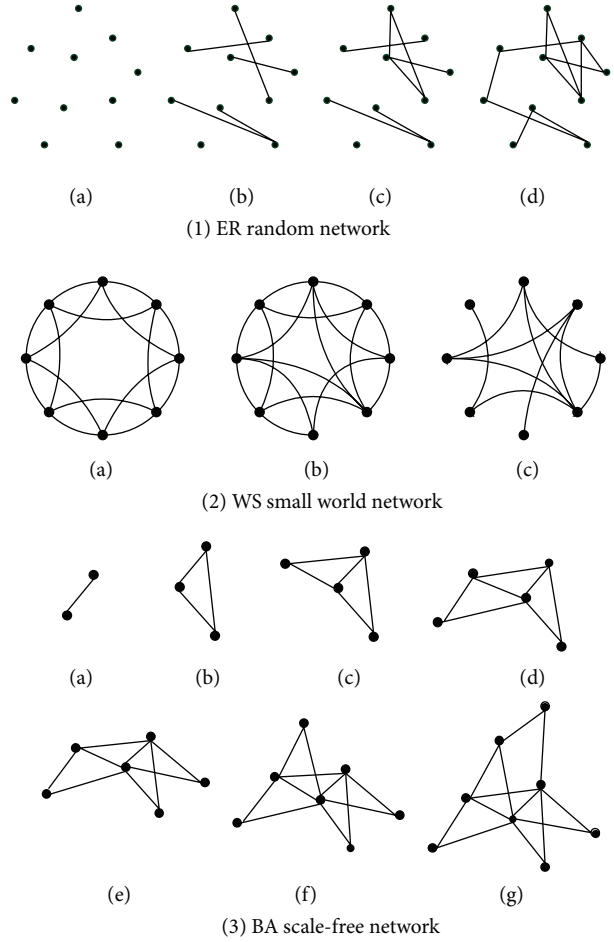


Figure 6: Process of building random network, small world network, and scale-free network.

connected, and the available resource for agents. According to the analysis of complex network effects in multiagent coordination, in our simulation, different topologies are taken into consideration, to allow users to simulate and analyze the influences of the network structures. The networked agents are organized as four typical network structures: random, grid, small world, and scale-free network. Please note that although we take the agents' mobility into consideration, the moving speed is far less than communication speed.

In the agent module, based on the knowledge of initial environment, the coordination processes include sensing environment, network interactions, and carrying out its own tasks. Especially, by encapsulating coordination into tokens, the network communication decision model is built with two algorithms: random routing and intelligent routing. Creating agents also includes the initialization of their heterogeneity. The key of the simulation is that agents initiate plans if they have gained enough information, either sensed or shared, and they may take different tasks and acquire required resources to achieve the system's common goal. Therefore, if the tokens encapsulated resources, tasks, or information are accepted by agents, they can perform the tasks with required resources and will credit the system with rewards. Otherwise, the token will not be stopped, and the agents will forward the tokens to the neighbor with communication costs.

7.2. Simulation of the Network. In CoodSim, the network design is the most important part to network centric agents. Similar to human group where members typically maintain a small number of associates, when the system scales up, the network shows the attributes of the complex network. In this paper, we are mainly interested in the performances when networked agents are organized as four popular complex network structures: ER random network [25], grid network, WS small network [26], and BA scale-free network [21]. The network construction processes of setting up the organizations of N network agents with an average of K neighbors (for the algorithm design, K is always an even number) is shown in Figure 6. The total number of links in the network is $K \times N/2$.

(i) ER random network: the network starts from nodes with empty links. For each time step, two nodes are picked up randomly and connected with a predefined probability $K/(N-1)$.

(ii) Grid network: each node only connects with $K/2$ adjacent neighbors. The process of construction of grid network is straight forward as the one in [27].

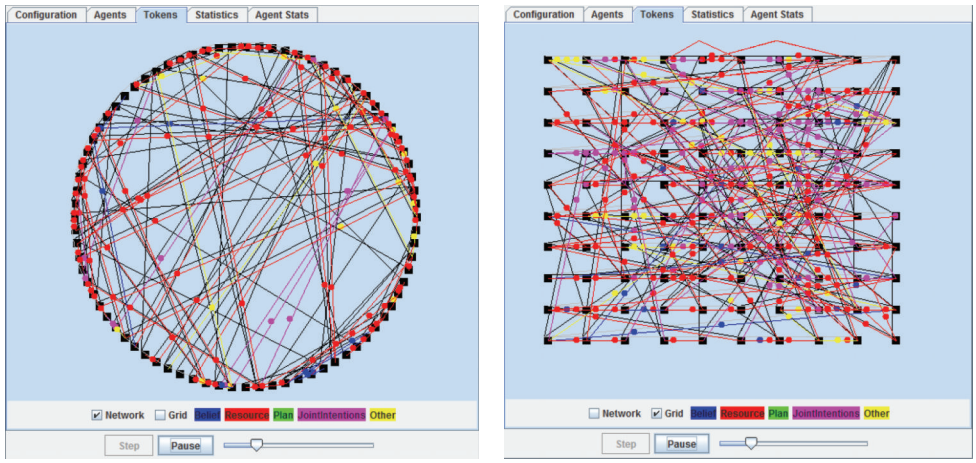
(iii) WS small world network: based on a regular grid network above, WS small world network is built with a relink process. In this process, every link in the network is selected with a probability P. If selected, one node of the link is kept but the other node will be replaced with a randomly picked node in the network.

(iv) BA scale-free network: the network starts with a fully connected $K/2 + 1$ nodes. If the degree of each existing node i is d_i, for each pending added node, the probability of linking it to the existing node i is $k_i / \sum_{i=1}^{m}(k_i)$.

In general, grid networks have the feature of clustering but do not present the small world effect [28]. The ER random network has small average path length, but it is not clustered relatively. The small world network is the transition from grid network to the ER random network, which has a small average distance and a high degree of clustering [29]. Because of the major hubs, the scale-free network exhibits strong robustness to failure, but sometimes it may weaken the performance of scale-free networks [30].

In addition to the network design, we display the interaction between networked agents with a straightforward way which is shown in Figure 7.

7.3. Performance Evaluation and Statistics. Because of the complexity of the scalable-team coordination problem, there are many parameters that can be adjusted and tested. In our simulation, by varying more than 20 parameters and covering the major aspects of scalable coordination, we evaluate the system performance with both rewards and communication cost. Specifically, in this paper, we mainly interested in the parameters that contribute most to networked centric agents.



(a) (b)

Figure 7: The interface for displaying interaction between networked agents.

Table 1: Characters and settings of different application domains.

Scene	Team size	Plans	Task\plan	Res	Cap types	Info type
WASM	100	30	2	50	1	60
USAR	50	20	5	100	5	40
RoboCup	50	20	2	100	2	40
Large agent team	500	100	2	150	2	200

(i) If the communication cost is high, the team will receive a much lower reward when reach the same team goal.

(ii) The communication processing rate is the number of tokens that an agent can pass per second. If there are more tokens, they have to be sent in the next second.

(iii) Real-time control means that a task or a plan has to be carried out in a short period of time.

(iv) Task importance describes how the reward is calculated. For example, the goal of a UAV team is to destroy as many enemy vehicles as possible, and the missing one is not a major concern. For USAR, however, missing one victim means that the team has failed, so every task and plan must be allocated.

(v) The team size is the number of agents in the team.

(vi) A heterogeneous team describes the number of capability types among team members.

(vii) The exclusive resource requirement describes the number of required resource types.

(viii) Plan complexity defines the number of preconditions to activate each plan.

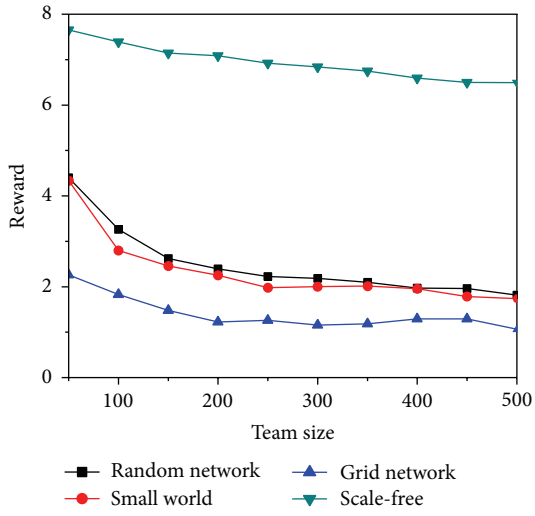
8. Simulation and Results

In this section, based on our simulator design, we briefly introduce the simulation of the networked multiagent system. The objective is to verify that our model is able to simulate the coordination of a few hundred networked agents.

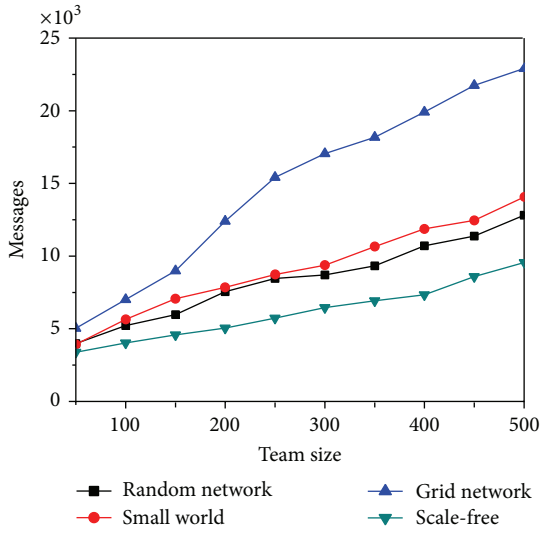
8.1. Multiagent System Coordination. Because of the complexity of large-scale multiagent coordination, we typically choose four typical application domains: urban search and rescue (USAR), controlling wide area search munitions (WASMs) [31], RoboCup [32], and large agent team in strategy game and scheduling [33]. For example, in USAR, coordination is mainly focused on heterogeneous teams and there are only one or two tasks for each of plan templates. These domains are summarized according to the features listed in Table 1.

The experiment is to testify whether different complex network properties may influence the performance of multiagent systems with intelligent token routing algorithm in different application domains. Four complex network topologies have been tested and we have presented them in two groups of experiments. In the first group, we range the multiagent system size in each domain from 50 to 500 with on average 4 neighbors for each agent. In the second one, we fix the system size at 100 and the average number of associates for each agent varies from 2 to 10.

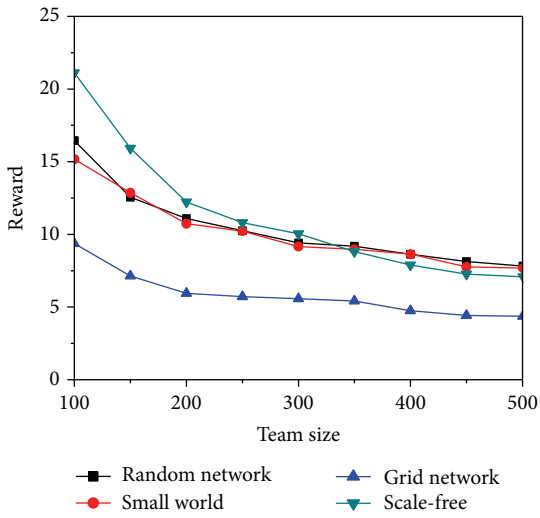
The first experiment's results are shown in Figure 8. In most domains, because of the property of the small world effect, the random network and the basic small world network outperform the grid-based network. In some domains, because of the property of the scale-free effect, the scale-free network may outperform the other three network topologies. The reason is that a few agents can act as hub nodes and obtain more knowledge of how to intelligently pass a given token in the right direction. On the other hand, in some domains such as RoboCup, the scale-free network produces more tokens



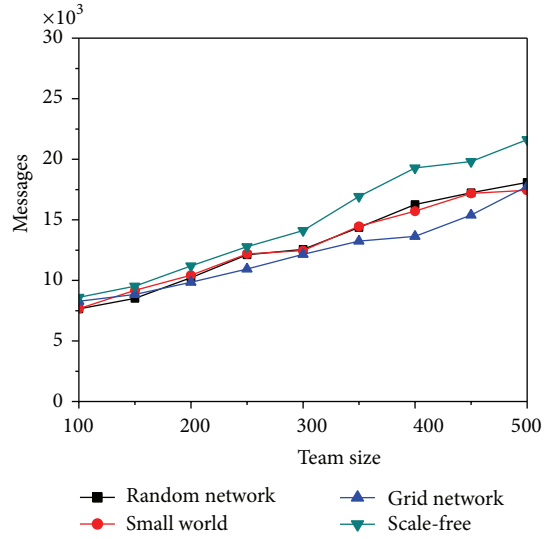
(a) Reward in WASM domain



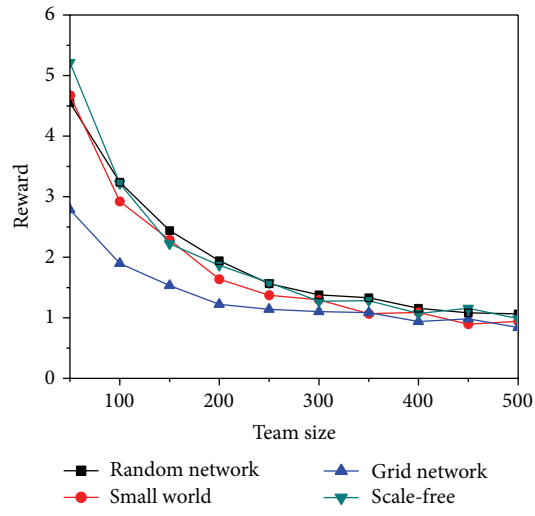
(b) Communication in WASM



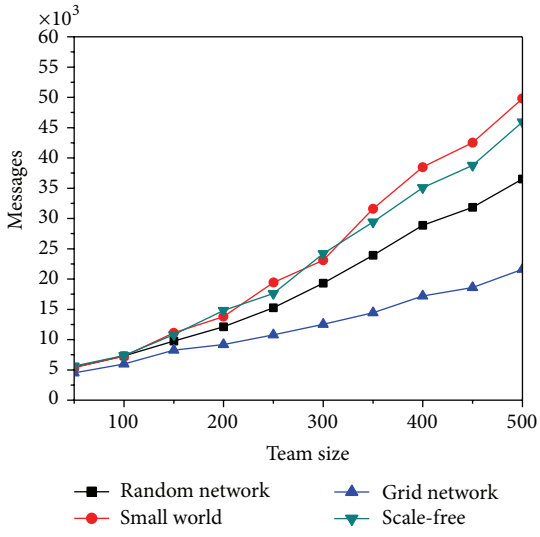
(c) Reward in USAR domain



(d) Communication in USAR

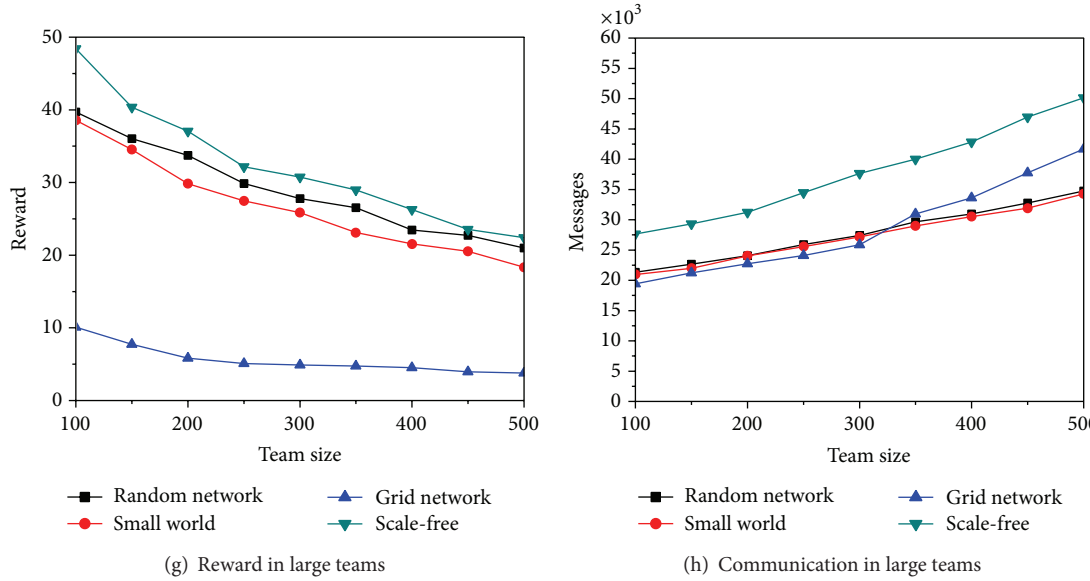


(e) Reward in RoboCup domain



(f) Communication in RoboCup

Figure 8: Continued.



(g) Reward in large teams

(h) Communication in large teams

FIGURE 8: Effects of complex networks on different scale multiagent systems.

and receives fewer rewards. The reason is that the hub nodes maintain many neighbors but the average tokens from each neighbor may be decreased. Therefore, the hub nodes cannot build a better model based on the previous tokens that they have received.

The results of the second experiment are shown in Figure 9. On average, all of the network topologies increase their performance when agents keep an average of four neighbors instead of the average of two. But when the average number of neighbors increases except the grid network, the other network topologies' performances become worse. Therefore, when the number of neighbors per agent increases, the average number of tokens from each neighbor is decreased and the agents cannot maintain better decision models based on previously received tokens. This is most prominent in the hub nodes in a scale-free network that the scale-free network's performance decreased so quickly. Conversely, the grid-based network enhances its performance in two ways. First, the number of neighbors is a constant number for each agent. No agents will have a large number of neighbors. With the increasing number of neighbors, the average distance in the grid-based network is drastically decreased. In this way, it is much easier to intelligently pass tokens quickly to their destinations. Therefore, the small world effect is important on system efficiency.

8.2. Communication Transmission over Network. In the second simulation, we analyzed and simulated the communication transmission mechanism based on our designed probability decision model. In these experiments, we used a system with 400 agents and on average, each agent has four neighbors. One agent is randomly chosen as the source of a token i and another is randomly picked as the sink. The sink agent will firstly send out 20 related tokens j with rel(i, j) and each will move MAX_STEPS = 50. Next, the source agent sends out token i and we measure how many tokens (communication costs) it takes to send i to the sink agent.

In Figure 10(a), we first varied the rel(i, j) from 0.5 to 1. The efficiency of token transmission is improved in all networks with intelligent routing model rather than randomly passing (with rel(i, j) = 0.5) with scale-free network having the best performance.

In Figure 10(b), we varied the number of related token j from the sink agent from 5 to 45. rel(i, j) is set to 0.9. It shows that communications will become stable after the number of related token j is more than 15. This result also shows us that a few tokens are enough for agents to build local decision model. Scale-free network needs less communications as well.

In Figure 10(c), we run experiments with rel(i, j) = 0.8 and each agent has on average 2 to 8 neighbors. It shows that more neighbors may not be helpful to improve network transportation with the same conclusion in Section 7.1.

In the last experiment shown in Figure 10(d), we ran experiments with different system sizes. The network transmission efficiency is measured as the percentage of agents involved for token passing:

$$\text{percentage} = \frac{\text{agents involved in token delivery}}{\text{Total \# of agent team}}. \quad (6)$$

The experiment's results show that with different sizes of systems, the efficiency keeps steady. The percentage of random network changes slightly and its communication keeps high while scale-free network keeps least communication.

9. Summary and Future Work

In this paper, we presented a novel approach to model and simulate complex network attributes in a networked multiagent system with hundreds of agents. The key is that in typical large multiagent systems, sparsely distributed agents can only

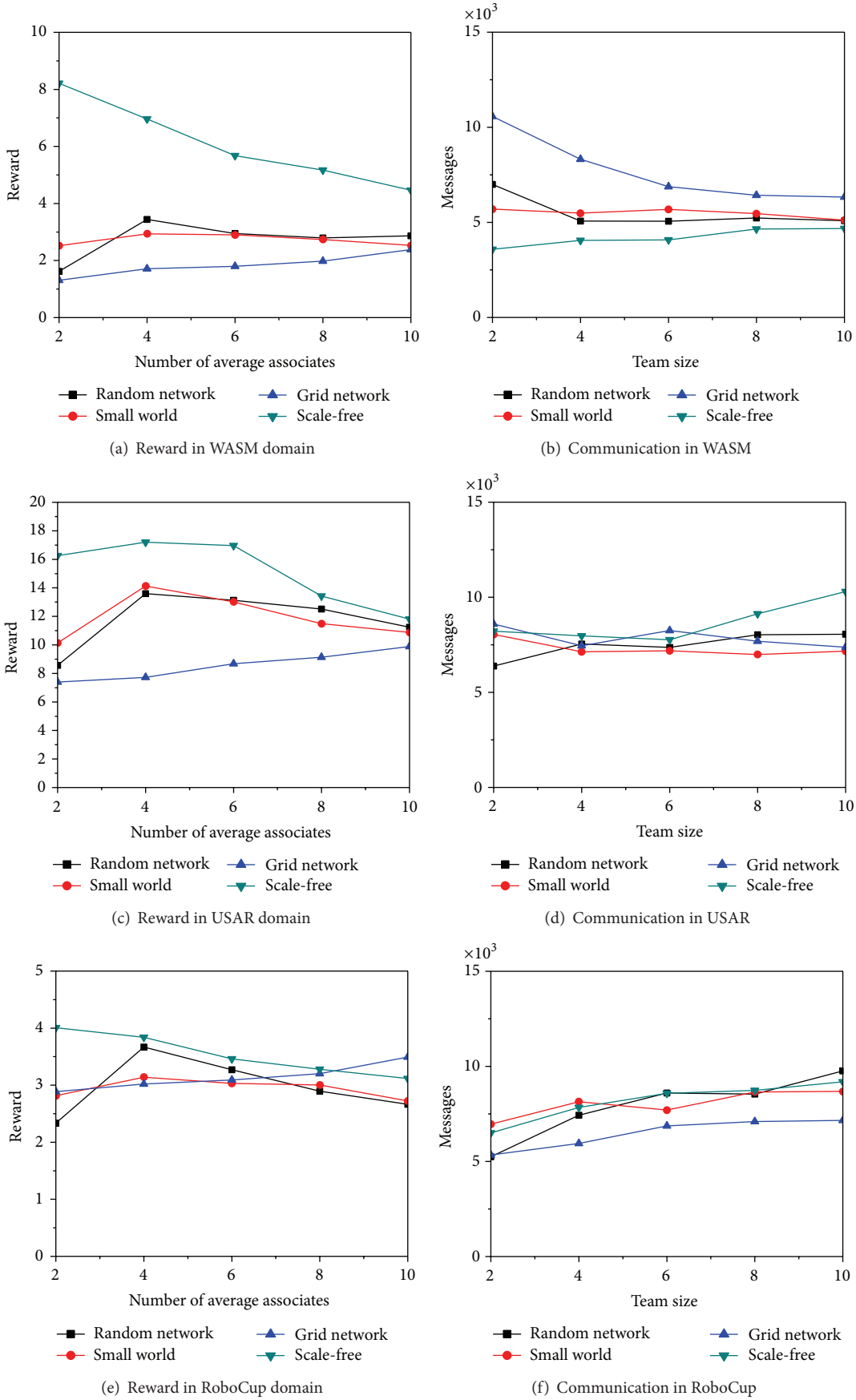
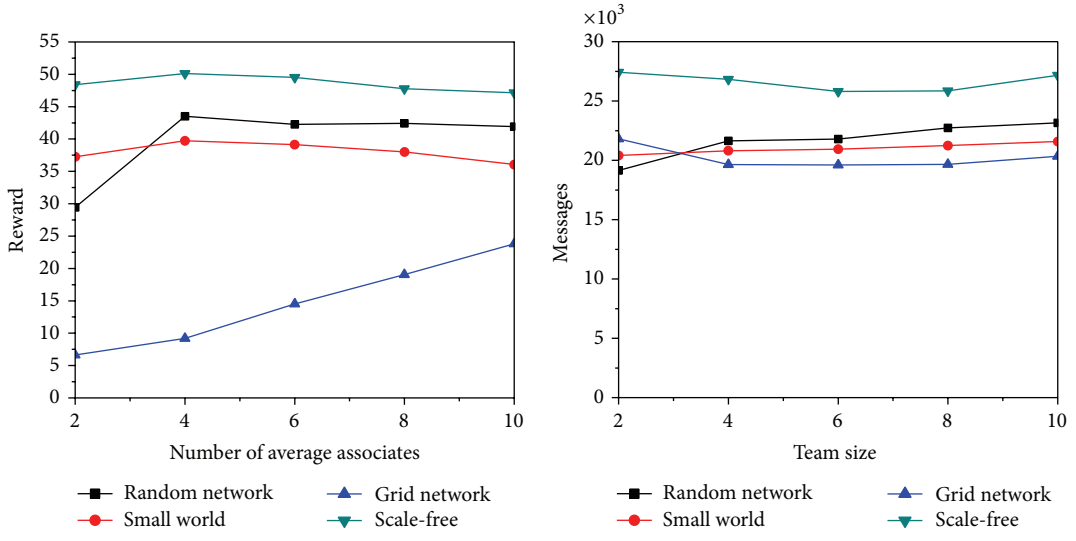
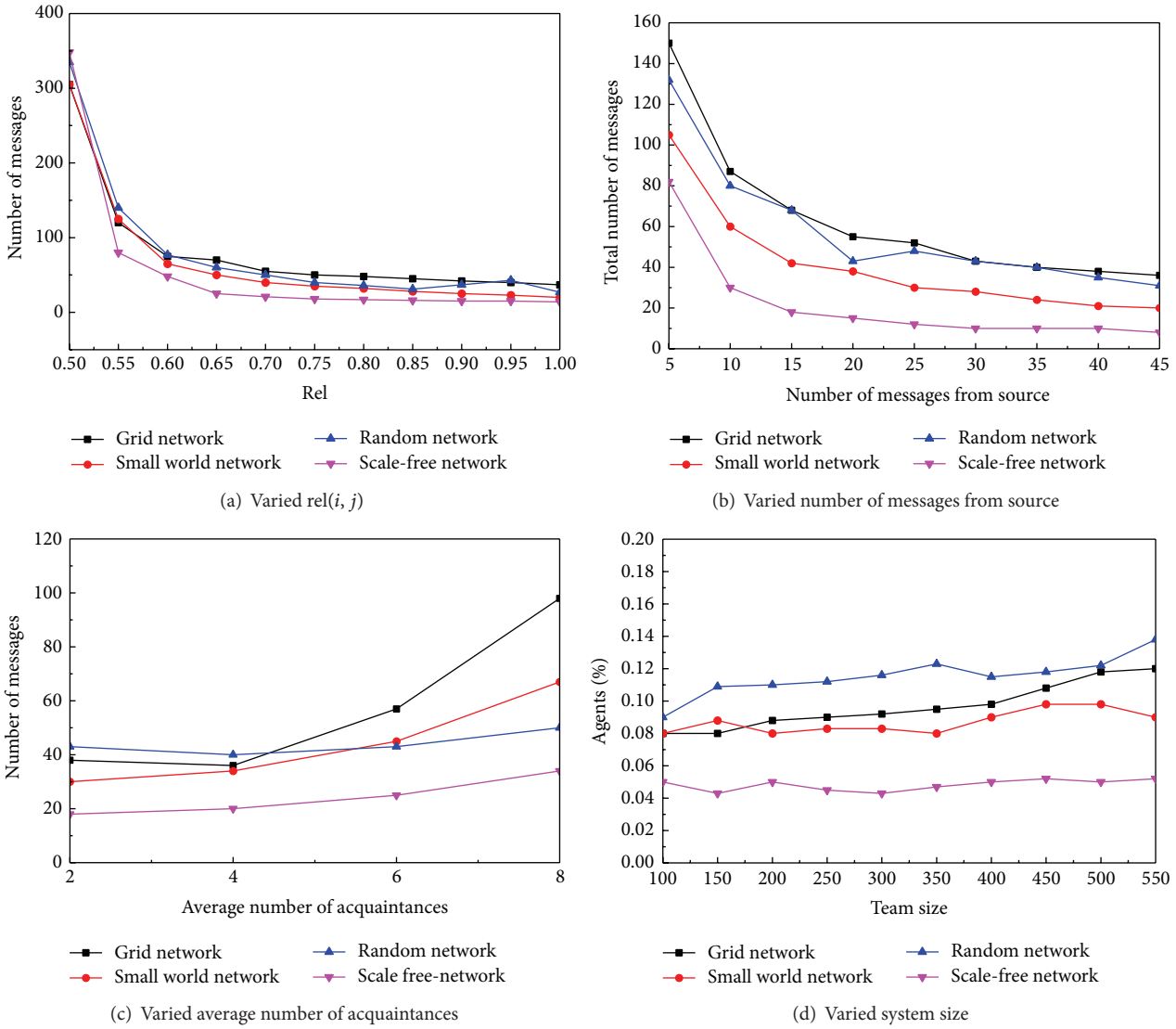


Figure 9: Continued.



(g) Reward in large teams (h) Communication in large teams

Figure 9: Effects of complex networks on different numbers of average associates.



(a) Varied rel(i, j) (b) Varied number of messages from source

(c) Varied average number of acquaintances (d) Varied system size

Figure 10: Communication transmission over different network topologies.

communicate directly with very few of the others and the network can be modeled as an adaptive complex network. By abstracting coordination and encapsulating it into tokens, we can model the interactions between networked agents and simulate them over the network. By building a simple local decision model on how to retransmit the coordination encapsulated in tokens, we can implement an integrated algorithm for decentralized agents to carry out the major aspects of coordination including sharing information, allocating tasks, and sharing resources; furthermore, we have taken a full account of the complex network effects and provide different topologies. All the designs have been encoded into our simulator CoordSim to test how the network structure affects the coordination performance. We discovered that in a typical networked system, small world effect uniformly helps the networked coordination, while scale-free effect by introducing high degree nodes may cut down the average distance of the network to help improve the performance but it may also result in the network congestion in the hub nodes to exacerbate the coordination efficiency.

While this work represents an important step towards simulating large decentralized networked multiagent system coordination, much work remains to be done. Critically in the simulation, we used the simplest instantiations of the algorithms, ignoring many performance enhancing techniques proposed in the literature which may be important to simulate real application domains. We intend to extend CoordSim to implement some of these extensions, for example, individual failures and agents' different mobility, which may impact performance in a large complex networked multiagent system.

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 in part by National Natural Science Foundation of China nos. 61370151, 61202211, and 60905042 and the Fundamental Research Funds for the Central Universities no. ZYGX2011X013.

References

[1] E. Domnori, G. Cabri, and L. Leonardi, "Multi-agent approach for disaster management," in *Proceedings of the International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (PGCIC '11)*, pp. 311–316, Barcelona, Spain, 2011.

[2] L. M. Camarinha-Matos and H. Afsarmanesh, "Virtual enterprise modeling and support infrastructures: applying multi-agent system approaches," in *Multi-Agent Systems and Applications*, vol. 2086 of *Lecture Notes in Computer Science*, pp. 335–364, 2001.

[3] Y. M. Chen and W. Y. Wu, "Cooperative electronic attack for groups of unmanned air vehicles based on multi-agent simulation and evaluation," *International Journal of Computer Science*, vol. 9, no. 2, pp. 1–6, 2012.

[4] R. Stranders, *Decentralised coordination of information gathering agents [Ph.D. thesis]*, University of Southampton, School of Electronics and Computer Science, 2010.

[5] Y. Xu, M. Lewis, K. Sycara, and P. Scerri, "Information sharing in very large teams," in *Proceedings of the Workshop on Challenges in Coordination of Large Scale MultiAgent Systems (AAMAS '04)*, New York, NY, USA, 2004.

[6] A. Farinelli, P. Scerri, and M. Tambe, "Building large-scale robot systems: distributed role assignment in dynamic, uncertain domains," in *Proceedings of the Workshop on Representations and Approaches for Time-Critical Decentralized Resource, Role and Task Allocation*, 2003.

[7] R. Mailler, V. Lesser, and B. Horling, "Cooperative negotiation for soft real-time distributed resource allocation," in *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS '03)*, pp. 576–583, Melbourne, Australia, July 2003.

[8] M. H. Burstein and D. E. Diller, "A framework for dynamic information flow in mixed-initiative human/agent organizations," *Applied Intelligence*, vol. 20, no. 3, pp. 283–298, 2004.

[9] J. Giampapa and K. Sycara, "Team oriented agent coordination in the RETSINA multi-agent system," in *Applied Intelligence on Agents and Process Management*, 2004.

[10] P. Scerri and K. Sycara, "Social networks for effective teams," in *Cooperative Networks: Control and Optimization*, pp. 295–313, 2008.

[11] K. Y. You and L. H. Xie, "Network topology and communication data rate for consensusability of discrete-time multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2262–2275, 2011.

[12] K. Fall and K. Varadhan, "The Network Simulator-NS-2," 2007, http://www.isi.edu/nsnam/ns.

[13] X. Chang, "Network simulations with OPNET," in *Proceedings of the 31st Winter Simulation Conference (WSC '99)*, pp. 307–314, Phoenix, Ariz, USA, December 1999.

[14] P. Scerri, D. V. Pynadath, N. Schurr, A. Farinelli, S. Gandhe, and M. Tambe, "Team oriented programming and proxy agents: the next generation," in *Proceedings of the 1st International Workshop on Programming Multi-Agent Systems (ProMAS '03)*, pp. 131–148, Melbourne, Australia, 2003.

[15] V. Lesser, K. Decker, T. Wagner et al., "Evolution of the GPGP/TÆMS domain-independent coordination framework," *Autonomous Agents and Multi-Agent Systems*, vol. 9, no. 1-2, pp. 87–143, 2004.

[16] E. H. Durfee and V. R. Lesser, "Partial global planning: a coordination framework for distributed hypothesis formation," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 21, no. 5, pp. 1167–1183, 1991.

[17] P. Scerri, A. Farinelli, S. Okamoto, and M. Tambe, "Allocating roles in extreme teams," in *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS '04)*, pp. 1502–1503, New York, NY, USA, 2004.

[18] M. Tambe, "Towards flexible teamwork," *Journal of Artificial Intelligence Research*, vol. 7, pp. 83–124, 1997.

[19] H. B. Deng, X. Zhang, and J. R. Liu, "Intelligence learning model of multi-agent system," in *Proceedings of the 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC '11)*, pp. 418–421, Deng Leng, China, August 2011.

[20] Y. Xu and P. Scerri, "Token based resource sharing in heterogeneous multi-agent teams," in *Principles of Practice in Multi-Agent Systems*, vol. 5925 of *Lecture Notes in Computer Science*, pp. 113–126, 2009.

[21] H. Jeong, R. Albert, and A.-L. Barabási, "Diameter of the world-wide web," *Nature*, vol. 401, no. 6749, pp. 130–131, 1999.

[22] Y. Xu, X. Li, and H. Liang, "Adjusting organization to improve coordination for large heterogeneous multi-agent teams," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT '12)*, vol. 2, pp. 282–286, Macau, 2012.

[23] 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 (AAMAS '05)*, pp. 543–550, Utrecht, The Netherlands, July 2005.

[24] L. Lovasz, "Random walks on graphs: a survey," *Combinatorics*, vol. 2, no. 1, pp. 1–46, 1993.

[25] P. Erdős and A. Renyi, "On the strength of connectedness of a random graph," *Acta Mathematica Hungarica*, vol. 12, no. 1-2, pp. 261–267, 1960.

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

[27] T. G. Lewis, *Network Science Theory and Applications*, 2009.

[28] S. A. Pandit and R. E. Amritkar, "Characterization and control of small-world networks," *Physical Review E: Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, vol. 60, no. 2, pp. R1119–R1122, 1999.

[29] J. Lü, X. Yu, G. Chen, and D. Cheng, "Characterizing the synchronizability of small-world dynamical networks," *IEEE Transactions on Circuits and Systems—I: Regular Papers*, vol. 51, no. 4, pp. 787–796, 2004.

[30] R. Cohen, K. Erez, D. Ben-Avraham, and S. Havlin, "Resilience of the Internet to random breakdowns," *Physical Review Letters*, vol. 85, no. 21, pp. 4626–4628, 2000.

[31] P. Scerri, E. Liao, Y. Xu et al., "Coordinating very large groups of wide area search munitions," in *Theory and Algorithms for Cooperative Systems*, World Scientific, 2004.

[32] M. Prokopenko, M. Butler, and T. Howard, "On emergence of scalable tactical and strategic behaviour," in *RoboCup-2000: Robot Soccer World Cup IV*, pp. 357–366, Springer, Berlin, Germany, 2001.

[33] M. Atkin, D. Westbrook, and P. Cohen, "Capture the flag: military simulation meets computer game," in *Proceedings of the AAAI Spring Symposium on AI and Computer games*, pp. 1–5, Palo Alto, Calif, USA, 1999.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 407639, 13 pages
http://dx.doi.org/10.1155/2014/407639

Research Article

Discovering the Influences of Complex Network Effects on Recovering Large Scale Multiagent Systems

Yang Xu, Pengfei Liu, Xiang Li, and Wei Ren

School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China

Correspondence should be addressed to Yang Xu; xuyang@uestc.edu.cn

Received 26 December 2013; Accepted 6 March 2014; Published 2 April 2014

Academic Editors: H. R. Karimi, X. Yang, and W. Zhang

Building efficient distributed coordination algorithms is critical for the large scale multiagent system design, and the communication network has been shown as a key factor to influence system performance even under the same coordination protocol. Although many distributed algorithm designs have been proved to be feasible to build their functions in the large scale multiagent systems as claimed, the performances may not be stable if the multiagent networks were organized with different complex network topologies. For example, if the network was recovered from the broken links or disfunction nodes, the network topology might have been shifted. Therefore, their influences on the overall multiagent system performance are unknown. In this paper, we have made an initial effort to find how a standard network recovery policy, MPLS algorithm, may change the network topology of the multiagent system in terms of network congestion. We have established that when the multiagent system is organized as different network topologies according to different complex network attributes, the network shifts in different ways. Those interesting discoveries are helpful to predict how complex network attributes influence on system performance and in turn are useful for new algorithm designs that make a good use of those attributes.

1. Introduction

In state of the art of artificial intelligence research, scalable multiagent system applications in complex environments have been popular in domains of military [1], crisis management [2], and business [3]. In those systems, to efficiently coordinate and share information, agents are required to communicate via flexible wireless media, such as mobile ad hoc networks. In those networks, agents may only be able to directly connect with a few of the others and the network topologies may dynamically change according to agents' movements or joint intentions. Moreover, when the system gets bigger, Musolesi et al. have found that the system presents the characteristics of complex networks [4], for example, small world effect discovered by Travers and Milgram [5] and scale free phenomenon discovered by Barabási and Albert [6]. Predicting team performances according to different communication network topologies is interesting and challenging. From our previous research we learned that the same coordination algorithm may lead to huge different performances when the complex network topologies vary [7]. Other researches support our discovery as well. For example, Scerri and Sycara mathematically analyzed how different complex networks may affect the system in information sharing, sensor fusion, and task allocation [8]. Gaston and DesJardins took a bottom-up research on network formation and found that the incomplete complex network structure varies in decentralized adaptation strategies on team performances [9].

Although the effects of complex networks are popular in large multiagent systems, not all distributed algorithms are tested under different complex network topologies in order to estimate how complex network effects change the system performance. As the network recovery is one of the most important network operations to maintain a desired system performance, in this paper, we made an initial effort on finding how the network topology of a multiagent system shifts when agents recover from their communication failures. For example, a UAV may be shot by a hostile missile or robots' connections broken by physical obstacles in a mobile ad

hoc network. Although the popular restoration of rerouting mechanism, for example, MPLS recovery algorithm, has been proven to be feasible on network recoveries, the potential shifts on the multiagent network topology may significantly change the system performance in an unpredictable way.

In our simulation, we simulated a series of large scale multiagent systems with different complex network topologies. A popular restoration of rerouting mechanism-MPLS algorithm is implemented to recover link and node failures. The network is evaluated by its diameter, average distance, and cluster [10]. To simulate the physical network, in those experiments, the communication capabilities of each link or node are limited. The experiment results are presented in two major sections: system robustness and influence of network topologies.

In the first section of the experiments, data that flows through the failure agents or links has to be rerouted and may cause congestions on existing links or agents. Based on our discussion about the efficiency of network recovery, the number of newly congested links or nodes (agents) that hurt system performance is investigated. More importantly, the network is in danger of being disconnected from those congestions. By comparing the influence on different network topologies, while there are some differences in different cases, our major discoveries are that a random network is more likely to be congested if more links are broken, while a small world network is the least likely. On the other hand, a scale free network appears to be more vulnerable to node congestions, while a grid network is the most robust.

In the second section, we test how the network topology may be shifted from the network recovery. We have found that most of network topologies are immune to the network recovery when the congestions are not so serious. However, when the number of congested links and nodes is highly increased, the network topologies may have been changed. It is especially the case when the network is organized as a scale free network or a small world network. However, since scale free and small world networks are the most important attributes in a large multiagent system, from our previous research experience, their changes may significantly affect the system performances.

2. Related Work

The structure and behavior of complex networks have been attractive in various studies [11]. Inspired by the discoveries on how the rich network structure facilitates effective organizational behavior [12], Gaston and DesJardins illustrated the importance of network topologies in multiagents networks [9]. Y. C. Jiang and J. C. Jiang analyzed the complex network in actor-oriented and actor-structure views to find the relationship between complex networks and multiagent systems [13]. Taylor et al. examined joint actions in the multiagent optimization problem, and the results are surprising because a high number of connections in a complex networked multiagent team can hurt system performance; even communication and computation costs are ignored [14]. Liu et al. proposed an integrated model based on small world network and multiagent system to simulate epidemic spatiotemporal transmission. They found that the small world effect brings better performance than the traditional model, and his discovery has been applied in real geographical multiagent applications [15].

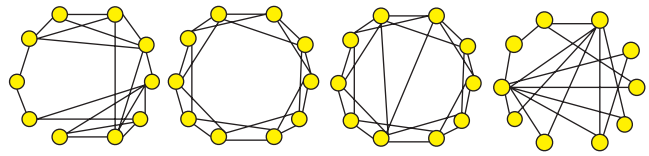


Figure 1: Different complex network topologies: random network, grid network, small world network, and scale free network.

On the other hand, a set of researches show that network operations could influence the performance of the large scale multiagent systems as well. For example, when facing the same network attack, the network vulnerabilities are different if the multiagent systems are organized as various types of complex networks [16]. D'Angelo and Ferretti explained that gossip protocols would change the communication of the complex network and have some impacts on routing efficiency [17]. Gong and Xu analyzed and tested that different parameters on a scale free network can make significant different efficiencies of information delay in multiagent systems [18]. Peschlow et al. [19] described a flexible dynamic partitioning algorithm that rapidly recovers information routing and optimizes the performance with different complex network topologies.

3. Modeling the Complex Networks

The network topology of a multiagent system is defined as an undirected graph $G = (V, E)$, where $V = \{1, 2, \ldots, N\}$, $N = |V|$, defines the agent set. E denotes the set of links between agents; that is, link $(i, j) \in E$ and i and j are neighbors. $n : V \rightarrow E$ defines the set of all neighbors of an agent. That is, $n(i) = \{b, j, \ldots, l\}$ is the neighbor of agent i. G could be organized as different network topologies based on the different properties of complex networks. In this paper, we are mainly interested in four of them shown in Figure 1: random network, grid network, small world network, and scale free network. Preliminary studies [20] have found that each topology encodes different fundamental properties, that is, network diameter, average distance between nodes, cluster, and degree distributions.

(i) Degree: the degree of agent 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: $p(k) = Pr[d = k]$ defined as a fraction of agents (the number of such agents is d) with the degree k.

(iv) Distance: distance(i, j) is defined as the least number of hops to communicate between the agents i and j. Specifically, distance$(i, j) = 1$, if $(i, j) \in E$.

(v) Average distance:

$$l = \frac{1}{N(N+1)} \sum_{\forall i,j \in V} \text{distance}(i,j) \quad (1)$$

is the average distance between any pairs of agents.

(vi) Network diameter: the diameter of the graph G is defined as argmax(D), where $D = \{\text{distance}(i,j) \mid i,j \in V\}$, which is the longest distance between pairs of agents.

(vii) Clustering coefficient: C_i for an agent i is given by the proportion of links between the agents within the set of $n(i)$ that is divided by the number of links between them. The set $e(i)$ is to record the existing links between these agents in $n(i)$; then

$$C_i = \frac{2|e(i)|}{|n(i)| * (|n(i)| - 1)}. \quad (2)$$

The clustering coefficient for the network is given by $C = (1/N)\sum_{i=1}^{N} C_i$.

Different complex network topologies can be described according to the properties mentioned above. Erdös and Rényi put forward a classical random network ER model [21]. In this model, a random network follows a Poisson degree distribution. Most nodes in a grid network keep the same degree, which is also called a regular network. Watts and Strogatz put forward the concept of small world network and the WS model [22]. This model presents much shorter average distance than that in a grid network. Moreover, some typical large scale networks such as mobile agents on internet [23] and hyperlinks on web [24] possess certain dynamics—*Matthew effect* [25], a power law distribution: $p(k) \propto k^{-r}$ $(2 < r < 3)$. Some researchers found an interesting formula: argmax(D) $\propto$ lnln$N$ [26], that the average distance may decrease as the network grows [27]. This formula precisely reflects the small world effect as well. Then, Barabási and Albert put forward a scale free network and the BA model [6]. Our simulations are based on those complex network models.

To simulate a physical network, we define the flow of the communication for a link $(i,j) \in E$ as $f(i,j)$ and its allowed bandwidth is set as a constant, written as $F_{\max}$. Therefore, $f(i,j)$ should not overflow to its bandwidth; otherwise, the link will be congested. Please note, if $f(i,j) = 0$, there is no communication in a physically connected link (i,j) and it is called a backup link that may be used for future communications. In addition, we define $r(i)$ as the amount of communication through agent i. $r(i)$ cannot be more than $C_{\max}(i)$, the max allowed bandwidth capability through agent i; otherwise the agent is congested as well. Moreover, the following properties are defined in our simulations.

(i) Network connection: G is disconnected if $\exists i,j \in V$, distance$(i,j) = \infty$, or no value can be assigned. Otherwise, we say the network G is connected.

(ii) Subgraph of a pair of agents $\langle i,j \rangle$: let $G'(i,j)$ be the subgraph of G, $G'(i,j) = (V', E')$, where V' is the set of agents on all the shortest paths between the pairs of agents $\langle i,j \rangle$, and $E' \subset E$ consists of all the links in those shortest paths.

(iii) Transition agents of agent i to j: let $s(i)$ be the set to record all the neighbors that can transfer data from agent i to j, and $s(i) \subset (n(i) \cap G')$.

(iv) Transition paths of agent i: let path(i) $= \{\ldots, \langle s,t \rangle, \ldots \mid \ldots, s,t,\ldots \in n(i)\}$ be the set that records all the pairs of agents that communicate data via agent i.

(1) find all transition agents $S(i)$;
(2) **for** each agent $w \in s(i)$ **do**
(3) $\quad SendData\left(w, \frac{f(i,j)}{|s(i)|}\right)$;
(4) **end for**

Algorithm 1: *Link_Recovery*$((i,j), f(i,j))$.

(1) $path(i) \leftarrow EnumerateStream(i)$;
(2) **for** each path $p \in path(i)$ **do**
(3) $\quad (u,v) \leftarrow FindNeigbors(p,i)$;
(4) $\quad Link\_Recovery((u,v), f(p))$;
(5) **end for**

Algorithm 2: *Agent_Recovery*(i).

4. MPLS-Based Recovery Mechanisms

When agents come to link failures or node failures, the restoration of rerouting mechanisms is usually exploited to maintain the communication between agents. In this paper, we implement a typical network recovery policy, MPLS [28], to restore communication between agents by rerouting mechanisms.

Algorithm 1 briefly describes how a failed link (i,j) whose data flow is $f(i,j)$ is recovered. We suppose there is a predefined sequence according to agent' ID that $i \prec j$. Assume that each agent is able to get the global state of the network, agent i can easily find all the alternative shortest paths to j, and w is one of the transition agents $s(i)$ (line 1). In line 3, the data flow will be divided evenly into pieces according to the number of shortest paths detected. Each of them will be sent to one of the transition agents and passed through predefined paths (line 3).

Algorithm 2 briefly shows how a failed agent i is recovered. The communication through an agent i may be composed of several streams from different links. Each stream p going through the agent i is written as a unique path $\{\ldots, u, i, v, \ldots\}$, where $u \prec v$. The value of data flow going through p is written as $f(p)$. Therefore, to recover node failure, Algorithm 2 first enumerates all the stream path(i) (line 1). For each stream p, we will find the pair of neighbors of i : $\langle u,v \rangle$ in p's path (line 3). Then, if we suppose there has been a link of (u,v) whose communication amount is $f(p)$,

```
(1) μ ← w;
(2) for each agent i ∈ s(i) do
(3)   if agent i was not been visited then
(4)     x ← F_max − (μ + f(i, j));
(5)     if x < 0 then
(6)       ε ← F_max − f(i, j);
(7)       return the link (i, j) is congested;
(8)     end if
(9)     if μ ≠ 0 then
(10)      t ← C_max − (ε + r(j));
(11)    end if
(12)    if t ≤ 0 then
(13)      return the agent j is congested;
(14)    else
(15)      if t ≤ ε then
(16)        μ ← t;
(17)        return the agent j is congested;
(18)      else
(19)        μ ← ε;
(20)      end if
(21)    end if
(22)  end if
(23) end for
```

Algorithm 3: *Congestion*(i, j).

the stream p can be rerouted in a way similar to link failure algorithm (Algorithm 1) (line 4).

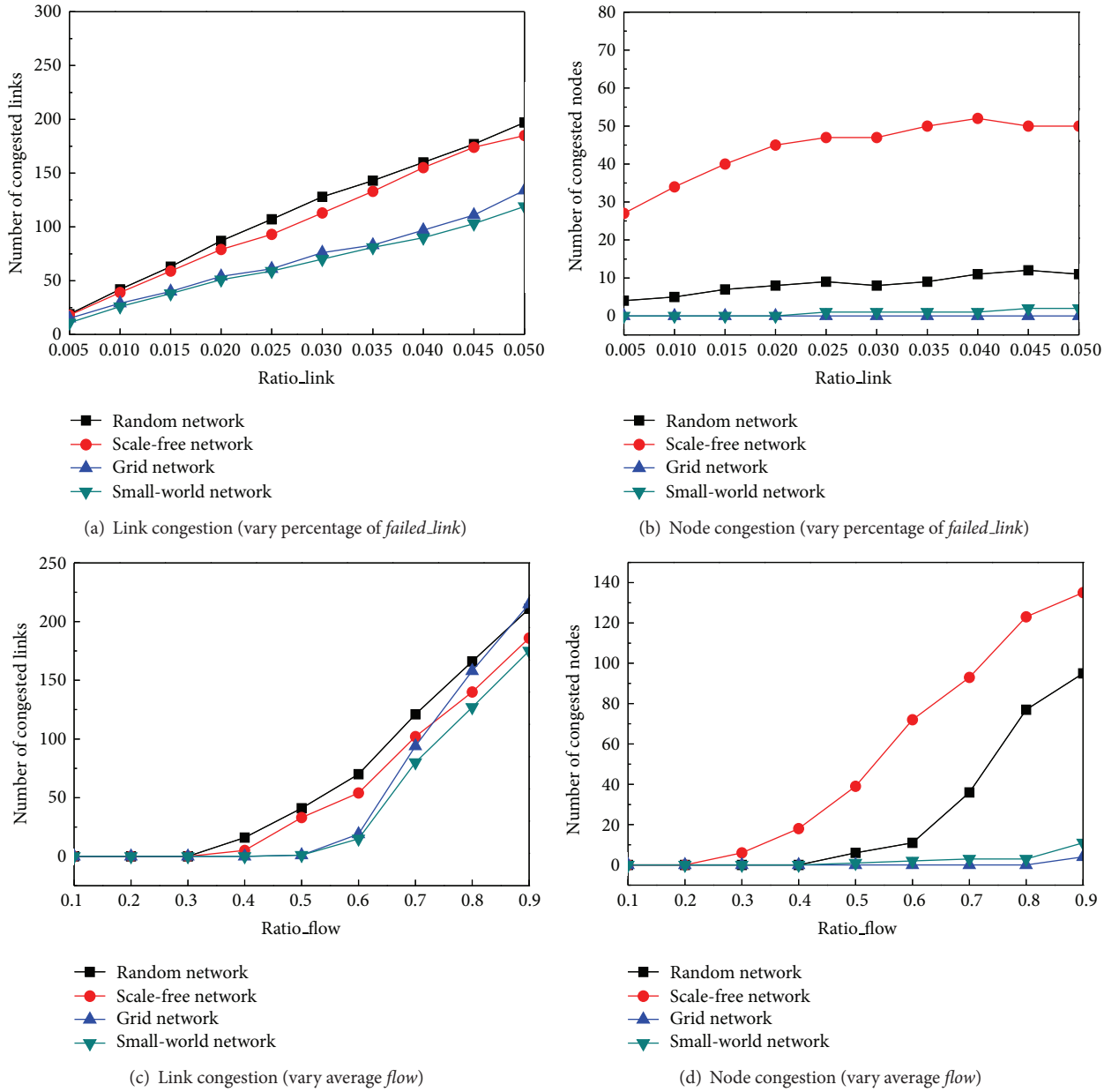
5. Network Robustness after MPLS Recovery

Although MPLS recovery mechanism can effectively enhance recovery efficiency, network congestion cannot be avoided due to the limited physical communication capacities of the links and agents. In order to detect network congestion including link congestions and node congestions, we designed Algorithm 3 to check existed network congestions around the multiagent system by rerouting data from a link failure (i, j) or a node failure j.

According to Algorithm 3, we can first summarize the link congestion by link (i, j). It is supposed that the amount of messages conveyed to the link (i, j) is set as μ (line 1). The available capacity x for an existing link is calculated as $x = F_{\max} - (\mu + f(i, j))$ (line 4). If there is no available capacity, link (i, j) is congested (lines 5–7). To judge the agents' congestion, we calculate the available capacity t for an existing agent j as $\varepsilon \leftarrow F_{\max} - f(i, j)$ and $t \leftarrow C_{\max} - (\varepsilon + r(i))$ (lines 6–10). If there is no available capacity, the agent j is congested (lines 12–21). The rerouting mechanisms modify the LSP in a failed spot and the length of the shortest path is often more than one agent; therefore, the amount of messages μ may be modified (lines 16–19), and the algorithm may be recursively judged many times in terms of Depth First Search (DFS).

In this section, we investigate how network recovering operation may create node or link congestions when G is organized as four different network topologies. The system size is $N = 1000$, average degree is $\overline{d} = 6$, and maximum allowed link overload is $F_{\max} = 10$. The agent having a higher degree usually plays an important role in the network; therefore the communication through the agent is larger. Based on this observation, the capacity of agent i follows $C_{\max}(i) = \lambda \times d(i) \times F_{\max}$, where $0 < \lambda < 1$ is a constant so that its capability is proportional to the degree of the agent. The results are evaluated according to newly congested links, congested agents, probability of network connectivity, and the average distance of the network by varying the ratio of failed links (*Ratio_link*), the ratio of failed nodes (*Ratio_agent*), and the ratio of average communication of each link $f(i, j)$ (*Ratio_flow*). Moreover, when $m = 0$, we assume there are no backup links and all the links are used for communication and, for each (i, j), $f(i, j) > 0$. If $m > 0$, additional 15% backup links are presented. The amount of communication for each link is randomly set. The experiment results are based on 100 runs.

5.1. Robustness on Link Failure Recovery. In this section, we examine the number of congested links and congested nodes when the MPLS algorithm recovered link failures. We fix the average communication of each link $f(i, j)$ as 50% of the $F_{\max}$ (*Ratio_flow* = 0.5). In Figures 2(a) and 2(b), we varied the number of link failures from 0.5% to 5% of all the links. In Figure 2(a), network recovery operation leads to congested links and congested nodes, and their number rapidly increases with the slow increase of *Ratio_link*. By comparing the number of congested links in the same *Ratio_link*, we have found that random network appears to have the largest number of congested links while the small world network appears to have the least number of congested links.



(a) Link congestion (vary percentage of *failed_link*)

(b) Node congestion (vary percentage of *failed_link*)

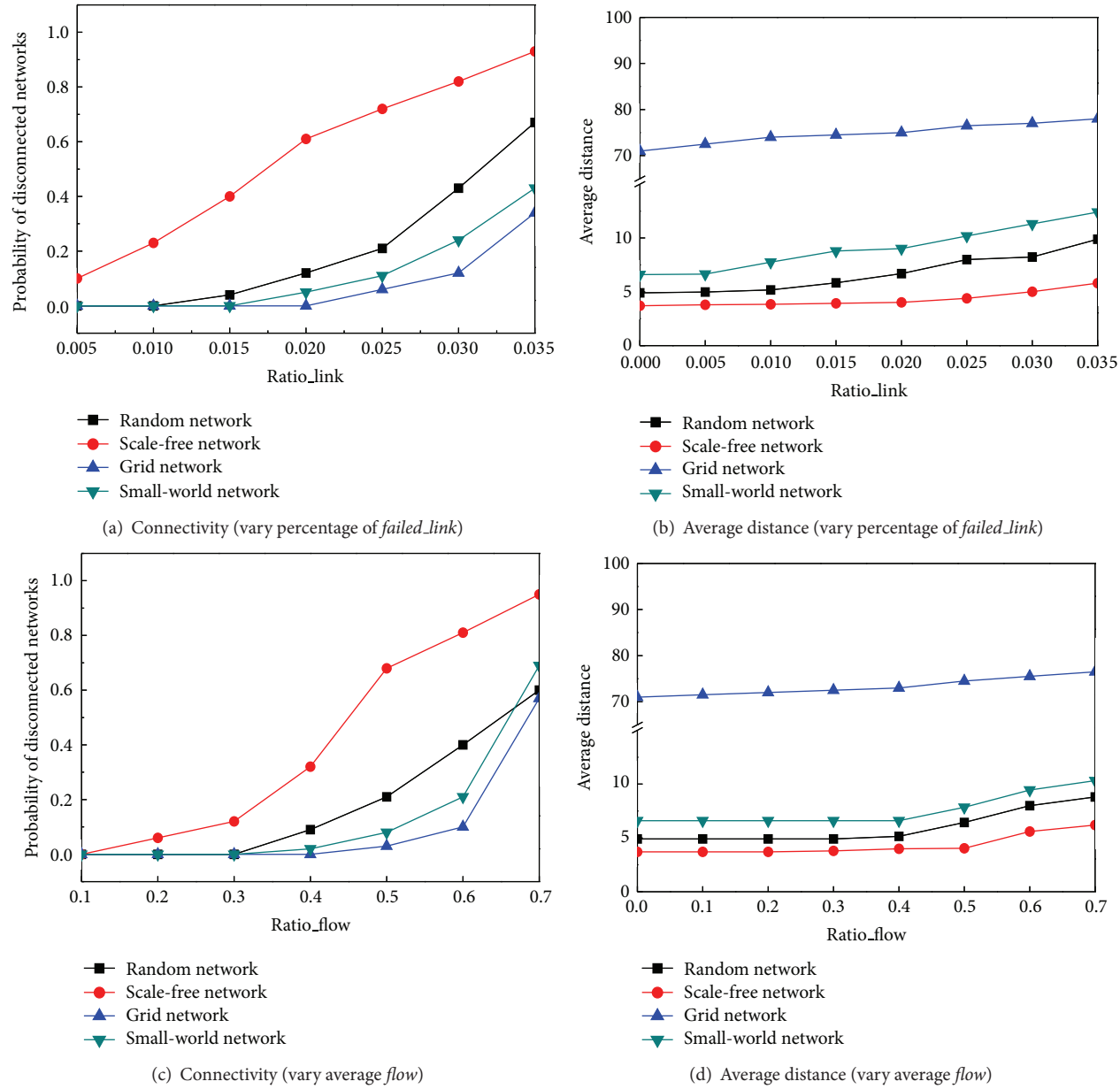
(c) Link congestion (vary average *flow*)

(d) Node congestion (vary average *flow*)

FIGURE 2: Congestions from link failure recovery by varying *Ratio_link* and *Ratio_flow*.

The experimental results in Figure 2(b) show that, except for the grid network, network recovery operation on failed links will lead to node congestions. Apparently, scale free network appears to have the largest number of congested agents in any settings, because of the stability of hub nodes which have large bandwidth (proportion to its degree). The other agents with limited bandwidth are prone to be jammed. Moreover, cluster may contribute to the node congestions as well. For example, the agent i whose degree is larger than others' in a scale free network has a larger clustering coefficient C_i, and the number of agents retransmitting data would be high. In addition, the cluster of a grid network is the smallest in the four networks, and the node congestions are less likely to be present.

In the next experiment, we fix the failed link as 2% (*Ratio_link* = 0.02). Figures 2(c) and 2(d) show that when the average communication *Ratio_flow* of each link $f(i, j)$ increases from 10% to 90% of the $F_{\max}$, both the congested nodes and links increase no matter what the complex network topologies are. Consistent with the conclusion from Figures 2(a) and 2(b), a random network performs the worst while a small world network performs the best according to congested links. A scale free network appears to have the largest number of congested nodes and random network



(a) Connectivity (vary percentage of *failed_link*)

(b) Average distance (vary percentage of *failed_link*)

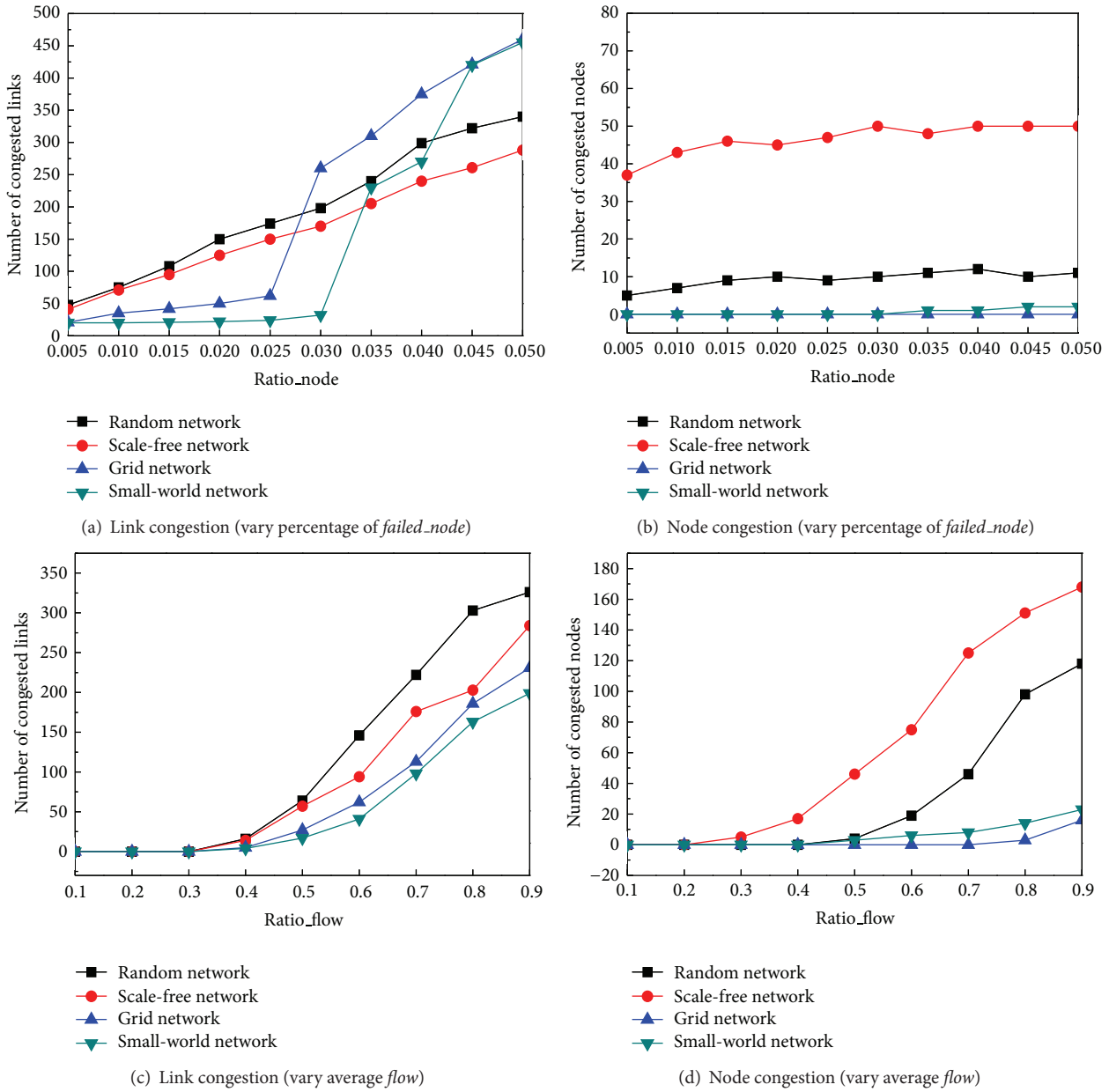
(c) Connectivity (vary average *flow*)

(d) Average distance (vary average *flow*)

FIGURE 3: Connectivity and average distance by varying *Ratio_link* and *Ratio_flow*.

performs worse as well. The grid network appears to have the least number of congested nodes. It can be explained the same as Figure 2(b).

Figure 4 summarizes how link failure recovery may influence the network performance in two sets of values: the probabilities that the network is broken and the average distance of the network. If network breaks, the system may not work. If the average distance increases, the system's performance decreases because communication flows have to take more hops to the destination. In Figures 3(a) and 3(b), the result shows that when there are more failed links to be fixed ($Ratio\_flow = 0.5$), the system is in a higher danger to be disconnected. However, a scale free network performs the worst and a grid network performs the best. The reason is that if a hub agent is congested, the network is more likely to be disconnected. On the other hand, agents in a grid network only connect to each other locally; the network is less likely to break down even when more and more links or agents are congested. Figures 3(a) and 3(b) also show that the average distance slowly increases in all network topologies, and, before the network is broken down, the scale free network keeps the shortest distance and reserves the small world effect best. Figures 3(c) and 3(d) illustrate that when the average flow increases ($Ratio\_link = 0.02$),



(a) Link congestion (vary percentage of *failed_node*)

(b) Node congestion (vary percentage of *failed_node*)

(c) Link congestion (vary average *flow*)

(d) Node congestion (vary average *flow*)

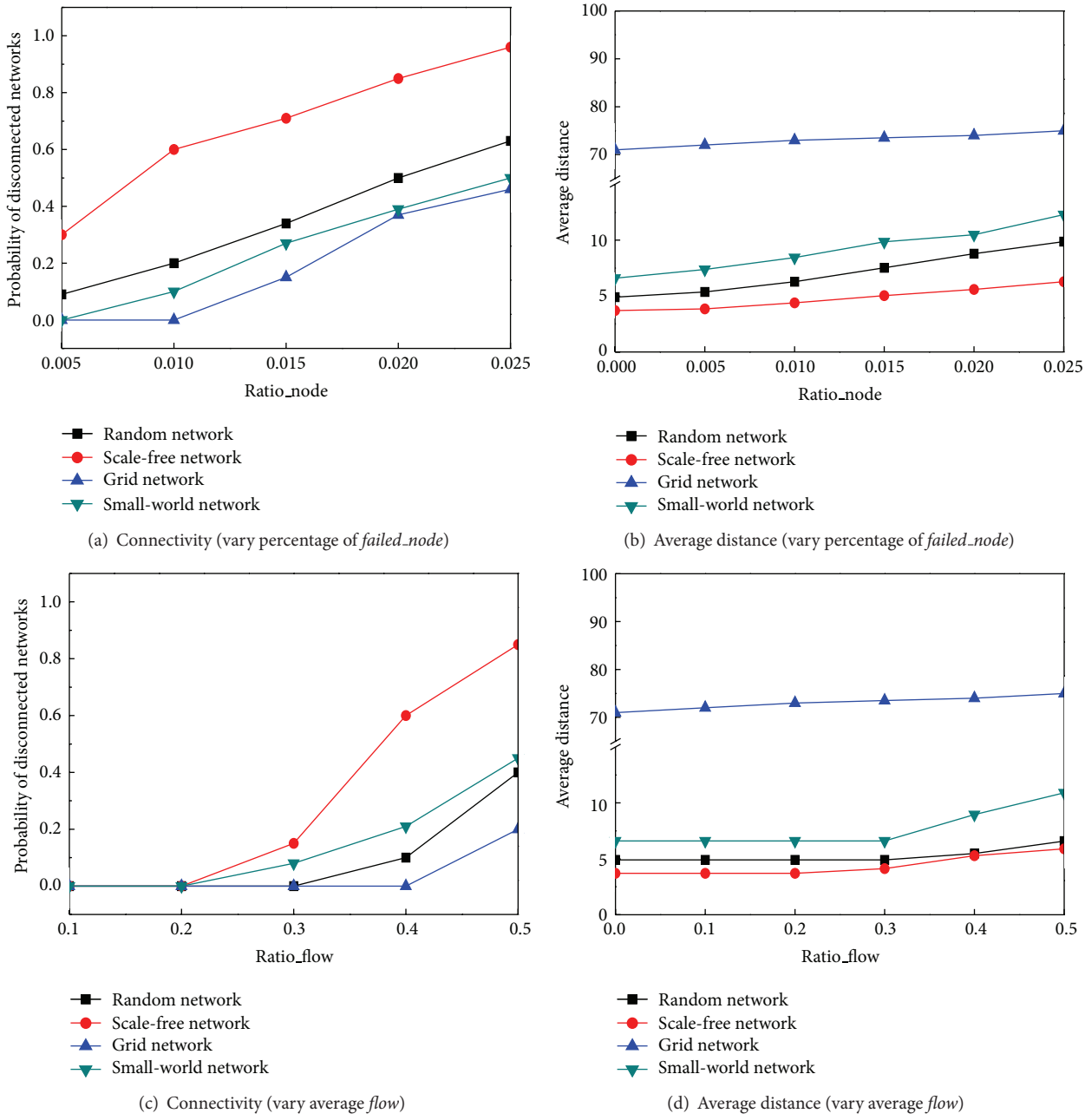
FIGURE 4: Congestions from node failure recovery by varying *Ratio_node* and *Ratio_flow*.

a similar conclusion can be reached as in Figures 3(a) and 3(b).

5.2. Robustness on Node Failure Recovery. In this section, we investigate the network performances after the network recovery policy recovered node failures. In Figures 2(b) and 4(b), we varied the failed agents in the network from 0.5% to 5%, and we set $Ratio\_flow = 0.5$. We found that the congested links increased quickly while congested agents slowly increased. As we expected, scale free networks made heavy congested nodes. Unlike Figures 2(a) and 4(a), although random and scale free networks have more number of congested links when the failed nodes are sparse, small world and grid networks create about 40% more congested links when failed nodes are more than 3.5% of all the agents.

Figures 4(c) and 4(d) show the results that when we set the failed agents to be fixed as 2% ($Ratio\_node = 0.02$) and varied the average flow of each link from 10% to 90% of the F_{max}, both congested agents and congested links are increasing in different complex network topologies. Consistent with the results of link failure recovery, a random network performs the worst according to congested links while a small world network works the best. On the other hand, as we expected, a scale free network always creates more congested nodes while a grid network creates the least.

Figure 5 shows when either the rate of failed nodes increases or the average flow rate increases, the probability



(a) Connectivity (vary percentage of *failed_node*)

(b) Average distance (vary percentage of *failed_node*)

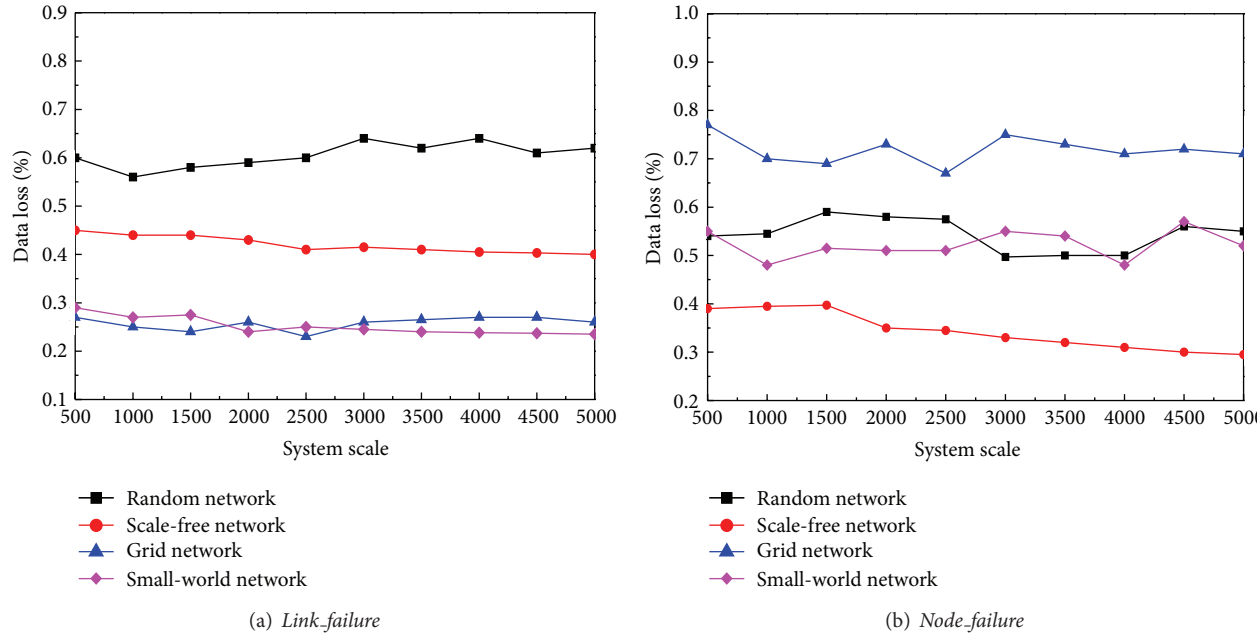
(c) Connectivity (vary average *flow*)

(d) Average distance (vary average *flow*)

FIGURE 5: Connectivity and average distance by varying *Ratio_node* and *Ratio_flow*.

that the network is broken increases. In Figures 5(a) and 5(b), when there are more failed nodes to be recovered (*Ratio_flow* = 0.5), the system is in a higher danger to be disconnected; however, a scale free network performs the worst and a grid network performs the best. The reason is that if any hub agents are congested, the network is much easier to be disconnected. Figures 5(a) and 5(b) also show that the average distance slowly increases in all network topologies, and the scale free network still keeps the shortest average distance as we expected. Figures 5(c) and 5(d) represent that when the average flow increases (*Ratio_node* = 0.02), we can make the similar conclusions as Figures 5(a) and 5(b). All the experiments in this section are based on the setting of $m > 0$ (there are 15% backup links), but we could reach the same conclusion when we set $m = 0$.

5.3. Data Loss in Network Recovery. As explained, when the multiagent system comes to network failures, although MPLS helps to reroute the data to maintain the system performance, network congestions in the nodes or the links between them will still bring communication loss. In this subsection, we investigate the percentages of data loss when the multiagent system is organized as different complex networks.



(a) *Link_failure* (b) *Node_failure*

FIGURE 6: Data loss from network recovery in different network topologies.

In the first experiment, we briefly use the basic setting of Section 5.1 that, in the multiagent system, there are 2% links that are broken and the original average data volume in each link is 50% of F_{max}. When the system was organized as four different complex network topologies, we measured the percentages of the data loss in different scales. The results are illustrated as in Figure 6(a). We can see that no matter the system size is, the grid network and small world network maintain good performances in link failure recoveries and the data loss rates keep the lowest. Consistent with our analysis in Section 5.1, the scale free network keeps a higher data loss rate and random network performs the worst in link failure recovery where its data loss rate closes to 60%.

In the second experiment, we briefly use the basic setting of Section 5.2, and there are 2% agents that are lost. When the system was organized as four different complex network topologies, Figure 6(b) briefly shows that no matter the system size is, scale free network occupies the stability of hub nodes and always performs the best, and it is especially the case when the network scales up. Consistent with our conclusion in Section 5.2, small world network and random network bring out close performances, while grid network made the worst data loss in node failure recovery. In some cases, the data loss rates are more than 70%.

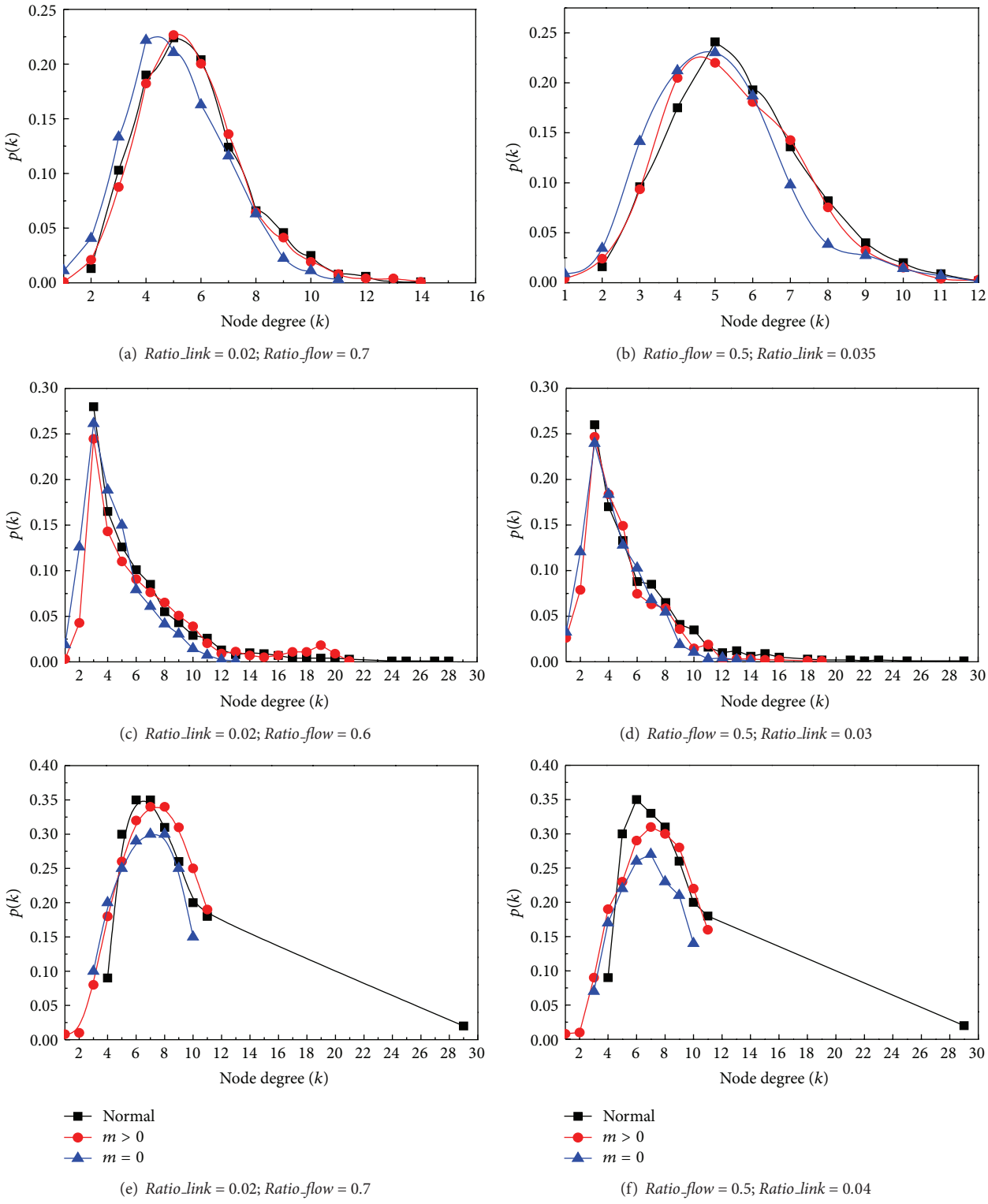
6. Network Shifts on Recovery from Different Topologies

In this section, we verify if network recovery operation would lead to the changes of complex network topologies. Similar experiment settings are kept as Section 5, and both $m > 0$ (consists of 15% backup links) and $m = 0$ are tested. During the experiments, we found that network recovery usually does not lead to distinct shifts of network topologies. However, when the number of congested links and nodes rapidly increases, network connectivity may be destroyed. We set $C_{max}(i) = \lambda \times F_{max}$, where $\lambda > 1$ is a constant so that agents' communication volumes are fixed. Our experiments are conducted by varying the parameters *Ratio_link*, *Ratio_node*, and *Ratio_flow* but we always maintain that the network connectivity is not broken (very few results with disconnected networks are excluded). The experiment's results are shown as degree distribution. Each graph represents one type of complex network and consists of three curves with three settings: the original network topology before any failures (*Normal*), the network topology after network recovery ($m > 0$), and the network topology after network recovery without any backup links ($m = 0$).

6.1. Link Failure Recovery. In this experiment, we tested how the different network topologies will be changed after link failure recovery. Each network topology will be presented in two different settings with different rate of link failure to be recovered (*Ratio_link*) and different average flow (*Ratio_flow*) which are very likely to break network connectivity.

Figures 7(a) and 7(b) show how a random network topology shifts on two different settings. Although the random network topology is kept and its distribution still follows a Poisson distribution, the distribution clearly shifts left after the link failure recovered (it is more distinct when $m = 0$). Therefore, its average degree is decreased with link failure recovery.

Figures 7(c) and 7(d) show that the scale free network significantly shifted. In both graphs, the scale free networks are losing their power law distribution and are more and more close to a Poisson distribution as random networks. In the



(a) *Ratio_link* = 0.02; *Ratio_flow* = 0.7

(b) *Ratio_flow* = 0.5; *Ratio_link* = 0.035

(c) *Ratio_link* = 0.02; *Ratio_flow* = 0.6

(d) *Ratio_flow* = 0.5; *Ratio_link* = 0.03

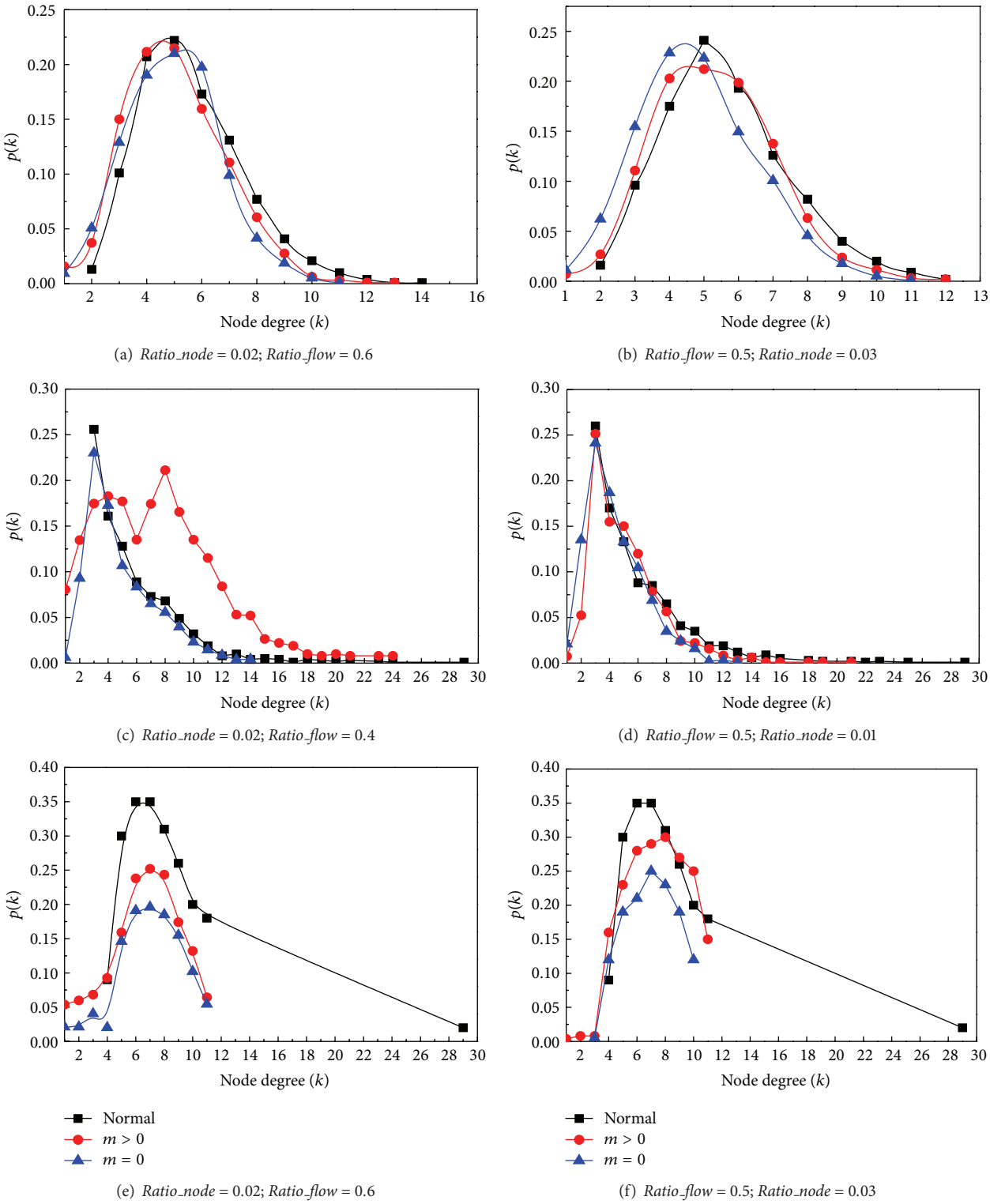
(e) *Ratio_link* = 0.02; *Ratio_flow* = 0.7

(f) *Ratio_flow* = 0.5; *Ratio_link* = 0.04

FIGURE 7: Network shifts from link failure recovery in different network topologies.

settings of $m = 0$, almost all the high degree agents are lost. The reason is that hub agents are easily congested when much more communication flow transmitted through hub agents. Therefore, the *small world effect* is gradually disappeared.

In Figures 7(e) and 7(f), the original small world network before link recovery presents a generalized binomial distribution [29]. However, the network cannot keep this topology in both graphs. All the agents with higher degrees in a small



(a) *Ratio_node* = 0.02; *Ratio_flow* = 0.6

(b) *Ratio_flow* = 0.5; *Ratio_node* = 0.03

(c) *Ratio_node* = 0.02; *Ratio_flow* = 0.4

(d) *Ratio_flow* = 0.5; *Ratio_node* = 0.01

(e) *Ratio_node* = 0.02; *Ratio_flow* = 0.6

(f) *Ratio_flow* = 0.5; *Ratio_node* = 0.03

FIGURE 8: Network shifts from node failure recovery in different network topologies.

world network are more likely to be congested, especially in the settings of $m \geq 0$. Moreover, when the average degree of the networks decreases, the average distance between nodes rapidly increases, and its degree distribution closes to a Poisson distribution after link failure recovery.

6.2. Node Failure Recovery. Similar to the experiments of link failure recovery in Section 6.1, we vary two parameters of the node failure recoveries: *Ratio_node* and *Ratio_flow*. Similar to link failure recovery schema, networks are very prone to be broken. Figures 8(a) and 8(b) show the changes

on the random network. Although the degree distributions slightly change and the average degree decreases, its Poisson distribution remains.

Similar to the conclusion from Figures 7(c) and 7(d), Figures 8(c) and 8(d) show that the scale free networks cannot keep their topologies in both settings and are more and more close to a random network. However, in Figure 8(c), the scale free network has a significant mutation with the setting of $m > 0$. Its degree distribution has been a combination of a Poisson distribution and a power law distribution. We found that the agent i whose degree is higher would present higher clustering coefficient C_i. Otherwise, low degree agents are loosely connected with each other [30]. Therefore, the scale free network works distinctly between high clustering coefficient region and low clustering coefficient region after node failures are recovered.

Figures 8(e) and 8(f) illustrate that small world networks cannot keep their topologies in node failure recovery. It is similar to the phenomenon in Figures 7(e) and 7(f). In both settings, degree distribution does not follow a generalized binomial distribution any more. Therefore, the average distance between agents may be significantly increased.

In conclusion, the power law distribution in scale free network and the binomial distribution in small network are unstable and can be easily destroyed by network congestions. On the other hand, a random network with Poisson distribution is stable. Moreover, network recovery operation by creating link or node congestions can significantly decrease the average degree of the network. In this section, we excluded the results from the grid network because it only has local connections and the congestions have little influences on its network topology.

7. Conclusions and Future Work

Network recovery plays an important role in maintaining the stability of the multiagent system in any application domain. However, its impacts on different complex network organizations are still undiscovered. In this paper, we made our initial efforts on studying those effects. We have found that although the MPLS recovery mechanism can efficiently recover link or node failures by rerouting communication flows via alternative paths, it may bring node or link congestions on those alternative paths. The congestions can significantly change the network topology and system performance. In addition, their effects are different on different complex networks. By conducting extensive experiments, we found that the small world effect and power law phenomenon in a scale free network are not stable in many cases. Based on our interesting discoveries, we may be able to make lots of progresses in near future. The first is to predict the system performance variances according to the changes of the network topology in multiagent coordination domains such as resource allocation, information sharing, and task assignments. Second, we could optimize the recovery algorithm efficiency based on the utilizations of complex network attributes.

Conflict of Interests

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

References

[1] C. Wang, B. Wang, G. Zhang, and Y. Liang, "Method of formation cooperative air defense decision based on multi-agent system cooperation," in *Communications and Information Processing*, vol. 289 of *Communications in Computer and Information Science*, pp. 529–538, 2012.

[2] I. G. Magarinoa and C. Gutierrezb, "Agent-oriented modeling and development of a system for crisis management," *Expert Systems with Applications*, vol. 40, no. 16, pp. 6580–6592, 2013.

[3] S. Cranefield and S. Ranathunga, "Embedding agents in business applications using enterprise integration patterns," in *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS '13)*, pp. 1223–1224, 2013.

[4] M. Musolesi, S. Hailes, and C. Mascolo, "An Ad Hoc mobility model founded on social network theory," in *Proceedings of the 7th ACM Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (ACM MSWiM '04)*, pp. 20–24, October 2004.

[5] J. Travers and S. Milgram, "An experimental study of the small world problem," *Sociometry*, vol. 32, no. 4, pp. 425–443, 1969.

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

[7] Y. Xu, M. Lewis, K. Sycara, and P. Scerri, "Information sharing in very large teams," in *Proceedings of the 3rd International Joint Conference on Autonomous Agent and Multi Agent Systems*, 2004.

[8] P. Scerri and K. Sycara, "Social networks for effective teams," in *Cooperative Networks: Control and Optimization*, Edward Elgar Publishing, 2008.

[9] 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.

[10] R. Albert, H. Jeong, and A.-L. Barabási, "Error and attack tolerance of complex networks," *Nature*, vol. 406, no. 6794, pp. 378–382, 2000.

[11] G. A. Pagani and M. Aiello, "The power grid as a complex network: a survey," *Physica A: Statistical Mechanics and Its Applications*, vol. 392, no. 11, pp. 2688–2700, 2013.

[12] Z. Lin and K. Carley, "DYCORP: a computational framework for examining organizational performance under dynamic conditions," *Journal of Mathematical Sociology*, vol. 20, no. 2-3, pp. 193–217, 1995.

[13] Y. C. Jiang and J. C. Jiang, "Understanding social networks from a multi-agent coordination perspective," *IEEE Transactions on Parallel and Distributed Systems*, 2013.

[14] M. E. Taylor, M. Jain, Y. n Jin, M. Yooko, and M. Tambe, "When should there be a, "me" in, "team"? Distributed multi-agent optimization under uncertainty," in *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS '10)*, pp. 109–116, 2010.

[15] T. Liu, X. Li, and X. P. Liu, "Integration of small world networks with multi-agent systems for simulating epidemic spatiotemporal transmission," *Chinese Science Bulletin*, vol. 55, no. 13, pp. 1285–1293, 2010.

[16] L. Gu, X. D. Zhang, and Q. Zhou, "Consensus and synchronization problems on small-world networks," *Journal of Mathematical Physics*, vol. 51, no. 8, Article ID 082701, 2011.

[17] G. D'Angelo and S. Ferretti, "Simulation of scale-free networks," in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, 2009.

[18] X. G. Gong and J. Xu, "Research on delay characteristics of information in scale-free networks based on multi-agent simulation," *Procedia Computer Science*, vol. 17, pp. 989–1002, 2013.

[19] P. Peschlow, T. Honecker, and P. Martini, "A flexible dynamic partitioning algorithm for optimistic distributed simulation," in *Proceedings of the IEEE 21st International Workshop on Principles of Advanced and Distributed Simulation (PADS '07)*, pp. 219–228, San Diego, Calif, USA, June 2007.

[20] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Review of Modern Physics*, vol. 74, pp. 47–97, 2002.

[21] 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.

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

[23] L. Bononi, M. Bracuto, G. D'Angelo, and L. Donatiello, "Exploring the effects of hyper-threading on parallel simulation," in *Proceedings of the 10th IEEE International Symposium on Distributed Simulation and Real-Time Applications (DS-RT '06)*, pp. 257–260, Torremolinos, Spain, October 2006.

[24] A. Broder, R. Kumar, F. Maghoul et al., "Graph structure in the web," *Computer Networks*, vol. 33, no. 1, pp. 309–320, 2000.

[25] R. K. Merton, "The Matthew effect in science," *Science*, vol. 159, no. 3810, pp. 56–63, 1968.

[26] R. Cohen, S. Havlin, and D. Ben-Avraham, "Structural properties of scale-free networks," in *Handbook of Graphs and Networks*, 2003.

[27] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graphs over time: densification laws, shrinking diameters and possible explanations," in *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 05)*, pp. 177–187, August 2005.

[28] V. Sharma and F. Hellstrand, "Framework for multi-protocol label switching (MPLS)-based recovery," RFC, 3469, 2003.

[29] K. Mahdi, H. Farahat, and M. Safar, "Temporal evolution of social networks in Paltalk," in *Proceedings of the 10th International Conference on Information Integration and Web-Based Applications and Services (iiWAS '08)*, pp. 98–103, November 2008.

[30] X. Yao, C. S. Zhang, J. W. Chen, and Y. D. Li, "On the formation of degree and cluster-degree correlations in scale-free networks," *Physica A: Statistical Mechanics and its Applications*, vol. 353, no. 1–4, pp. 661–673, 2005.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 754749, 15 pages
http://dx.doi.org/10.1155/2014/754749

Research Article

A Novel Joint Problem of Routing, Scheduling, and Variable-Width Channel Allocation in WMNs

Chun-Cheng Lin,[1] Wan-Yu Liu,[2] Chun-Hung Chou,[1] and Der-Jiunn Deng[3]

[1] *Department of Industrial Engineering and Management, National Chiao Tung University, Hsinchu 300, Taiwan*
[2] *Department of Tourism Information, Aletheia University, New Taipei City 251, Taiwan*
[3] *Department of Computer Science and Information Engineering, National Changhua University of Education, Changhua 500, Taiwan*

Correspondence should be addressed to Der-Jiunn Deng; djdeng@cc.ncue.edu.tw

Received 15 December 2013; Accepted 25 February 2014; Published 2 April 2014

Academic Editors: H. R. Karimi, Z. Yu, and W. Zhang

This paper investigates a novel joint problem of routing, scheduling, and channel allocation for single-radio multichannel wireless mesh networks in which multiple channel widths can be adjusted dynamically through a new software technology so that more concurrent transmissions and suppressed overlapping channel interference can be achieved. Although the previous works have studied this joint problem, their linear programming models for the problem were not incorporated with some delicate constraints. As a result, this paper first constructs a linear programming model with more practical concerns and then proposes a simulated annealing approach with a novel encoding mechanism, in which the configurations of multiple time slots are devised to characterize the dynamic transmission process. Experimental results show that our approach can find the same or similar solutions as the optimal solutions for smaller-scale problems and can efficiently find good-quality solutions for a variety of larger-scale problems.

1. Introduction

Wireless mesh networks (WMNs) are able to achieve a higher network coverage rate and a larger transmission range through massive deployment of nodes and data exchange between nodes. Aside from the conventional functions of wireless networks, mesh routers in WMNs can dynamically access, set up, and debug one another; that is, even if one of the mesh routers shuts down, the network itself can perform self-recovery and discovery. In general, the practical problems for routing, scheduling, and channel allocation in WMNs are computationally intractable, and, hence, a lot of heuristic and metaheuristic algorithms for efficiently solving the related problems have been proposed, for example, the ant simulated annealing algorithm for the routing problem [1], the approximate dynamic programming approach for link scheduling [2], the local search based algorithm for energy efficient spatial scheduling [3], and the genetic algorithm for the channel assignment problem with partially overlapping channels [4], among others.

In addition to the individual problems for routing, scheduling, and channel allocation in WMNs, the metaheuristic algorithms for the joint problems for WMNs have been investigated; for example, the joint routing and scheduling problem was solved by a genetic algorithm in [5]; the joint routing and channel allocation problem was solved by a simulated annealing algorithm [6] and a genetic algorithm in [7], among others. However, most of the previous joint problems for WMNs were associated with only two of the three concerns, and hence this paper further investigates the joint problem of routing, scheduling, and channel allocation (RSC for short) to address the network scenario more realistically. Another important issue on WMNs is the influence of signal interference on network performance, which can be lessened by using multiple radios and multiple orthogonal channels; for example, see [8, 9]. A recent work in [10] has proposed a software technique with very little overhead that can dynamically adapt the widths of multiple channels at different times so as to attain more concurrent transmissions and reduce signal interference under limited resource of

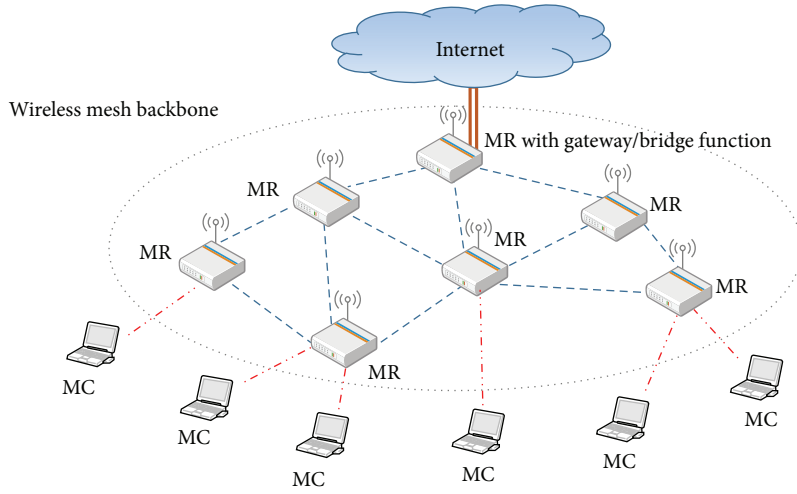


Figure 1: Illustration of a wireless mesh network.

wireless spectrum. Note that although the previous works for the joint RSC problem in [11, 12] have proposed to apply such a variable-width channel allocation scheme to control signal interference, some delicate constraints are not concerned in their linear programming models.

In light of the above, this paper first constructs a linear programming model that better reflects the joint RSC problem in WMNs and then designs the simulated annealing (SA) algorithm with a novel encoding mechanism to solve the problem. Our approach attempts to more realistically address how WMN operates and to rapidly identify feasible paths and schedule, as well as variable-width channel allocation of data transmission in large-scale network topologies. In addition, we further take the influence of signal interference on the channel capacity into consideration to better reflect real network situations. Our experimental results are first compared with the optimal solutions for the small-scale problems that do not consider the influence of signal interference on channel capacity, and then a comprehensive experimental analysis on more complex problems is conducted. Note that the work in [11] has proposed an SA approach for a joint routing, link scheduling, and spectrum allocation, but its linear programming model and our SA approach are different from ours.

The main contributions of this paper are stated as follows:

(i) the SA algorithm with a novel encoding mechanism is designed and implemented to solve a novel joint RSC problem with variable-width channel allocation scheme;

(ii) we establish a linear programming model for the novel joint RSC problem, in which in order to be more realistic than the previous works in [11, 12], we consider that the channel capacity is influenced by the allocated channel width, transmission range, and signal interference;

(iii) we propose an innovative solution encoding design in which the configurations of multiple time slots are devised to characterize the dynamic data transmission process in WMNs.

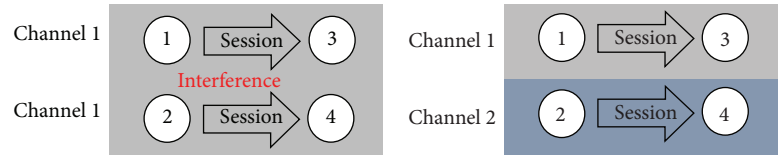
The rest of this paper is organized as follows. Section 2 covers the literature review and channel allocation schemes. Section 3 gives the solution encoding design of our SA algorithm for the concerned problem and then describes the details of the algorithm. Section 4 briefs the designs of all experimental scenarios and their results. Section 5 concludes this paper with future work.

2. Preliminaries

This section first provides reviews on related literatures, then describes the variable-width channel allocation scheme, and finally introduces the formulas on signal interference.

2.1. Related Work. The WMN is a wireless multihop technology that uses all the network devices within the transmission range as relay nodes. Based on a routing mechanism of WMN, data can be transmitted from the source node, along relay nodes on the routing path, and finally to the destination node. As shown in Figure 1, all the devices in the WMN are classified as either a mesh router (or MR) or mesh client (or MC). Besides its basic routing function, the MR in the network also serves as a gateway and bridge to transmit data to the Internet. Sessions are transmitted between MCs and MRs by utilizing the P2P mechanism as specified in the ad hoc mode. To an MC, the support for its hardware and software is far easier than an MR, because the MC does not function as a gateway and bridge. Thus, it only needs a wireless adaptor and the traffic load processed by its protocol is lighter than that of an MR. In terms of mobility, an MR is more stationary than an MC. To an MC, MRs are the mesh backbone of the network.

Most of the previous works related to this paper are dedicated to improving overall network throughput and developing new methods to address the issues on routing, scheduling, and channel allocation. The routing problem is concerned



(a) Single-channel allocation (b) Multichannel allocation

Figure 2: Difference between single-channel and multichannel allocations.

with identifying the optimal path for data transmission; for example, Li et al. [13] proposed an adaptive routing algorithm for multiple subscribers in WMNs; Draves et al. [8] applied the cumulative weighted expected transmission duration method to evaluate whether the multichannel allocation approach can effectively improve network throughput. The scheduling problem is mainly about the priority and sequence of data transmission; for example, Tang et al. [14] constructed a model that can schedule appropriately, maximize output efficiency, and control power and applied an algorithm based upon the time series theory to solve all related issues. The channel allocation problem is concerned with signal interference and concurrent transmission, and an appropriate allocation of limited spectrum resources can maximize the transmission efficiency; for example, Ko et al. [15] developed a modularized distributed channel allocation approach and made channel selection decisions based on the information acquired from the data forwarding mechanism.

As more and more studies are committed for those three problems individually, some works started discussing how the joint problems consisting of two or three individual problems can be improved at once. For instance, for joint routing and scheduling problem, Badia et al. [5] established an integer linear programming model and proposed a genetic algorithm approach for the joint routing and scheduling problem. Li et al. [16] proposed an optimization architecture for joint multipath routing and scheduling in WMNs, while Luo et al. [17] incorporated the greedy algorithm and column generation algorithm to effectively raise the overall network throughput. For the joint routing and channel allocation problem, Raniwala et al. [18, 19] used local traffic information, incorporated with distributed and centralized algorithms, to handle dynamic channel allocation and routing searches. Cheng and Yang [6] proposed a simulated annealing algorithm based upon the encoding of the path oriented mechanism and used the tree data structure to express each candidate solution for multiple paths. Valarmathi and Malmurugan [20] designed a traffic-aware and forecasting mechanism to control routing congestion and used the distributed channel allocation to avoid signal interference within the same channel.

With more complicated concerns, some works focused on the joint problem of routing, scheduling, and channel allocation problem (RSC for short). For instance, Tang and Brandt-Pearce [21] used the free space optics communication, a wireless spectrum technology, to offset insufficient spectrum resources and used a mixed-integer programming approach to solve the joint RSC problem of the two wireless technologies. Uddin et al. [11, 12] adopted the variable-width channel allocation scheme to construct a joint RSC model and solved it with the column generation algorithm and some heuristics.

2.2. Variable-Width Channel Allocation. Channel allocation is an important topic in the study of IEEE 802.11 MAC protocols. A good channel allocation approach can drastically reduce interference for concurrent transmission, increase throughput, and reduce transmission delay [15]. The use of multiple channels may increase the complexity of channel allocation but can further lessen the influence of signal interference on network performance. The difference between single-channel and multichannel allocations [22] is illustrated in Figure 2 in which nodes 1 and 2 transmit their sessions to nodes 3 and 4, respectively. Under single-channel allocation, both of the two links are allocated with channel 1, so that nodes 1 and 2 could interfere with each other during the transmission process. In this case, it is required to set transmission priority for the two sessions. On the other hand, under multichannel allocation, the link between nodes 2 and 4 can be allocated to another channel so that the sessions with different destination nodes can be transmitted concurrently, so the channel capacity is only affected by the signal interference within a channel.

Through settings of software, MRs, as well as MCs, the variable-width channel allocation scheme can be adopted [10]. Besides the standard 20 MHz channel width allocation, different channel widths (including 5 MHz, 10 MHz, 20 MHz, and 40 MHz) for multiple channels can be set according to the transmission range as well as the number of sessions to be transmitted. Thus, the work in [10] quantified the influences of channel width on the transmission range and the number of transmitted sessions, as well as power consumption, and discovered that a smaller channel width is more appropriate for a link with less traffic load, because if a small channel width is allocated, the remaining spectrum band width can be utilized to create more orthogonal channels for other concurrent transmissions.

The work in [23] already proposed a model for the variable-width channel allocation problem and indicated that the variable-width channel allocation scheme can make a fairer and more effective use of limited spectrum resource and clearly perform superior to the conventional fixed-width channel allocation schemes. The work in [24] proposed an algorithm to suppress interference in concurrent transmission and used the variable-width channel allocation scheme to increase the number of orthogonal channels. According

to the work in [24], the variable-width channel allocation scheme can improve the overall network throughput.

2.3. Signal Interference. Consider the transmission between two neighboring nodes i and j in which the Euclidean distance between them is $d_{i,j}$. Assuming absence of interference and a simple 2-ray propagation model, the signal-to-noise ratio (SNR) at the intended receiver and the derived transmission range D_w are given as follows, respectively:

$$\mathrm{SNR}_{i,j} = \frac{P_o d_{i,j}^{-\alpha}}{N_0 W}, \qquad D_w = \sqrt[\alpha]{\frac{P_o}{N_0 W \beta}}, \tag{1}$$

where P_o is the transmission power; α is the path loss exponent; N_0 is the power spectral density of thermal noise; W is the allocated channel width; and β is the required threshold that the SNR must be no less than to achieve a particular data rate. From the above equations, a larger transmission range D_w can be accomplished with a smaller channel width W. That is, if the traffic load on a link is heavier, a larger channel width should be allocated for that link.

Considering the presence of interference from other transmissions, the signal-to-interference-plus-noise ratio (SINR) at receiver j and the Shannon capacity of link (i, j) are, respectively, given as follows:

$$\mathrm{SINR}_{i,j} = \frac{P_o d_{i,j}^{-\alpha}}{N_0 W + \sum_{\alpha \neq i} P_o d_{a,j}^{-\alpha}}, \qquad C_{i,j} = W \cdot \log_2(1+\beta), \quad \text{if } \mathrm{SINR}_{i,j} \geq \beta, \tag{2}$$

where β is the required threshold that the SINR must be no less than to achieve a particular data rate. From the above equations, a larger channel capacity can be achieved if a larger channel width W is allocated.

3. Our SA Approach to the RSC Problem

This section first describes our concerned RSC problem for WMNs, establishes a linear programming model for the problem, and covers the design of the SA algorithm, including the solution encoding mechanism, generation of the initial solution, and search of the neighboring solution.

3.1. Problem Description and Modelling. Consider to transmit M concurrent sessions in an n-node multihop WMN with K orthogonal channels, where each session m is transmitted from node s_m to node d_m and its traffic load is R_m bits for $1 \leq m \leq M$; each channel is allocated with a spectrum band of 5, 10, 20, or 40 MHz at different times; that is, it is a variable-width channel allocation scheme. The transmission is involved with the joint problem of routing, scheduling, and channel allocation as each session may be separated into subsessions that go along different paths to its destination, and those routing paths are chosen according to the underlying schedule of different concurrent transmissions. The objective of our problem is to minimize the total system activation time for transmitting the M concurrent sessions while the minimum SINR requirement for transmissions is satisfied.

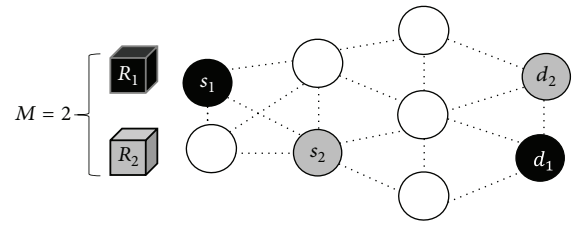
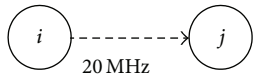


Figure 3: Illustration of transmitting two sessions in a 9-node network topology. The sizes of the two sessions are R_1 an R_2, respectively, and the pairs of their source and destination nodes are (s_1, d_1) and (s_2, d_2), respectively.

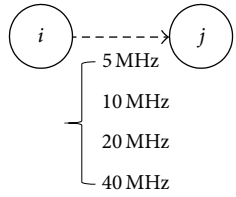
Since different channel widths may influence the transmission range, only the two neighboring nodes within the transmission range (determined by the allocated channel width) and without violating the minimum SINR requirement can communicate with each other. The link in which the two adjacent nodes can communicate with each other is called an active link. Note that the active links at different times are different. Let E denote the set of all possible active links. For example, in Figure 3, two sessions with sizes R_1 and R_2 are transmitted in a 9-node network topology; all the active links are represented by dotted lines; nodes s_1 and d_1 (both in black) are the source and destination nodes for session 1, while nodes s_2 and d_2 (both in gray) are the source and destination nodes for session 2.

For the variable-width channel allocation scheme, this paper continues using the setting of the total spectrum width in [11, 12]: $B = 80$ MHz. Different from the conventional fixed-width channel allocation scheme that allocates 20 MHz to all links for transmission, this paper adopts four possible channel widths: 5, 10, 20, and 40 MHz, as illustrated in Figure 4. If the allocated channel width for a link is smaller, then the transmission range is larger, and there are more orthogonal channels available to be used, but the channel capacity is smaller. On the contrary, if the allocated channel width for a link is larger, then the transmission range is smaller, and there are less remaining spectrum resources that are available to be allocated, but the channel capacity is larger.

In what follows, the linear programming model for our concerned problem is described in detail. We assume a time-division multiple access WMN, in which time is divided into T times slots. Note that each link is active and needs to be allocated with a channel width only when the transmission range constraint is not violated and the SINR requirement is satisfied. Hence, the multihop transmissions via active links at each time slot may take different time; that is, the durations of any two time slots may not be equal. For $1 \leq t \leq T$, let λ_t denote the duration of the tth time slot, which is called the tth system activation time. The objective of our concerned



(a) Fixed-width channel allocation



(b) Variable-width channel allocation

Figure 4: Illustration of the difference between fixed-width and variable-width channel allocation schemes.

problem is to minimize the total system activation time as follows:

$$\text{Minimize} \quad \sum_{t=1}^{T} \lambda_t. \tag{3}$$

Assume that there are K orthogonal channels that can be allocated within a time slot. By using the variable-width channel allocation scheme, the total spectrum width (80 MHz) can be sliced into at least 2 channels of width 40 MHz and up to 16 channels of width 5 MHz, and, hence, $2 \le K \le 16$. Let k denote a particular channel. The link variable $x_{i,j}^{k,t}$ is defined as follows:

$$x_{i,j}^{k,t} = \begin{cases} 1, & \text{if link } (i,j) \text{ on channel } k \text{ is active at time slot } t; \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Suppose that each node is allocated with a single-radio channel; that is, each link within a time slot does not allow any traffic flow from both directions of this link simultaneously and is allocated with only one channel. Hence, the transmission constraint at each time slot is characterized as follows:

$$\sum_{k=1}^{K} \left(\sum_{i \in V:(i,v) \in E} x_{i,v}^{k,t} + \sum_{j \in V:(v,j) \in E} x_{v,j}^{k,t} \right) \le 1, \quad \forall v \in V, \quad \forall t = 1, \ldots, T, \tag{5}$$

where V denotes the set of nodes in the network topology.

The spectrum band width $b^{k,t}$ allocated for channel k at time slot t is given as follows:

$$b^{k,t} = \begin{cases} 0, & \text{if channel } k \text{ is allocated with 0 MHz;} \\ 5, & \text{if channel } k \text{ is allocated with 5 MHz;} \\ 10, & \text{if channel } k \text{ is allocated with 10 MHz;} \\ 20, & \text{if channel } k \text{ is allocated with 20 MHz;} \\ 40, & \text{if channel } k \text{ is allocated with 40 MHz.} \end{cases} \tag{6}$$

Since the total sum of all the K orthogonal channels at time slot t cannot exceed the total spectrum width B, the constraint for $b^{k,t}$ is characterized as follows:

$$\sum_{k=1}^{K} b^{k,t} \le B, \quad \forall t = 1, \ldots, T. \tag{7}$$

If $b^{k,t} = 0$, then there is no spectrum band width allocated for channel k at time slot t; that is, $x_{i,j}^{k,t}$ must be equal to 0. Hence, we have the following constraint:

$$x_{i,j}^{k,t} \le b^{k,t}, \quad \forall (i,j) \in E, \ \forall k = 1, 2, \ldots, K. \tag{8}$$

For link variable $x_{i,j}^{k,t}$ and channel width variable $b^{k,t}$, the allocated channel width and the active status for link (i, j) at time slot t are represented by $G_{i,j}^{t}$ as follows:

$$G_{i,j}^{t} = \begin{cases} b^{k,t}, & \text{if link } (i,j) \text{ is active } \left(\text{i.e., } x_{i,j}^{k,t} = 1\right); \\ 0, & \text{if link } (i,j) \text{ is inactive } \left(\text{i.e., } \sum_{k=1}^{K} x_{i,j}^{k,t} = 0\right). \end{cases} \tag{9}$$

To linearize the definition of $G_{i,j}^{t}$ for alleviation of computing the mathematical model, three large positive constants, M_1, M_2, and M_3, are used to construct this model:

$$G_{i,j}^{t} \le b^{k,t} + M_1 \left(1 - x_{i,j}^{k,t}\right), \quad \forall (i,j) \in E, \ \forall k = 1, 2, \ldots, K, \tag{10}$$

$$G_{i,j}^{t} \le M_2 \sum_{k=1}^{K} x_{i,j}^{k,t}, \quad \forall (i,j) \in E, \tag{11}$$

$$G_{i,j}^{t} \ge b^{k,t} - M_3 \left(1 - x_{i,j}^{k,t}\right), \quad \forall (i,j) \in E, \ \forall k = 1, 2, \ldots, K. \tag{12}$$

From the SNR equation in (1), since the allocated channel width W determines the maximal transmission range D_W, we require the constraint: if the distance between two nodes exceeds this range, sessions cannot be successfully transmitted. From the above derivation, link (i, j) is allocated with channel width $G_{i,j}^{t}$, so the transmission range is $D_{G_{i,j}^{t}}$ by (1), and, hence, we have the following constraint:

$$d_{i,j} \le D_{G_{i,j}^{t}}, \quad \text{if link } (i,j) \text{ is active at } t \ \left(\text{i.e., } x_{i,j}^{k,t} = 1\right), \quad \forall (i,j) \in E, \quad \forall k = 1, 2, \ldots, K. \tag{13}$$

To linearize the above inequality, we have the following:

$$d_{i,j} \le D_{G_{i,j}^{t}} + M_4 \left(1 - x_{i,j}^{k,t}\right), \quad \forall (i,j) \in E, \ \forall k = 1, 2, \ldots, K, \tag{14}$$

where M_4 is a large positive constant.

For each active link (i, j) at each time slot t, if the SINR (i.e., $\text{SINR}_{i,j}^{t}$) is greater than the threshold β, the channel

capacity is not affected; otherwise, the channel capacity drops. The linearized SINR equation is

$$\mathrm{SINR}_{i,j}^{t} \le \frac{P_o d_{i,j}^{-\alpha} x_{i,j}^{k,t} + M_5\left(1 - x_{i,j}^{k,t}\right)}{N_0 G_{i,j}^{t} + \sum_{(a,b)\in E; a \ne i} P_o d_{a,j}^{-\alpha} x_{a,b}^{k,t}}, \quad \forall (i,j) \in E, \; \forall k = 1, 2, \ldots, K, \tag{15}$$

where M_5 is a large positive constant. Let $C_{i,j}^{t}$ denote the Shannon capacity at time slot t, which can be determined by the following constraint:

$$C_{i,j}^{t} \le G_{i,j}^{t} \cdot \log_2\left(1 + \min\left(\mathrm{SINR}_{i,j}^{t}, \beta\right)\right). \tag{16}$$

Note that the above inequality can be linearized easily.

Assuming that $f_{i,j}^{m}$ is the amount of session m flowing through link (i, j), the flow constraints are characterized as follows:

$$\sum_{j\in V:(i,j)\in E} f_{i,j}^{m} - \sum_{j\in V:(j,i)\in E} f_{j,i}^{m} = 0, \quad \forall i \in V - \{s_m, d_m\}, \; \forall m = 1, \ldots, M, \tag{17}$$

$$\sum_{j\in V:(s_m,j)\in E} f_{s_m,j}^{m} - \sum_{j\in V:(j,s_m)\in E} f_{j,s_m}^{m} = R_m, \quad \forall m = 1, \ldots, M, \tag{18}$$

$$\sum_{j\in V:(d_m,j)\in E} f_{d_m,j}^{m} - \sum_{j\in V:(j,d_m)\in E} f_{j,d_m}^{m} = -R_m, \quad \forall m = 1, \ldots, M. \tag{19}$$

Equation (17) enforces that if the concerned node i is neither the source node nor the destination node, each session m flowing into node i must come out eventually. Equations (18) and (19) enforce that both the total outflows of the source node and the total inflows of the destination node for session m must be R_m.

Let $f_{i,j}^{m,t}$ be the amount of session m flowing through link (i, j) during time slot t. That is,

$$\sum_{t=1}^{T} f_{i,j}^{m,t} = f_{i,j}^{m}, \quad \forall (i,j) \in E, \; \forall m = 1, \ldots, M. \tag{20}$$

Hence, the tth system activation time can be calculated as follows:

$$\lambda_t = \max_{(i,j)\in E}\left(\frac{\sum_{m=1}^{M} f_{i,j}^{m,t}}{C_{i,j}^{t}}\right), \quad \forall t = 1, \ldots, T. \tag{21}$$

The following constraint enforces that the total traffic load flowing through link (i, j) must not exceed the maximum capacity that this link can sustain:

$$\sum_{t=0}^{T} \lambda_t \cdot G_{i,j}^{t} \cdot \log_2(1 + \beta) - \sum_{m=1}^{M} f_{i,j}^{m} \ge 0, \quad \forall (i,j) \in E. \tag{22}$$

The following constraints enforce that some variables are nonnegative:

$$f_{i,j}^{m} \ge 0, \qquad \lambda_t \ge 0. \tag{23}$$

To summarize, the objective of our linear programming model is to minimize the total system activation time in (3), under the one-time-slot transmission constraint (5), the channel width constraints (7)–(12), the transmission range constraint (14), the SINR constraint (15), the Shannon capacity constraint (16), the one-session traffic flow constraints (17)–(19), the system activation time constraints (20)-(21), the one-link traffic flow constraint (22), and the nonzero variable constraints (23).

It is worth mentioning that the differences between our model and the previous model for the RSC problem proposed in [11, 12] are the introduction to (14)–(16) and (20)-(21). The detailed explanations are stated as follows:

(i) unlike the previous work, inequality (14) takes into consideration the influence of the allocated channel width on the transmission range. It enforces that the transmission range between two nodes due to the allocated channel width is no less than their Euclidean distance;

(ii) the previous work did not consider SINR in its linear programming model. In this paper, (15) considers that different active links and their different transmission ranges result in different SINRs;

(iii) previous work only considered the total channel capacity constraint for transmission. In this paper, the influence of signal interference on the channel capacity for each link is considered in (14)–(16), and, thus, the channel capacities for each link at each time slot may be different;

(iv) based on the above, different capacities lead to different system activation time for each time slot. Hence, this paper introduces (20)-(21) for calculating the system activation time for each time slot.

3.2. Our SA Algorithm. The SA algorithm [23, 25] is an iterative improvement process inspired from simulating the annealing process and has frequently been used to solve a variety of combinatorial optimization problems, for example, periodic routing problem of a retail distribution system [26] and dynamic facility layout problem [27], among others. The basic idea of the SA is to simulate the annealing process of metal crystals to iteratively refine the solution quality and search for nearly optimal solutions. First, the encoding of any solution for the concerned problem into a system state of the metal is defined, and the relationship between the objective function of the concerned problem and the energy function of the state is specified. Next, the SA algorithm initializes the initial system state as well as some parameters on temperatures. In each iteration of the SA algorithm, the system temperature is decreased to a fixed level according an annealing schedule and a neighboring state of the current system state at the current system temperature is generated.

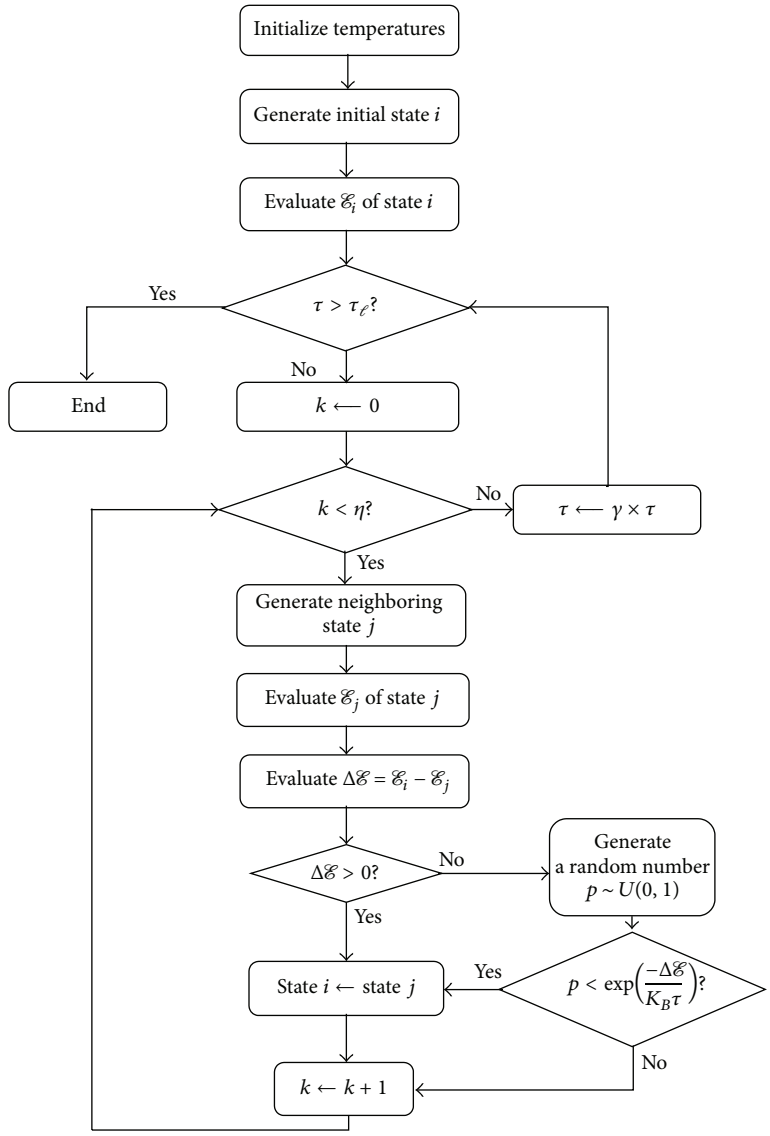


Figure 5: Flow chart of our SA algorithm.

If the energy level of this neighboring state is lower (better) than that of the current state, the neighboring state replaces the current state; otherwise, there is an acceptance probability that allows the neighboring state with a higher (worse) energy level to replace the current state. Once the current state stabilizes at the current temperature, we repeat the same annealing process until the current temperature drops below the lowest temperature. By the acceptance probability mechanism, the system can be released from the trap to constantly seek for a better local optimal solution and could therefore find a global optimal solution for the concerned problem.

The flow chart of the SA algorithm is given in Figure 5. The main steps of the SA algorithm are stated as follows:

(i) set the highest temperature τ_h, the lowest temperature τ_ℓ, and the current temperature $\tau = \tau_h$;

(ii) generate the initial state i and evaluate its energy level $\mathscr{E}_i$;

(iii) change the current state via perturbation to generate a new state j, and evaluate its energy level E_j. State j is also known as the neighboring state of state i;

(iv) calculate the energy difference $\Delta\mathscr{E} = \mathscr{E}_i - \mathscr{E}_j$ between states i and j. If $\Delta\mathscr{E} > 0$, that is, the neighboring state j has a lower energy level, then state j replaces the current state i. Otherwise, generate a random number $p \sim U(0, 1)$ and calculate the acceptance probability $P(\Delta\mathscr{E}, \tau) = \exp(-\Delta\mathscr{E}/(K_B \cdot \tau))$, where K_B is the Boltzmann constant and is usually set to 1 and τ is the current system temperature. If $p < P(\Delta\mathscr{E}, \tau)$, then the neighboring state j is accepted to replace the current system state i; otherwise, it is rejected;

(v) if the maximal number η of iterations is not reached under the current temperature τ, we go back to Step (iii);

(vi) if $\tau > \tau_\ell$, the current temperature τ is decreased according to the annealing schedule $\tau \leftarrow \gamma \times \tau$, where $\gamma \in (0, 1)$, and then we go back to Step (iii); that is, the algorithm stops when $\tau \leq \tau_\ell$.

In light of the above, we need to design the energy function for a system state and some main components of the algorithm specifically for our concerned problem. Since this paper is concerned with a minimization problem and the SA also solves for the minimal energy level, we can directly use (3) as the energy function for the system state. Hence, the remaining components that we have to design include the state encoding mechanism, the method of generating the initial state, and our neighborhood search method, which are detailed in the rest of this section.

3.2.1. State Encoding. To solve our problem with the SA algorithm, we need to design a mechanism to encode the solution of the concerned problem into a system state. For our problem, the information of a system state needs the number of nodes of the network topology, the size of each session, the next hop of session transmission, and the destination node, as well as the allocated channel width. Since time is divided into multiple time slots, we first encode the configuration of a time slot as illustrated in Figure 6(a), which include the following five attributes:

(i) node: this records the ID of a node in the network topology;

(ii) Queues 1–3: since we allow at most three sessions to stay at the same node at a time point, we assume that each node has a queue with three entries labeled by "Queue 1"–"Queue 3." Each entry of the queue can store a session staying at the current node, and its value records its size. If the value is 0, it means that the queue entry is empty;

(iii) Destinations 1–3: for $i = 1, 2, 3$, the "Destination i" entry records the ID of the destination node of the session at "Queue i" of the current node;

(iv) next hop: this records the ID of the next hop node to which the session at "Queue 1" is going to be transmitted. If the ID of the next hop node coincides with that of "Destination 1," then the transmission of the session at "Queue 1" will be finished after this time slot;

(v) channel width: this records the allocated channel width for the link from the current node to the next hop; for example, a 20 MHz channel width is allocated to link (1, 6) in Figure 6(a).

To guarantee the solution feasibility, the configuration of a single time slot needs to satisfy the following constraints:

(i) the queue at each node is first-in-first-out, and, hence, for $i = 1, 2$, if the "Queue i" entry is 0, then the "Queue $i + 1$" entry must be 0;

(ii) if all the queue entries of a node are 0 (meaning that there is no session at the node), then the "Next hop," "Destination," and "Channel width" entries must be 0 for this node;

(iii) if the "Next hop" entry of a node is 0 (meaning that there is no session transmitted from this node), then "Channel width" must be;

(iv) because of limited channel width and signal interference, some sessions may be stranded at the current time slot. For example, in Figure 6(a), the "Next hop" and "Channel width" entries at Node 3 are both 0 for the 1.72 Mbits session, and, hence, this session stays at Node 3 at the current time slot and will only be transmitted at later time slots;

(v) within the same time slot, the total sum of the allocated channel widths for "Channel width" cannot exceed the total spectrum band width, that is, 80 MHz, in this paper.

Take Figure 6(a) as an example for a 6-node network topology for transmission of six sessions: two sessions at Node 1 and one session at each of Nodes 2–5. Figure 6(b) is the encoding for the configuration after finishing the time slot of Figure 6(a), which is explained as follows:

(i) assume that all the allocated channel widths in Figure 6(a) are large enough to allow transmission of each session;

(ii) the session of size 3.58 Mbits at Node 1 in Figure 6(a) disappears in Figure 6(b) because its destination ID is the same as the ID of the next hop in Figure 6(a); that is, it arrives at its destination after this time slot;

(iii) the session of size 4.71 Mbits in "Queue 2" of Node 1 moves to "Queue 1" and their corresponding destinations are adjusted accordingly;

(iv) there is no action for the session of size 1.72 Mbits at Node 3 because no channel width is allocated to the link adjacent to Node 3;

(v) three sessions (of sizes 1.13, 6.55, and 0.71 Mbits, resp.) in Figure 6(a) have the same next hop node (Node 3). However, Node 3 has only two empty queue entries (i.e., "Queue 2" and "Queue 3"). In this case, we arbitrarily select two of the three sessions to be transmitted. In this example (see Figure 6(b)), the sessions of sizes 6.55 and 0.71 Mbits arrive at Node 3, while the session of size 1.13 Mbits stays at the original Node 2.

A complete encoding for a solution of our concerned problem contains configurations of $T + 1$ time slots, as shown in Figure 7. As time slots take turns, the configuration of each time slot determines its own queue and destination entries based on the results from the configuration of the previous time slot and randomly determines its "Next hop" and "Channel width" entries. The same process is repeated until all sessions are transmitted to their final destinations. In choosing the "Next hop" and "Channel width" entries, we need to

Node	1	2	3	4	5	6
Queue 1	3.58	1.13	1.72	6.55	0.71	0
Queue 2	4.71	0	0	0	0	0
Queue 3	0	0	0	0	0	0
Destination 1	6	3	2	5	6	0
Destination 2	4	0	0	0	0	0
Destination 3	0	0	0	0	0	0
Next hop	6	3	0	3	3	0
Channel width	20	10	0	40	10	0

(a) Encoding the current time slot

⇒

Node	1	2	3	4	5	6
Queue 1	4.71	1.13	1.72	0	0	0
Queue 2	0	0	0.71	0	0	0
Queue 3	0	0	6.55	0	0	0
Destination 1	4	3	2	0	0	0
Destination 2	0	0	6	0	0	0
Destination 3	0	0	5	0	0	0
Next hop	5	3	2	0	0	0
Channel width	40	20	20	0	0	0

(b) Encoding the next time slot

Figure 6: Encoding the configuration for a single time slot.

Time slot T

Node	1	2	3	4	5	6
Queue 1	0	0	0	0	2.56	0
Queue 2	0	0	0	0	0	0
Queue 3	0	0	0	0	0	0
Destination 1	0	0	0	0	6	0
Destination 2	0	0	0	0	0	0

Time slot 1

Node	1	2	3	4	5	6	
Queue 1	0	3.47	2.56	0	0	0	0

Time slot 0

Node	1	2	3	4	5	6	
Queue 1	3.47	0	2.56	3.89	0	0	0
Queue 2	0	0	0	0	0	0	0
Queue 3	0	0	0	0	0	0	0
Destination 1	4	0	6	5	0	0	0
Destination 2	0	0	0	0	0	0	0
Destination 3	0	0	0	0	0	0	0
Next hop	2	0	0	5	0	0	
Channel width	40	0	0	20	0	0	

Figure 7: The encoding for a solution of our concerned problem includes the configurations of $T + 1$ time slots.

consider the influences caused by the transmission range to the next hop, the transmission interference constraint, and the variable-width channel allocation. For example, if the transmission distance between nodes is too long, or 40 MHz cannot be used as the maximal channel width, then the 40 MHz option is removed, and one from the remaining options (5, 10, or 20 MHz) is chosen to allocate the channel width. Within each time slot, the allocated channel width and SINR are adopted for the Shannon capacity equation to acquire the system activation time λ_t for each time slot t, and finally all the times are summed up to calculate the total system activation time, which is the energy value of the encoded system state.

3.2.2. Generation of the Initial State. In the beginning of the SA algorithm, an initial system state needs to be decided, and this state should match the initial solution and meet all the constraints described in the previous subsection; that is, this system state must permanently maintain the feasibility of the represented solution. Consider to transmit a number of sessions in a network topology, in which the number of nodes, the distance between nodes of the topology, and the size, the source node, and the destination node of each session are known. The steps of generating the initial state are given as follows:

(i) generate all the queue and destination entries in the configuration of the initial time slot;

(ii) under Constraint (5), one of the nodes to which the session at "Queue 1" can be transmitted is chosen randomly and is assigned to the "Next hop" entry of the session. If the distance $d_{i,j}$ between two nodes is greater than $D_{5\,\mathrm{MHz}}$ (the maximal possible transmission range), then the "Next hop" entry should be chosen again for the session;

(iii) by (1) and Constraint (7), a channel width option is chosen randomly to determine the "Channel width" entry for the session to be transmitted, that is, the one at "Queue 1." If there are no suitable channel width options, both the "Next hop" and "Channel width" entries for the session are set to 0; that is, the session will be transmitted at later time slots;

(iv) go back to Step (ii) until all the "Channel width" and "Next hop" entries are determined;

(v) if there are still sessions that have not yet reached their destinations at this time slot, keep their queue and destination entries, and go back to Step (ii) at the next time slot. If all the sessions within this time slot have reached their destination nodes, the transmission process is finished;

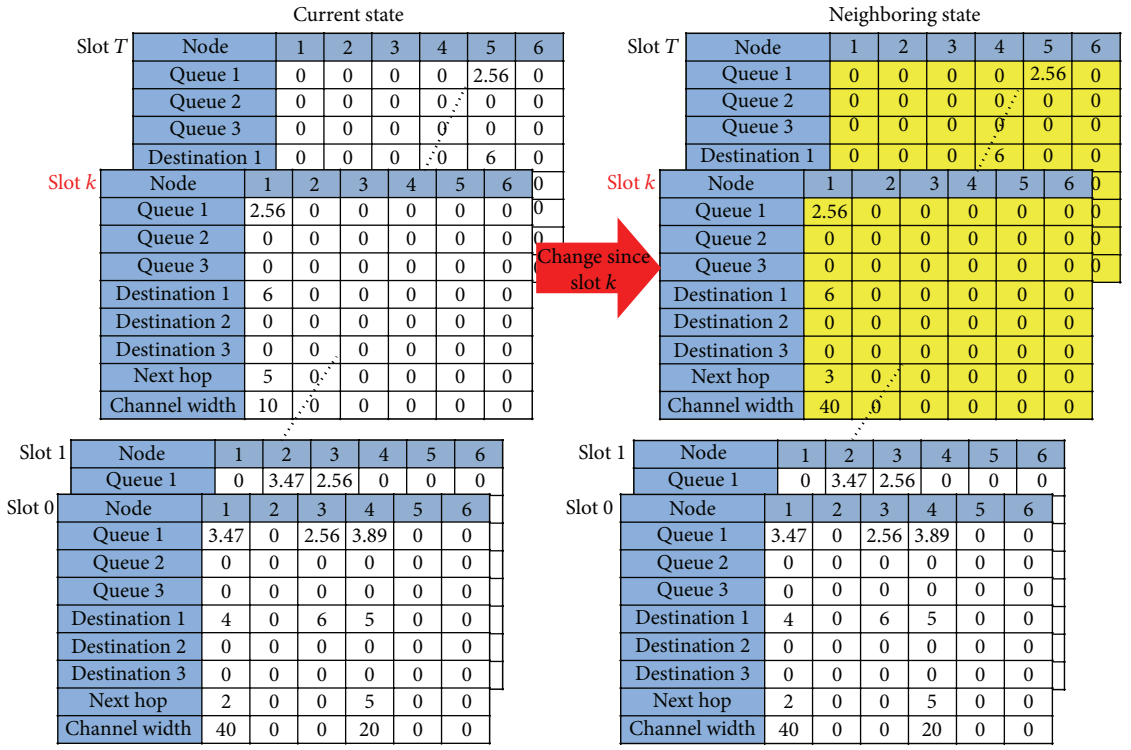


FIGURE 8: Illustration of neighborhood selection.

(vi) calculate and sum up the system activation time of each time slot to obtain the total system activation time, which is the energy value of the initial state.

3.2.3. Neighborhood Selection. This section describes how to search the neighboring state to achieve iterative improvement via the SA algorithm. Consider the current system state that contains configurations of $T + 1$ time slots. We randomly choose a time slot k. In the neighboring state as illustrated in Figure 8, the configurations from time slot 0 to time slot $(k - 1)$ are the same as those of the current state, while the configurations from time slot k to time slot T are changed using the the same procedure of generating the initial state in the previous subsection.

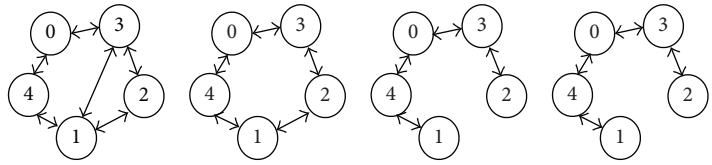
Consider the example in Figure 8, in which k is chosen as the cut-off point to generate the new neighboring state. In the new neighboring state, the configurations from time slot 0 to time slot $(k - 1)$ are kept unchanged as the current state, and the sessions that have not been transmitted to their destination nodes at time slot $(k - 1)$ will be transmitted at later time slots. As shown in Figure 8, for the new neighboring configuration, there is a session (of size 2.56 Mbits) at Node 1 at time slot k and its destination node is still Node 6. The "Next hop" and "Channel width" entries are then randomly chosen. Form Figure 8, it can be seen that for the new entity, "Next hop" = 3 and "Channel width" = 40 MHz at time slot k, unlike the current state, in which "Next hop" = 5 and "Channel width" = 10 MHz at time slot k. Following the same procedure, a new neighboring state will be generated after all sessions are transmitted successfully.

Note that the new neighboring state is not necessarily better than the current state and may even have a worse energy value. Thus, an iterative improvement process must be carried out with the SA algorithm to accept an inferior solution and skip the local optimal solution before a superior feasible solution or a nearly optimal solution can be found.

4. Experimental Results and Analysis

In order to evaluate the performance of our proposed approach, we implement the proposed algorithm and conduct a variety of experiments. Since our joint RSC problem is new, we first compare our experimental results for a small-scale problem (in which the SINR is not considered) with the optimal solutions generated by the column generation algorithm. Note that column generation is an exact algorithm, which is only suitable for small-scale problems. Then, we evaluate the performance of our SA algorithm for larger-scale problems with the concern on the SINR caused by the transmission range between nodes. Our algorithm is implemented in C++ programming language and runs on an Intel Core i5-3210 M @ 2.50 GHz CPU PC with 8 GB memory to analyze the performance. The parameters settings for our SA algorithm in the experiments are listed as follows: Kotzman parameter $K_B = 1$; highest temperature $T_h = 1000$; lowest temperature $T_\ell = 10$; cooling constant $\gamma = 0.95$; and maximum number of iterations $\eta = 10000$.

4.1. Experimental Results for Small-Scale Problems. We continue using the problem instance (where SINR interference is



(a) 5 MHz (b) 10 MHz (c) 20 MHz (d) 40 MHz

Figure 9: The network topology of adopting the channel width of (a) 5 MHz, (b) 10 MHz, (c) 20 MHz, and (d) 40 MHz.

not considered) and parameter settings in [11] as follows: P_o = 1 m Watt, $\alpha = 2$, $N_0 = 10^{-6}$ Watt/MHz, and $\beta = 1.3$.

Recall that the variable-width channel allocation scheme is applied. The network topology considered in [11] using 5 MHz, 10 MHz, 20 MHz, and 40 MHz channel widths is shown in Figures 9(a)–9(d), respectively. As compared with Figure 9(a), the topology in Figure 9(b) does not have link (1, 3), implying that the adoption of 5 MHz channel width can transmit farther. Similarly, for Figures 9(c) and 9(d), the greater the allocated channel width is, the shorter the sessions can be transmitted. Our experiment in this subsection is to transmit the three sessions in Table 1 in this topology.

The problem in the network scenario is solved by our SA algorithm with 5 MHz fixed-width, 40 MHz fixed-width, and variable-width channel allocation schemes, respectively, and the experimental results compared with the optimal solutions generated by the column generation algorithm are given in Tables 2–4. From the result of adopting 5 MHz fixed-width channel allocation scheme in Table 2, our SA algorithm can find a solution with the same solution as the optimal solution generated by the column generation algorithm, though the orders of active links are different. As for the running time, our SA algorithm with the 5 MHz fixed-width channel allocation scheme only takes 3.739 seconds to obtain the solution.

From the result of adopting 40 MHz fixed-width channel allocation scheme in Table 3, the total system activation time found by our SA algorithm is 2.25908 seconds, which is slightly slower than 2.11524 seconds by the column generation algorithm. In this case of using 40 MHz fixed-width channel allocation scheme, the allocated channel width increases and the transmission range decreases, and, hence, more hops are needed so that the SA algorithm is harder to acquire the optimal solution. However, our algorithm can still acquire the nearly optimal solution in 21.949 seconds, showing the computational efficiency of our algorithm. Note that we only compare the minimal and the maximal fixed-width channel allocation schemes with the variable-width channel allocation scheme, because the results of 10 MHz and 20 MHz are between those of 5 MHz and 40 MHz.

From the result of adopting the variable-width channel width scheme in Table 4, the total system activation time found by the SA algorithm is 1.51011 seconds, which is slightly slower than 1.33181 seconds by the column generation algorithm. Compared with the result of adopting the 5 MHz fixed-width channel allocation scheme in Table 2, the total system activation time is 4.66991 seconds faster or shows 76% improvement; compared with the result of adopting the 40 MHz fixed-width channel allocation scheme in Table 3, the total system activation time is 0.74897 seconds faster or shows 33% improvement. Thus, it can be concluded that although our SA algorithm with variable-width channel width allocation scheme may not find the optimal solution as compared to the column generation algorithm, it can still find good-quality feasible solutions, and it is more efficient than the column generation algorithm.

Table 1: The size, source, and destination nodes of three sessions.

Session number	Session size	Source	Destination
1	27.4085 Mbits	Node 3	Node 1
2	6.914 Mbits	Node 0	Node 3
3	9.72211 Mbits	Node 1	Node 2

4.2. Experimental Results for Larger-Scale Problems. This subsection focuses on the changes of the number of session transmissions in the topologies of different sizes to evaluate the performance difference between variable-width and fixed-width channel allocation schemes. Different from the results in previous subsection, the influences of node locations and transmission ranges between nodes are considered to reflect the impacts of SINR on the channel capacity.

4.2.1. Scenario Test of Transmitting 5 Sessions in a 10-Node Network. The network topology considered in this experiment contains 10 nodes on a 100 m × 100 m area, which are nodes #1—#10 in Figure 10. In this topology, we restrict that the y-coordinates of all the 10 nodes are no greater than 50, because as the space used for the topology is reduced to half, the probability of *session strain* can greatly decrease. Note that the *session strain* refers to the fact that the initial position of a session happens to be on an outlier node and even though a 5 MHz channel width (the channel width with the ability to transmit sessions to the longest range) is allocated, the session still cannot be transmitted even to the next node, causing the session to strain at the initial node permanently. The problem in this experiment is to transmit the five sessions of different traffic loads in Table 5 in the 10-node network topology.

The results of executing 30 runs of our SA algorithm with variable-width, 5 MHz fixed-width, and 40 MHz fixed-width channel allocation schemes are plotted in Figure 11, from which it is obvious to see that the performance of both variable-width and 40 MHz fixed-width channel allocation schemes is better than the 5 MHz fixed-width scheme. The averages of their total system activation times of executing 30

Table 2: Result of 5 MHz fixed-width channel allocation.

5 MHz fixed-width	Column generation		Our approach	
Time slot t	Active links	Activation time (sec)	Active links	Activation time (sec)
1	(1, 2)	0.467333	(0, 3), (1, 2)	1.61815
2	(3, 1)	4.56187	(3, 1)	4.56187
3	(0, 3), (1, 2)	1.15082	—	—
Total activation time		6.18002		6.18002

Table 3: Result of 40 MHz fixed-width channel allocation.

40 MHz fixed-width	Column generation		Our approach	
Time slot t	Active links	Activation time (sec)	Active links	Activation time (sec)
1	(0, 3), (1, 4)	0.367965	(0, 3)	0.143846
2	(4, 0)	0.202269	(1, 4)	0.202269
3	(0, 3), (1, 4)	0.202269	(3, 0), (4, 0)	0.570234
4	(3, 0), (4, 1)	0.570234	(0, 4)	0.570234
5	(0, 4), (3, 2)	0.202269	(0, 3), (4, 1)	0.570234
6	(4, 1)	0.570234	(3, 2)	0.202269
Total activation time		2.11524		2.25908

runs of our SA algorithm with the three channel allocation schemes are 4.6255, 16.1476, and 5.3161 seconds, respectively. In average, compared to the 5 MHz fixed-width channel allocation scheme, the variable-width scheme takes about 11.5 seconds less time, or 249% improvement, which is truly remarkable. As for the 40 MHz fixed-width channel allocation scheme, the variable-width scheme takes 0.6906 seconds less time, or 15% improvement.

Although the variable-width channel allocation scheme performs better than the 40 MHz fixed-width channel allocation scheme in average, they still cross with each other in Figure 11 and their performance difference is not immediately discernable. Thus, the t-test was conducted to check if the variable-width channel allocation scheme performs statistically better than the 40 MHz fixed-width scheme. The t-test result shows that the significance level of Levene's test for equality is 0.964 and greater than 0.05, implying that variance has to be set the same. In the t-test with the same means, the significance level is 0.012, which is lower than 0.05. Thus, we can conclude that at the 95% confidence level, the performances of the variable-width and 40 MHz fixed-width channel allocation schemes are significantly different, and the total system activation time of the variable-width channel allocation scheme is clearly shorter than that of the 40 MHz fixed-width scheme.

4.2.2. Scenario Test of Transmitting 7 Sessions in a 10-Node Network. In addition to the five sessions in Table 5, two more sessions in Table 6 are considered in the same network topology in the previous subsection, to evaluate the performance with heavier traffic loads.

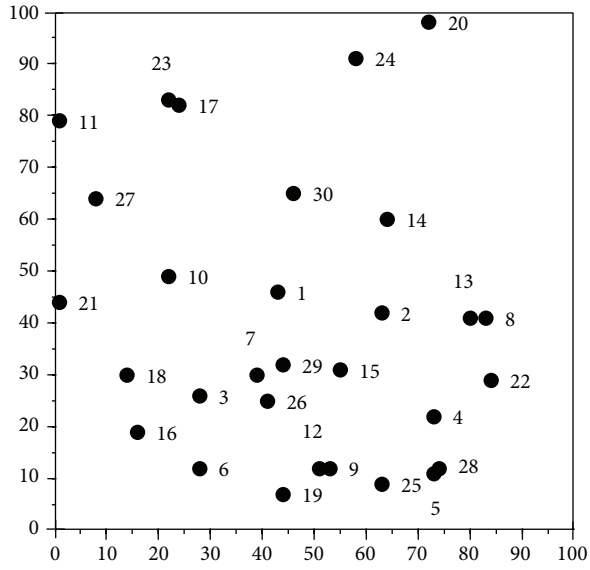
The results of executing 30 runs of our SA algorithm for the same problem of the previous subsection with the addition of transmitting 2 more sessions are plotted in Figure 12, in which the averages of their total system activation times of adopting variable-width, 5 MHz fixed-width, and 40 MHz fixed-width channel allocation schemes are 6.2013, 20.465, and 7.7593 seconds, respectively; that is, all of them are increased as compared with the case of transmitting 5 sessions, and the increasing rates are 34.07%, 26.74%, and 45.96%, respectively. This implies that, in the same topology, the heavier the traffic load is, the more the orthogonal channels are required to transmit more sessions concurrently and reduce the required total system activation time. As shown in Figure 12, the results of the variable-width and 40 MHz fixed-width channel allocation schemes do not significantly cross with each other, and our conducted t-test also shows that the significance level is higher, indicating that, as traffic load grows, the performance of the variable-width channel allocation scheme gets better.

4.3. Scalability Test of Simulated Annealing Algorithm. The experiment of increasing both the numbers of nodes and sessions is conducted to evaluate performance of our SA algorithm in more complex network topologies with 20 nodes and 30 nodes, respectively (which contains nodes #1—#20 and nodes #1—#30 in Figure 10, resp.). Besides the seven sessions specified in Tables 5 and 6, three additional sessions are added in this experiment to increase the transmission traffic, and their information is shown in Table 7.

The results of executing 30 runs of our SA algorithm to transmit 10 sessions in 10-node, 20-node, and 30-node networks are shown in Figure 13, in which the averages of their total system activation times are 14.8829, 18.6813, and 21.0589 seconds, respectively. In the average, when the number of nodes as well as the level of problem complexity increases, the result is worse, and a better solution cannot be attained as expected with the increase in number of available paths. From Figure 13, the results are different for different number of nodes, indicating that, under the condition that the numbers of sessions to be transmitted are the same,

Table 4: Result of variable-width channel allocation.

Variable channel width	Column generation		Our approach	
Time slot t	Active links	Activation time (sec)	Active links	Activation time (sec)
1	(1, 4), 40 MHz (3, 0), 40 MHz	0.158315	(1, 2), 10 MHz (3, 1), 5 MHz (3, 0), 40 MHz	0.50337
2	(3, 2), 40 MHz (4, 1), 40 MHz	0.158315	(0, 3), 40 MHz (0, 4), 40 MHz	0.50337
3	(3, 0), 40 MHz (4, 1), 40 MHz	0.254402	(4, 1), 40 MHz	0.50337
4	(1, 2), 10 MHz (4, 0), 40 MHz	0.030858	—	—
5	(0, 3), 40 MHz (1, 2), 10 MHz	0.159686	—	—
6	(0, 4), 40 MHz (1, 2), 10 MHz	0.570234	—	—
Total activation time		1.33181		1.51011



(x, y)-coordinates of 30 nodes:

#1: (43, 46)	#11: (1, 79)	#21: (1, 44)
#2: (63, 42)	#12: (51, 12)	#22: (84, 29)
#3: (28, 26)	#13: (80, 41)	#23: (22, 83)
#4: (73, 22)	#14: (64, 60)	#24: (58, 91)
#5: (73, 11)	#15: (55, 31)	#25: (63, 9)
#6: (28, 12)	#16: (16, 19)	#26: (41, 25)
#7: (39, 30)	#17: (24, 82)	#27: (8, 64)
#8: (83, 41)	#18: (14, 30)	#28: (74, 12)
#9: (53, 12)	#19: (44, 7)	#29: (44, 32)
#10: (22, 49)	#20: (72, 98)	#30: (46, 65)

Figure 10: The locations of the 30 nodes in a network.

the topology with a higher number of nodes can attain a better result than the topology with a smaller number nodes. Although our SA algorithm may not always obtain the optimal solutions, it can still acquire good-quality solutions within a short time.

Table 5: Size, source, and destination nodes of five sessions.

Session number	Session size	Source	Destination
1	15.433 Mbits	Node 6	Node 8
2	6.841 Mbits	Node 1	Node 7
3	9.72211 Mbits	Node 3	Node 2
4	4.567 Mbits	Node 2	Node 9
5	17.524 Mbits	Node 8	Node 10

Table 6: Besides the five sessions listed in Table 5, two more sessions are added in this experiment.

Session number	Sessions	Source	Destination
6	19.713 Mbits	Node 7	Node 3
7	8.428 Mbits	Node 10	Node 1

Table 7: Besides the seven sessions in Tables 5 and 6, three additional sessions are added in this experiment.

Session number	Sessions	Source	Destination
8	4.51 Mbits	Node 4	Node 5
9	16.244 Mbits	Node 5	Node 6
10	8.577 Mbits	Node 9	Node 4

5. Conclusion and Future Work

This paper has proposed a simulated annealing (SA) algorithm for the joint routing, scheduling, and channel allocation problem in WMNs, in which the variable-width channel allocation scheme is considered to find a delicate balance between mutually conflicting goals, such as concurrent session transmissions and signal interference. A novel encoding design of the SA algorithm is utilized in this

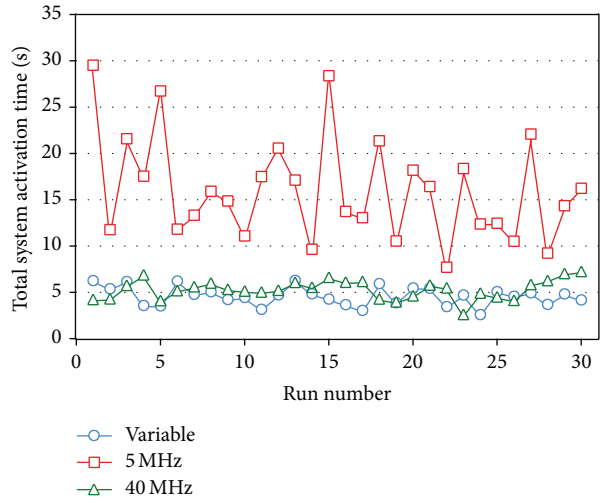


Figure 11: Line chart of the experiment result of transmitting 5 sessions in a 10-node network.

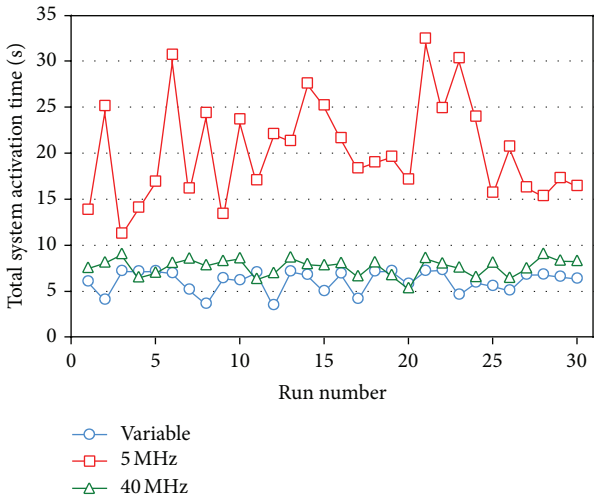


Figure 12: Line chart of the experiment result of transmitting 7 sessions in a 10-node network.

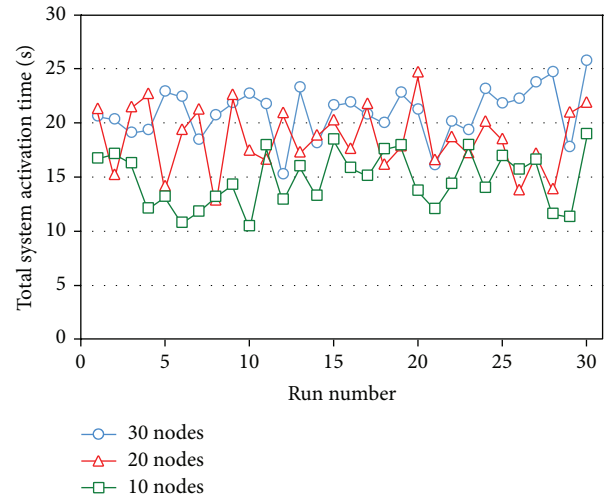


Figure 13: Line chart of the experiment result of transmitting 10 sessions in networks with different numbers of nodes.

paper to illustrate the dynamic transmission process. Our experimental results show that, in comparison to the column generation algorithm for a small-scale problem, in which the SINR is not considered, our SA algorithm can acquire the same or similar solutions as the optimal solutions generated by the column generation algorithm. As compared to fixed-width channel allocation schemes, the variable-width scheme preforms better than the fixed-width scheme. As for the scalability test of our SA algorithm, when the number of nodes grows, the total system activation time also increases. While this result is attributable to the increase in problem complexity and difficulty for identification of the optimal path, causing drop in efficiency, the SA algorithm can still find a good-quality feasible solution. Also, different from the column generation algorithm for exact optimal solutions, our SA algorithm is more efficient to solve the joint problem.

A line of future work is to design the solution encoding mechanism, or integrate our SA algorithm with other artificial intelligent approaches to solve the problem. It would also be of interest to extend the joint problem with mobile nodes or investigate the multiradio multichannel version of the joint problem.

Conflict of Interests

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

Acknowledgments

The authors thank the anonymous referees for the comments that improved the content as well as the presentation of this

paper. This work has been supported in part by NSC 102-2219-E-009-013, NSC102-2410-H156-008, and NSC 102-2221-E018-012-MY3, Taiwan.

References

[1] Y. Zhao, R. Sun, and L. F. Xu, "An ant simulated annealing routing algorithm for wireless mesh network," in *Proceedings of the 7th International Conference on Internet Technology and Applications (ITAP '10)*, pp. 1–4, IEEE Press, Wuhan, China, August 2010.

[2] K. Papadaki and V. Friderikos, "Approximate dynamic programming for link scheduling in wireless mesh networks," *Computers & Operations Research*, vol. 35, no. 12, pp. 3848–3859, 2008.

[3] D. Quintas and V. Friderikos, "Energy efficient spatial TDMA scheduling in wireless networks," *Computers & Operations Research*, vol. 39, no. 9, pp. 2091–2099, 2012.

[4] Y. Ding, Y. Huang, G. Zeng, and L. Xiao, "Channel assignment with partially overlapping channels in wireless mesh networks," in *Proceedings of the 4th Annual International Conference on Wireless Internet (WICON '08)*, Maui, Hawaii, USA, 2008.

[5] L. Badia, A. Botta, and L. Lenzini, "A genetic approach to joint routing and link scheduling for wireless mesh networks," *Ad Hoc Networks*, vol. 7, no. 4, pp. 654–664, 2009.

[6] H. Cheng and S. Yang, "Joint QoS multicast routing and channel assignment in multiradio multichannel wireless mesh networks using intelligent computational methods," *Applied Soft Computing*, vol. 11, no. 2, pp. 1953–1964, 2011.

[7] T.-Y. Lin, K.-C. Hsieh, and H.-C. Huang, "Applying genetic algorithms for multiradio wireless mesh network planning," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 5, pp. 2256–2270, 2012.

[8] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom '04)*, pp. 114–128, ACM Press, Philadelphia, Pa, USA, October 2004.

[9] N. Kumar, N. Chilamkurti, J. H. Park, and D.-S. Park, "Load balancing and adaptive scheduling for data intensive prioritised traffic in multi-radio multi-channel wireless mesh networks," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 12, no. 1, pp. 3–13, 2013.

[10] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, and P. Bahl, "A case for adapting channel width in wireless networks," in *Proceedings of the 14th Annual Conference of the ACM Special Interest Group on Data Communication (SIGCOMM '08)*, pp. 135–146, ACM Press, Seattle, Wash, USA, August 2008.

[11] M. F. Uddin, H. M. K. Alazemi, and C. Assi, "Optimal flexible spectrum access in wireless networks with software defined radios," *IEEE Transactions on Wireless Communications*, vol. 10, no. 1, pp. 314–324, 2011.

[12] M. F. Uddin and C. Assi, "Joint routing and scheduling in WMNs with variable-width spectrum allocation," *IEEE Transactions on Mobile Computing*, vol. 12, no. 11, pp. 2178–2192, 2013.

[13] Y. Li, Y. Xiong, L. Zhou, and R. Zhu, "Adaptive optimization-based routing in wireless mesh networks," *Wireless Personal Communications*, vol. 56, no. 3, pp. 403–415, 2011.

[14] J. Tang, G. Xue, C. Chandler, and W. Zhang, "Link scheduling with power control for throughput enhancement in multihop wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 3, pp. 733–742, 2006.

[15] B.-J. Ko, V. Misra, J. Padhye, and D. Rubenstein, "Distributed channel assignment in multi-radio 802.11 mesh networks," in *Proceedings of the 5th IEEE Wireless Communications and Networking Conference (WCNC '07)*, pp. 3978–3983, IEEE Press, Kowloon, Hong Kong, March 2007.

[16] Y. Li, L. Zhou, Y. Yang, and H.-C. Chao, "Optimization architecture for joint multi-path routing and scheduling in wireless mesh networks," *Mathematical and Computer Modelling*, vol. 53, no. 3-4, pp. 458–470, 2011.

[17] J. Luo, C. Rosenberg, and A. Girard, "Engineering wireless mesh networks: joint scheduling, routing, power control, and rate adaptation," *IEEE/ACM Transactions on Networking*, vol. 18, no. 5, pp. 1387–1400, 2010.

[18] A. Raniwala, K. Gopalan, and T. C. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *Mobile Computing and Communications Review*, vol. 8, no. 2, pp. 50–65, 2004.

[19] A. Raniwala and T.-C. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proceedings of the 24th Annual Joint Conference of IEEE Computer and Communications Societies (INFOCOM '05)*, vol. 3, pp. 2223–2234, IEEE Press, Miami, Fla, USA, March 2005.

[20] K. Valarmathi and N. Malmurugan, "Distributed channel assignment and congestion control routing in wireless mesh networks," in *Proceedings of the 3rd IEEE International Conference on Electronics Computer Technology (ICECT '11)*, pp. 148–153, IEEE Press, Kanyakumari, India, April 2011.

[21] Y. Tang and M. Brandt-Pearce, "Link allocation, routing and scheduling of FSO augmented RF wireless mesh networks," in *Proceedings of the 15th IEEE International Conference on Communications (ICC '12)*, pp. 3139–3143, IEEE Press, Ottawa, Canada, 2012.

[22] W. H. Su, T. Y. Lin, and G. C. Yin, "Multi-channel scheduling (MCS) for IEEE 802.11 CSMA-based wireless networks," in *Proceedings of the 3rd International Conference on Advanced Information Technologies (AIT '09)*, IEEE Press, 2009.

[23] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[24] R. Gummadi, R. Patra, H. Balakrishnan, and E. Brewer, "Interference avoidance and control," in *Proceedings of the 7th Workshop on Hot Topics in Networks (Hotnets-VII '08)*, pp. 13–18, ACM Press, Calgary, Canada, 2008.

[25] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[26] A. Hamzadayi, S. Topaloglu, and S. Y. Kose, "Nested simulated annealing approach to periodic routing problem of a retail distribution system," *Computers & Operations Research*, vol. 40, no. 12, pp. 2893–2905, 2013.

[27] A. R. McKendall Jr., J. Shang, and S. Kuppusamy, "Simulated annealing heuristics for the dynamic facility layout problem," *Computers & Operations Research*, vol. 33, no. 8, pp. 2431–2444, 2006.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 395212, 11 pages
http://dx.doi.org/10.1155/2014/395212

Research Article

SVM-Based Spectrum Mobility Prediction Scheme in Mobile Cognitive Radio Networks

Yao Wang,[1,2] Zhongzhao Zhang,[1] Lin Ma,[1] and Jiamei Chen[1]

[1] *Communication Research Center, Harbin Institute of Technology, Harbin 150080, China*
[2] *Communication Department, Shenyang Artillery Academy, Shenyang 110867, China*

Correspondence should be addressed to Zhongzhao Zhang; zzzhang@hit.edu.cn

Received 27 December 2013; Accepted 19 February 2014; Published 30 March 2014

Academic Editors: H. R. Karimi, Z. Yu, and W. Zhang

Spectrum mobility as an essential issue has not been fully investigated in mobile cognitive radio networks (CRNs). In this paper, a novel support vector machine based spectrum mobility prediction (SVM-SMP) scheme is presented considering time-varying and space-varying characteristics simultaneously in mobile CRNs. The mobility of cognitive users (CUs) and the working activities of primary users (PUs) are analyzed in theory. And a joint feature vector extraction (JFVE) method is proposed based on the theoretical analysis. Then spectrum mobility prediction is executed through the classification of SVM with a fast convergence speed. Numerical results validate that SVM-SMP gains better short-time prediction accuracy rate and miss prediction rate performance than the two algorithms just depending on the location and speed information. Additionally, a rational parameter design can remedy the prediction performance degradation caused by high speed SUs with strong randomness movements.

1. Introduction

Cognitive radio (CR) as a solution for the next generation wireless networks brings new hope to address the wireless spectrum inefficiency problem which has attracted a great deal of attention in recent years [1–4]. In general, CR paradigms are classified in three types: interweave, underlay, and overlay. In interweave or opportunistic spectrum access (OSA) model [5, 6], CUs can use the licensed spectrums opportunistically when the spectrums are detected idle by spectrum sensing. It is very sensitive to PU traffic pattern and it relies on the detection error for the models [6, 7]. Thus, it is essential to investigate the spectrum mobility which is the foundation of resource allocation and network construction.

In a CRN, the spectrum mobility for CUs includes two aspects: spectrum mobility in the time domain and spectrum mobility in the space domain [8]. The time-varying and space-varying characteristics of the spectrum mobility lead to the problem that it is hard to access the licensed spectrums for CUs in a real network. Time-varying characteristic is because of the random variations of PUs' arrivals and departures. Thus, some related literatures have focused on the impact of PUs' activity on CRNs [9–11]. In [12], a selective opportunistic spectrum access scheme is proposed with the aid of PUs' traffic prediction techniques. The scheme can estimate the probability of a channel being idle and choose the best order of spectrum sensing to maximize spectrum efficiency. The definition of channel availability vector is introduced to characterize the state information of licensed channels [13]. And a prediction-based sensing approach is presented to maximize system throughput which reduces the sensing time. In [14], a forecast scheme of call arrival rate and call holding time for PUs is proposed. CUs can reduce the frequency hopping rate through the traffic pattern prediction of PUs.

In the space domain, the movement of CUs directly results in the changing of the spectrum availability. Nevertheless, the movement of CUs, as one of the most important factors in wireless communication systems, is not adequately discussed for CRNs in existing works. A mobility model

describing airborne nodes is proposed in [15]. And a stability-capacity-adaptive routing scheme is proposed to achieve high throughput and small transmission time based on the model. In [16], an optimal power control algorithm in mobile CR ad hoc networks is proposed. Without causing harmful interference to PUs, the network achieves maximized throughput based on the algorithm in the legacy network. In [17], a cluster-based routing protocol which can increase throughput and reduce data delivery latency is presented to mend the route in mobile CRNs. In [18], a general scheduling framework with the mobility information is conducted to solve maximum throughput channel scheduling problem for mobile CRNs. And two polynomial time optimal algorithms are proposed and evaluated by using the mobility trace obtained from a real public transportation system.

However, few of existing works investigate the following two issues: (1) considering time domain and space domain characteristics of spectrum mobility together and (2) considering the prediction of spectrum mobility. In practice, a CRN should be forward looking rather than reactive [19]. And a prediction-based CRN can not only improve system performance but also minimize interference to PUs [20–22], because spectrum detecting may take a long time or delay. In [23], a neural network based channel status predictor using multilayer perceptron is proposed. The system spectrum utilization is improved and the sensing energy is saved greatly by predicting the idle channels. In [24], a channel handoff scheme based on SVM is presented to reduce the handoff time. The channel handoff caused by the random movement of PUs and CUs is considered in the prediction design. In [25], a binary time series approach is used to predict the future occupancy of neighboring channels. This approach performs very well for deterministic occupancy even without updating data.

The key contributions of this paper are as follows: (1) we first take the two issues discussed above into account at the same time. And an effective joint feature vector extraction scheme is originally designed through the theoretical analysis on joint information of CUs' mobility and PUs' working activities. (2) Based on the extracted joint feature vector, a novel SVM-based spectrum mobility prediction scheme considering the time-space domain of spectrum mobility together is proposed for mobile CRNs in order to ameliorate the traditional prediction methods only utilizing the location and speed information directly. (3) Finally, simulations are conducted to confirm the effectiveness of the proposed prediction mechanism. The new prediction mechanism achieves higher short-time prediction performance than the conventional algorithms with little training nodes, which is vital in CRNs.

The rest of the paper is organized as follows. The system model is described in Section 2 and the spectrum availability of SUs is discussed in Section 3. In Section 4, a spectrum mobility prediction scheme is proposed based on SVM. The simulation results are shown along with a discussion in Section 5. At last, Section 6 concludes the paper.

2. System Model

2.1. Mobile CRN System Model. In this paper, we consider a mobile CRN scenario where N_c CUs coexist with N_p PUs illustrated in Figure 1. Assume that each PU_p $(p = 1 : N_p)$ has a licensed access to a spectrum c_p with a coverage radius R_p. Thus, the number of PUs is equal to the number of spectrums in the network. Each CU_c $(c = 1 : N_c)$, with an interference radius r, can exploit locally unused licensed spectrum opportunistically without causing any interference to the corresponding PU_p. Suppose that the CRN assigns spectrums periodically with an allocation interval time T_c which is the interval time between two times of spectrum allocation. We also assume spectrum sensing is ideal in this paper.

Figure 1 gives out an instantaneous snapshot of a mobile CRN deployment with 20 mobile CUs. Two PUs are located in the area. The activity of each PU_p ($p = 2$ in Figure 1) is characterized as an on/off (busy/idle) model. The busy time and idle time of PU_pcan be modeled by the exponential distribution with means α_p and β_p, respectively [26–28]. The probability density function (PDF) can be written, respectively, as

$$f_{\mathrm{ON}}\left(t, \alpha_p\right) = \frac{1}{\alpha_p} e^{-t/\alpha_p}, \quad t \geq 0,$$
$$f_{\mathrm{OFF}}\left(t, \beta_p\right) = \frac{1}{\beta_p} e^{-t/\beta_p}, \quad t \geq 0. \tag{1}$$

In this paper, a random mobility model which characterizes the movement of CUs in a two-dimensional space is considered [29]. The movement of each CU_c consists of a sequence of random length intervals called mobility epochs during which CU_c moves at a constant speed in a constant direction. And the mobility epoch lengths T_e are independently exponentially distributed with mean $1/\lambda_e$. The probability distribution function can be expressed as

$$M_e(x) = P\left(T_e \leq x\right) = 1 - e^{-\lambda_e x}. \tag{2}$$

During each epoch, the mobile direction of CU_c is uniformly distributed over $[0, 2\pi)$ and the speed of CU_c is uniformly distributed over $[0, v_{\max}]$. We assume mobility is uncorrelated among all the CUs in a network. And it is reasonable to assume that epoch length, speed, and direction are uncorrelated in the model. Figure 2 shows a mobility trajectory of one given CU_c as an example.

Definition 1. Given a licensed spectrum c_p and an instantaneous time t, the instantaneous spectrum availability $\mathrm{ISA}_p^c(t)$ for one CU_c can be defined as

$$\mathrm{ISA}_p^c(t) = \begin{cases} -1 & D_{p,c}(t) < R_p + r_c \cap \left(\alpha_p(t) = -1\right) \\ 1 & D_{p,c}(t) < R_p + r_c \cap \left(\alpha_p(t) = 1\right) \\ 1 & D_{p,c}(t) \geq R_p + r_c. \end{cases} \tag{3}$$

$\mathrm{ISA}_p^c(t) = 1$ means that licensed spectrum c_p is instantaneously available at t for CU_c and $\mathrm{ISA}_p^c(t) = -1$ means

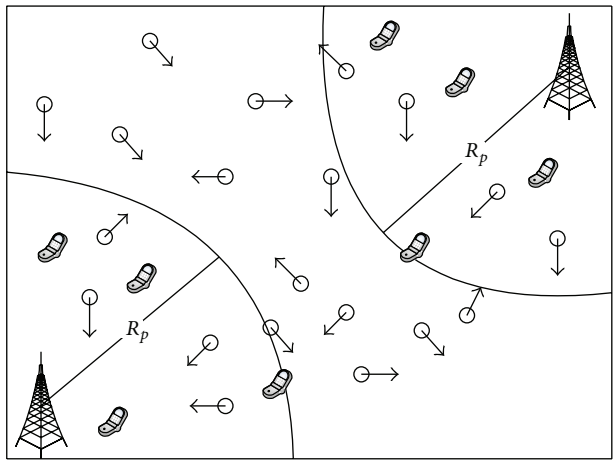


Figure 1: System model of a mobile CRN.

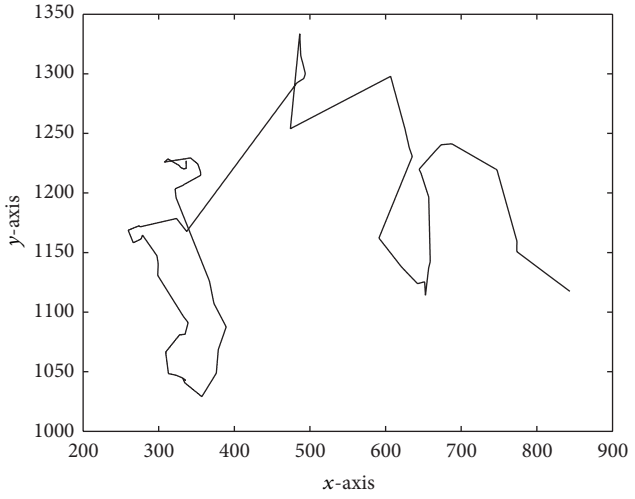


Figure 2: A mobility trajectory example for one given CU_c.

that licensed spectrum c_p is not instantaneous available at t for CU_c, where $D_{p,c}(t)$ represents the Euclidean distance between PU_p and CU_c at t. $\alpha_p(t)$ represents the time-varying working activity of PU_p, denoted as

$$\alpha_p(t) = \begin{cases} 1 & PU_p \text{ is idle at } t \\ -1 & PU_p \text{ is busy at } t. \end{cases} \quad (4)$$

Definition 2. Given a licensed spectrum c_p and a continuous period T, the continuous spectrum availability $CSA_p^c(T)$ for one CU_c can be defined as

$$CSA_p^c(T) = \left\{(c, p) \mid ISA_p^c(t) = 1 \text{ from } t_0 \text{ to } t_0 + T\right\}, \quad (5)$$

where t_0 is a given reference time. $CSA_p^c(T) = 1$ means that spectrum c_p is available to CU_c not only at t_0 but also at any time between t_0 and $t_0 + T$. $CSA_p^c(T) = 1$ means that spectrum c_p is not available to CU_c at some time between t_0 and $t_0 + T$. In practice, T could be a slot or some slots, during which CU_c can achieve activation and access to the network. Apparently, we focus more on $CSA_p^c(T)$ because a continuous time period (or a short-time duration) rather than an instantaneous time is more meaningful for OSA or resource allocation in CRN.

Our objective is to give out the spectrum mobility prediction for all the CUs based on the joint theoretical analysis of CUs' mobility and PUs' working activities under the mobile model. Obviously, it is a complicated nonlinear problem which cannot be solved by normal algorithm. In this paper, a new prediction scheme is presented to solve this problem by SVM.

2.2. Support Vector Machine. As a highly competitive learning method, SVM is gaining popularity in many fields based on the statistical learning theory [30–32]. SVM adopts structural risk minimization principle which has been shown superior to empirical risk minimization principle used by traditional neural networks [24]. Moreover, the generalization ability of SVM is strong [33]. SVM is initially used to solve the classification problem. Assume there is l training sample data denoted as

$$D = \left\{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in R^d\right\}_{i=1}^{l}, \quad (6)$$

where $\mathbf{x}_i$ is an input vector containing multiple features. $y_i \in \{-1, +1\}$ is a class indicator. d is the dimension of sample data. Optimal hyper plane is constructed as

$$\mathbf{w} \cdot \mathbf{x}_i + b = 0, \quad (7)$$

where $\mathbf{w}$ are weights and b is offset argument. The samples on $H1$ and $H2$ are support vectors. The according equations are $\mathbf{w} \cdot \mathbf{x}_i + b = +1$ and $\mathbf{w} \cdot \mathbf{x}_i + b = -1$, respectively. So, the classification margin is $2/\|\boldsymbol{\omega}\|$. For our nonlinear problem, the representation of the sample data has to be changed from the original input space to a higher dimensional space which is referred to as the feature space. This quadratic programming (QP) problem can be expressed as

$$\begin{aligned} \min \quad & \Phi(\mathbf{w}, \xi) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{l}\xi_i, \\ \text{s.t.} \quad & y_i\left[(\mathbf{w} \cdot \mathbf{x}_i) + b\right] \geq 1 - \xi_i, \quad i = 1 : l. \end{aligned} \quad (8)$$

ξ_i is relaxation factor and C is cost parameter which is a given value. A Lagrange function is constructed to solve the above constrained optimization problem (8) as follows:

$$\begin{aligned} & Q(\mathbf{w}, b, \alpha, \beta, \xi) \\ & = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{l}\xi_i \\ & \quad - \sum_{i=1}^{l}\alpha_i\left[c_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i\right] - \sum_{i=1}^{l}\beta_i\xi_i, \end{aligned} \quad (9)$$

where α_i and β_i are Lagrange multipliers. In order to get the solution of the original problem, we calculate the partial derivative for different variables as

$$\begin{aligned}\frac{\partial Q(\mathbf{w}, b, \alpha, \beta, \xi)}{\partial \mathbf{w}} &= \mathbf{w} - \sum_{i=1}^{l} \alpha_i c_i \mathbf{x}_i = 0,\\ \frac{\partial Q(\mathbf{w}, b, \alpha, \beta, \xi)}{\partial b} &= \sum_{i=1}^{l} \alpha_i c_i = 0,\\ \frac{\partial Q(\mathbf{w}, b, \alpha, \beta, \xi)}{\partial \xi} &= C - \alpha_i - \beta_i = 0.\end{aligned} \tag{10}$$

Based on (10), the original optimization problem shown in (9) is transformed to a dual optimization problem as

$$\begin{aligned}\max \quad & J(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^{l} \alpha_i \alpha_j c_i c_j K(\mathbf{x}_i, \mathbf{x}_j),\\ \text{s.t.} \quad & 0 \le \alpha_i \le C,\\ & \sum_{i=1}^{l} \alpha_i c_i' = 0, \quad i = 1, 2, \ldots, l.\end{aligned} \tag{11}$$

$K(\mathbf{x}_i, \mathbf{x}_j)$ is a chosen kernel function which will be discussed later in the prediction scheme section. Thus, we can obtain the optimization $\mathbf{w}^*$

$$\mathbf{w}^* = \sum_{i=1}^{l} \alpha_i^* c_i' \mathbf{x}_i, \tag{12}$$

where α_i^* can be solved by (11). And b which does not appear in the dual problem can be calculated through the original constraint. Consider

$$b^* = \frac{1}{2} \left[\max_{c_i=-1} \left(\langle \mathbf{w}^* \cdot \mathbf{x}_i \rangle \right) + \min_{c_i=+1} \left(\langle \mathbf{w}^* \cdot \mathbf{x}_i \rangle \right) \right]. \tag{13}$$

Therefore, the final prediction output expression can be written as

$$D(l') = \operatorname{sign} \left(\sum_{i=1}^{l} \alpha_i c_i K(\mathbf{x}_i, \mathbf{x}_j) + b \right). \tag{14}$$

3. Joint Feature Vector Extraction

The common idea is utilizing SVM via domain information such as location and speed directly. However, it does not make good use of the CUs' mobility characteristic and PUs' working activities information. The traditional methods, thus, result in low prediction accuracy performance, which will be discussed in the simulation part.

In this section, $\mathrm{CSA}_p^c(T)$ as joint feature vector is extracted for SVM through theoretical deduction. Obviously, $\mathrm{CSA}_p^c(T)$ is related to a period T. In this paper, we focus mainly on spectrum allocation interval time T_c which is meaningful for a real CRN. Two situations need to be investigated in order to derive $\mathrm{CSA}_p^c(T_c)$: (1) CU_c is in the coverage area of PU_p at t_0 and (2) CU_c is not in the coverage area of PU_p at t_0, respectively.

For the first situation, let CU_c predict a period T_p during which CU_c can use c_p continuously. And our idea is to derive $\mathrm{CSA}_p^c(T_c)$ through computing $\mathrm{CSA}_p^c(T_p)$. In fact, $\mathrm{CSA}_p^c(T_p)$ includes two main situations: (1) $C_1(T_p)$ representing the situation that CU_c does not move into PU_p' coverage scope between t_0 and $t_0 + T_p$ and (2) $C_2(T_p)$ representing the situation that CU_c moves into PU_p' coverage scope at $t_0 + T_s$ $(0 \le T_s \le T_p)$ while the activity of PU_p is inactive between $t_0 + T_s$ and $t_0 + T_p$. We believe $C_1(T_p)$ and $C_2(T_p)$ dominate the main situations although other complicated situations as small probability events also exist.

$C_1(T_p)$ also contains two parts: (1) P_{out1} representing the situation that the velocity of CU_c does not change from t_0 to $t_0 + T_p$ and (2) P_{out2} representing the other situations. Consider

$$C_1(T_p) = P_{\mathrm{out1}} + P_{\mathrm{out2}}. \tag{15}$$

From (2), P_{out1} can be easily obtained as

$$P_{\mathrm{out1}} = 1 - M_e(T_p) = e^{-\lambda_e T_p}. \tag{16}$$

It is difficult to get the accurate value of P_{out2} because we cannot know the velocity change information (the change in time, speed, and direction) at any time for CU_c. However, the approximate value $E(P_{\mathrm{out2}})$ by estimating P_{out2} can be derived [34]. When $T_p < T_c$, CU_c has to change its movement speed and direction (or any of them) before $t_0 + T_p$, which makes CU_c away from PU_p. Therefore, $\mathrm{CSA}_p^c(T_c)$ can be obtained as

$$\begin{aligned}\mathrm{CSA}_p^c(T_c) &\approx E(P_{\mathrm{out2}})\\ &= \frac{1}{\lambda_e T_p} + \varepsilon_a\\ &\quad + e^{-\lambda_e T_p} \left(\frac{1}{2} p_{\mathrm{away}} \lambda_e T_p - \frac{1}{\lambda_e T_p} - \varepsilon_a - 1 \right),\end{aligned} \tag{17}$$

where p_{away} denotes the probability that CU_c moves away from PU_p after the first velocity (speed and direction) change. In practice, $\varepsilon_a \ge 0$ represents other situations (small probability events except the situations discussed above). For example, CU_c changes its speed three times while CU_c still does not move into PU_p's coverage scope from t_0 to $t_0 + T_c$. ε_a is used to balance the equation and we will discuss it later.

When $T_p \ge T_c$, $\mathrm{CSA}_p^c(T_c)$ for CU_c can be obtained as

$$\begin{aligned}\mathrm{CSA}_p^c(T_c) &\approx C_1(T_c) + E(P_{\mathrm{out2}})\\ &= \frac{1}{\lambda_e T_c} + \varepsilon_a + e^{-\lambda_e T_c} \left(\frac{1}{2} p_{\mathrm{away}} \lambda_e T_c - \frac{1}{\lambda_e T_c} - \varepsilon_a \right).\end{aligned} \tag{18}$$

For the second situation, the movement of CU_c and the working state of PU_p should be investigated simultaneously.

Different from the first situation, we believe that the prediction of the idle state for PU_p is more essential due to the original position of CU_c. Similar to the first situation, let CU_c predict a continuous period T_p^{out} that CU_c will not move out of PU_p's coverage boundary from t_0 to $t_0 + T_p^{\text{out}}$. It is noted that T_p^{out} is different from the T_p mentioned above.

When $T_p^{\text{out}} < T_c$, $\text{CSA}_p^c(T_c)$ for CU_c can be obtained as

$$\text{CSA}_p^c(T_c) = p_{\text{in}} \cdot \int_{t_0}^{t_0+T_p^{\text{out}}} f_{\text{OFF}}\left(t, \beta_p\right) dt + \varepsilon_{\text{in}}. \tag{19}$$

Similar to ε_a above, ε_{in} is used to denote all the other small probability events. $\int_{t_0}^{t_0+T_p^{\text{out}}} f_{\text{OFF}}(t)dt$ represents the idle probability of c_p between t_0 and $t_0 + T_p^{\text{out}}$. p_{in} represents the probability that CU_c moves out of PU_p' coverage boundary before $t_0 + T_p$. p_{in} consists of two parts. P_{in1} denotes the situation that the velocity of CU_c remains unchanged from t_0 to $t_0 + T_p$. And P_{in2} denotes the other situations. P_{in} can be easily obtained as

$$P_{\text{in}} = \frac{1}{\lambda_e T_p^{\text{out}}} + \varepsilon_b + e^{-\lambda_e T_p}\left(\frac{1}{2} p_b \lambda_e T_p^{\text{out}} - \frac{1}{\lambda_e T_p^{\text{out}}} - \varepsilon_b\right), \tag{20}$$

where $\varepsilon_b \geq 0$ tries to represent all the other situations. p_b represents the probability that CU_c moves away from PU_p after the first change in velocity. According to (1), (19), and (20), $\text{CSA}_p^c(T_c)$ for CU_c can be obtained as

$$\begin{aligned}
&\text{CSA}_p^c(T_c) \\
&= \left(\frac{1}{\lambda_e T_p^{\text{out}}} + \varepsilon_b + e^{-\lambda_e T_p^{\text{out}}}\left(\frac{1}{2} p_b \lambda_e T_p^{\text{out}} - \frac{1}{\lambda_e T_p^{\text{out}}} - \varepsilon_b\right)\right) \\
&\quad \times \left(e^{-\mu_p t_0} - e^{-1/\beta_p \cdot (t_0 + T_p^{\text{out}})}\right) + \varepsilon_{\text{in}}.
\end{aligned} \tag{21}$$

When $T_p^{\text{out}} \geq T_c$, $\text{CSA}_p^c(T_c)$ is mainly determined by PU_p's working activity. Thus, $\text{CSA}_p^c(T_c)$ for CU_c can be obtained as

$$\begin{aligned}
\text{CSA}_p^c(T_c) &= \int_{t_0}^{t_0+T_c} f_{\text{OFF}}\left(t, \beta_p\right) dt + \varepsilon_c \\
&= e^{-1/\beta_p \cdot t_0} - e^{-1/\beta_p \cdot (t_0 + T_c)} + \varepsilon_c.
\end{aligned} \tag{22}$$

ε_c denotes all the other spectrum availability situations. Thus, we obtain joint feature vector sets $\mathbf{S}_i^c = \{\text{CSA}_p^c(T_c)_i\}_{i=1}^l$ according to different situations based on (17), (18), (21), and (22). Moreover, we can get the ultimate prediction expression according to (14)

$$\begin{aligned}
&D\left(l'\right) = \text{sign}\left(\sum_{i=1}^{m} \alpha_i c_i' K\left(\mathbf{S}_i^c, \mathbf{x}_j\right) + b\right), \\
&\text{s.t.} \quad \mathbf{S}_i^c = \left\{\text{CSA}_p^c(T_c)_i\right\}_{i=1}^l.
\end{aligned} \tag{23}$$

4. Spectrum Mobility Prediction Scheme

In this section, a new SVM-based spectrum mobility prediction scheme is proposed based on the analysis and deduction above. The main steps of the proposed prediction scheme are as follows.

Step 1 (CRN initialization). Initialize the original locations of PUs and CUs randomly in the two-dimensional deployment area. The coordinates of PUs are not changed once generated. And PUs' initial working states are stochastic. Initialize the original speed, direction, and epoch lengths for each CU_c. Assume the maximum velocity for CUs is $v_{\max}$. The beginning time of the system is set to t_0. The parameters C, ξ_i of SVM are initialized. Simulations are based on many times to make sure of the accuracy of the result. In addition, set T_c, α_p, β_p, $\varepsilon_{\text{away}}$, ε_a, ε_b, ε_c, p_{away}, $p_b\lambda_e$, R_p, and r_c.

Step 2 (operate the mobile CRN model). PUs' working states obey an exponential on/off model. The PDF satisfies (1). And the mobility of CUs follows the mobile model mentioned in Section 2.1. The mobility epoch lengths are independently exponentially distributed with mean $1/\lambda_e$. It is noted that wrap-around technique is adopted during simulation in order to make the total number of SUs unchanged in the simulation area.

Step 3 (calculate the joint feature vectors $\mathbf{S}_i^c$ for SVM). Here, there are three situations to be investigated. Firstly, CU_c does not move into the coverage of PU_p at t_0 granted that CU_c moves with $v_{\max}$ towards PU_p. Secondly, CU_c is out of the coverage of PU_p at t_0, but CU_c may move into the coverage of PU_p between t_0 and $t_0 + T_c$. Finally, CU_c is in the coverage of PU_p at t_0. The detail calculation steps for $\mathbf{S}_i^c$ are described in Algorithm 1.

Step 4 (execute prediction by SVM). Firstly, a SVM prediction model is generated according to the history input vectors $\mathbf{S}_i^c$ from Step 3. Secondly, put the data to be predicted into the generated SVM model. Then, compute the prediction results and record the results. Here, we adopt the RBF kernel as mapping function for SVM in simulation. Because the RBF kernel function tends to obtain more robust results than other kernels and can reduce numerical difficulties, the RBF kernel function can be defined as

$$K\left(\mathbf{x}, \mathbf{y}\right) = \exp\left(-\gamma \left\|\mathbf{x} - \mathbf{y}\right\|^2\right), \quad \gamma > 0. \tag{24}$$

Step 5 (the system resets). Execute Steps 1 to 4 until simulation numbers are satisfied for testing. Then, the operation stops. Compute the prediction performance: the prediction accuracy rate p_{accuracy} and the miss detection probability p_{miss}. p_{accuracy} is defined as

$$p_{\text{accuracy}} = \frac{\left|\left\{i \mid y_i' \cdot f\left(\mathbf{x}_i'\right) > 0\right\}\right|}{l'} \times 100\%, \tag{25}$$

where $\{\mathbf{x}_i'\}_1^{l'}$ are testing data which are to be predicted. And $y_i' \in \{-1, +1\}$ are true labels for testing data. l' is

Calculation steps of $\mathbf{S_i^c}$
if $D_{p,c}(t_0) \geq R_p + r_c + v_{\max} \cdot T_c$
  Spectrum c_p is available for CU_c.
else if $R_p + r_c \leq D_{p,c}(t_0) < R_p + r_c + v_{\max} \cdot T_c$
  if $T_p < T_c$
    Calculate $\mathrm{CSA}_p^c(T_c)$ by (17).
  else
    Calculate $\mathrm{CSA}_p^c(T_c)$ by (18).
  end
else if $0 < D_{p,c}(t_0) < R_p + r_c$
  if $T_p^{\mathrm{out}} < T_c$
    Calculate $\mathrm{CSA}_p^c(T_c)$ by (21).
  else
    Calculate $\mathrm{CSA}_p^c(T_c)$ by (22).
  end
end
The vectors $\mathbf{S_i^c}$ are constructed for SVM.

Algorithm 1: Joint feature vectors extraction for CU_c.

the total number of testing data. $f(\mathbf{x}_i') \in \{-1, +1\}$ are the predicted decision values. Here, $\{-1, +1\}$ represents the busy/idle working activity for a given PU. $|\cdot|$ represents the element numbers for a given set. Actually, p_{accuracy} reflects the accuracy degree of spectrum prediction mechanism. The higher the p_{accuracy} is, the better the prediction effect is.

In addition, the miss prediction rate p_{miss} (the rate that the spectrum is predicted to be idle while it is actually busy) is investigated for the proposed prediction mechanism. Because p_{miss} can reflect the actual interference to the PUs to some extent. The smaller the p_{miss} is, the better the prediction mechanism is. p_{miss} can be defined as

$$p_{\mathrm{miss}} = \frac{\left|\left\{i \mid y_i' \cdot f\left(\mathbf{x}_i'\right) < 0, f\left(\mathbf{x}_i'\right) = 1\right\}\right|}{l'} \times 100\%. \quad (26)$$

Note that $p_{\mathrm{accuracy}} + p_{\mathrm{miss}} \leq 100\%$.

5. Simulation Results and Analysis

In this section, experimental results of the prediction performances for our proposed scheme are investigated. Simulation parameters are shown in Table 1. We compare the proposed prediction mechanism with the traditional prediction schemes (SVM with initial location coordinates of CUs and SVM with initial location coordinates and speed of CUs) to evaluate the prediction performances under different parameters.

Note that we assume $\varepsilon_a = \varepsilon_b = \varepsilon_c = \varepsilon_{\mathrm{in}} = \varepsilon_{\mathrm{away}}$ during simulation for simplicity, because they are very small positive values that are set to balance the according equations. In addition, the total testing number is set to 1000 in order to avoid randomness during simulation.

Figure 3 shows p_{accuracy} among different algorithms versus training node number. The proposed SVM-SMP converges at about 40 training data faster than SVM-location algorithm (SVM-LA) with a convergent result at about 100 training data. p_{accuracy} of SVM-location-speed algorithm (SVM-LSA) is much worse than the other comparison algorithms which shows that the initial speed parameters have a bad effect on the prediction performance. It is caused by the time-varying characteristic of CUs' velocity (speed or direction).

As shown in Figure 4, p_{accuracy} decreases with the increasing of T_c for the three different algorithms. p_{accuracy} of the proposed SVM-SMP is better than SVM-LA when T_c is relatively small ($1 \leq T_c \leq 4.5$). However, p_{accuracy} of SVM-SMP is worse than SVM-LA when T_c is big enough ($T_c > 4.5$). It is because the proposed prediction scheme is based on the short-time prediction idea according to the feature vector

Table 1: Simulation parameters.

Parameter	Value
Total simulation number for testing	1000
Simulation area	5000 m × 5000 m
R_p	1000 m
r_c	500 m
Kernel function	RBF kernel
C	10
ξ_i	0.01
T_c	1 s~15 s
α_p	1/3 s
β_p	1/3 s
$p_a = p_b = p_{\mathrm{away}}$	0.5
$\varepsilon_a = \varepsilon_b = \varepsilon_c = \varepsilon_{\mathrm{in}} = \varepsilon_{\mathrm{away}}$	0~0.2
$1/\lambda_e$	3 s
CUs' maximum velocity $v_{\max}$	0 m/s~50 m/s
Total number of PUs	2
Total number of training CUs	20~240
Total number of testing CUs	1000

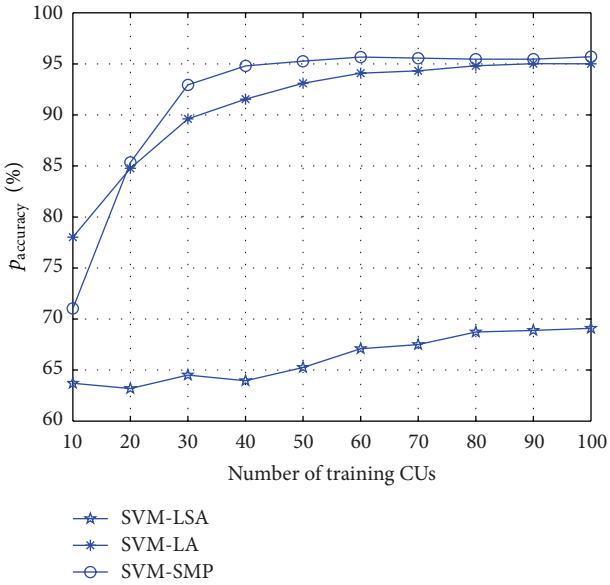


Figure 3: Prediction accuracy rate $p_{accuracy}$ for CUs versus training node number.

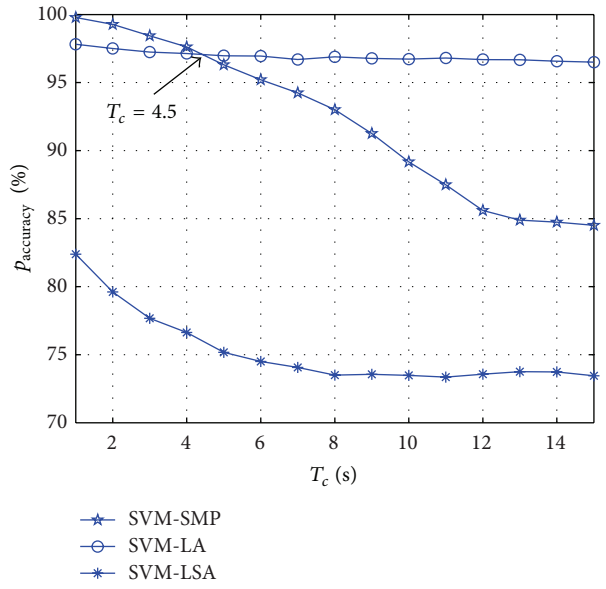


Figure 4: Prediction accuracy rate $p_{accuracy}$ for CUs versus T_c.

extraction analysis in (17), (18), (21), and (22). Moreover, wrap-around technique makes CUs stay at their original positions with big probability at the simulation boundary. Thus, SVM-LA decreases not that fast with the increasing of T_c. And the SVM-SMP works well when the prediction time is not long and vice versa. Note that the short-time prediction performance is mainly focused on in this paper because CR itself should achieve communication in a very short time.

As illustrated in Figure 5, p_{miss} is studied versus T_c for different algorithms. p_{miss} of SVM-SMP is nearly equal to 0 when T_c is small ($1 \leq T_c \leq 5$), which is better than SVM-LA. However, p_{miss} of SVM-SMP increases very fast when

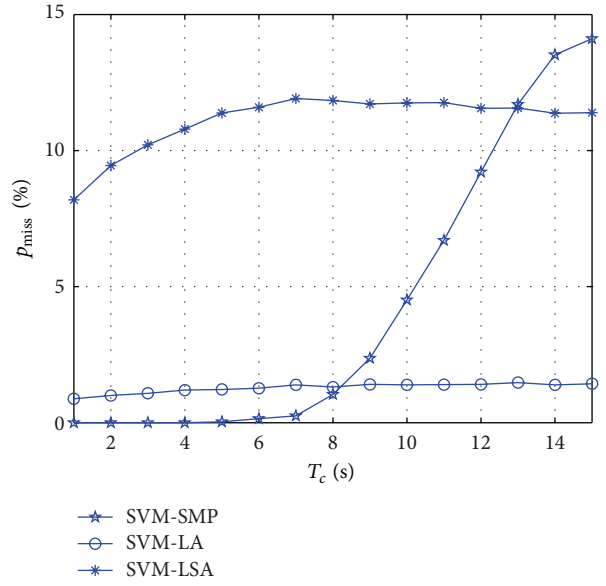


Figure 5: Miss prediction rate p_{miss} for CUs versus T_c.

T_c is big enough. Figure 5 shows the good short-time miss prediction rate characteristics of the new algorithm which is very essential to CR.

In Figures 6 and 7, $p_{accuracy}$ and p_{miss} versus v_{max} are investigated between SVM-SMP and SVM-LA. From Figure 6, $p_{accuracy}$ of SVM-SMP is better than that of SVM-LA when $1 \leq T_c \leq 4$ s. However, $p_{accuracy}$ of SVM-SMP is worse than that of SVM-LA when $T_c = 5$ s and $v_{max} > 42$ m/s. It shows that the proposed SVM-SMP lose the advantages when the prediction time and speed are too big simultaneously. As shown in Figure 7, p_{miss} of SVM-SMP is less than 0.1%. Comparatively, p_{miss} of SVM-LA is approximately 1%. Thus, SVM-SMP shows good p_{miss} performance with v_{max} changing.

In Figures 8 and 9, we investigate $p_{accuracy}$ and p_{miss} versus λ between SVM-SMP and SVM-LA. From Figure 8, $p_{accuracy}$ of SVM-SMP decreases obviously with the increasing of λ when v_{max} is big (such as 50 m/s). However, $p_{accuracy}$ of SVM-SMP does not change very obviously with the increasing of λ when v_{max} is small (such as 10 m/s). It is due to the fact that the bigger the λ is, the stronger the irregular movements of CUs are. Thus, it is difficult for the prediction when CUs are moving with high speed and strong irregular movements. As shown in Figure 9, p_{miss} of SVM-SMP is much better than that of SVM-LA when v_{max} is small (such as 10 m/s, 30 m/s). However, p_{miss} of SVM-SMP is worse than that of SVM-LA when $v_{max} = 50$ m/s and $\lambda > 13$, which validates the performance degradation of the prediction performance again when v_{max} is relatively big with strong irregular movements. It is because SVM-SMP is based on the assumption of weak irregular movements for CUs.

In Figures 10 and 11, $p_{accuracy}$ and p_{miss} versus β_p are investigated for SVM-SMP and SVM-LA. In Figure 10, $p_{accuracy}$ of SVM-SMP is obviously better than $p_{accuracy}$ of SVM-LA when the prediction time is short such as 1 s and 3 s. However, $p_{accuracy}$ of SVM-SMP is worse than $p_{accuracy}$ of

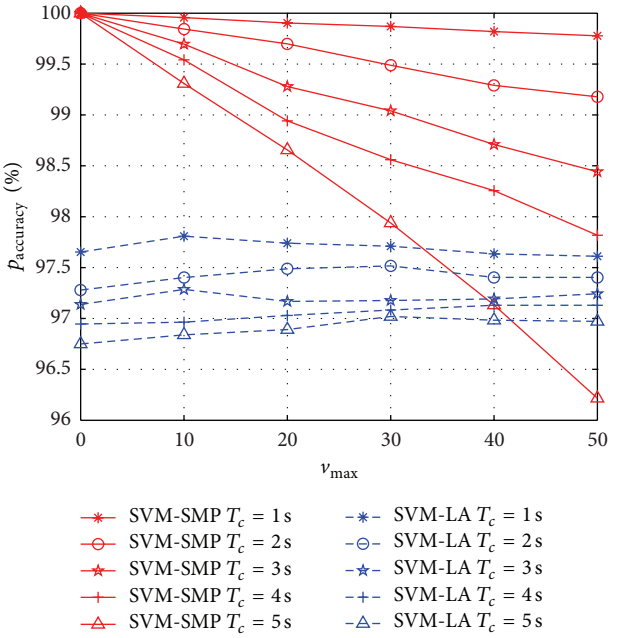


Figure 6: Prediction accuracy rate $p_{accuracy}$ for CUs versus v_{max}.

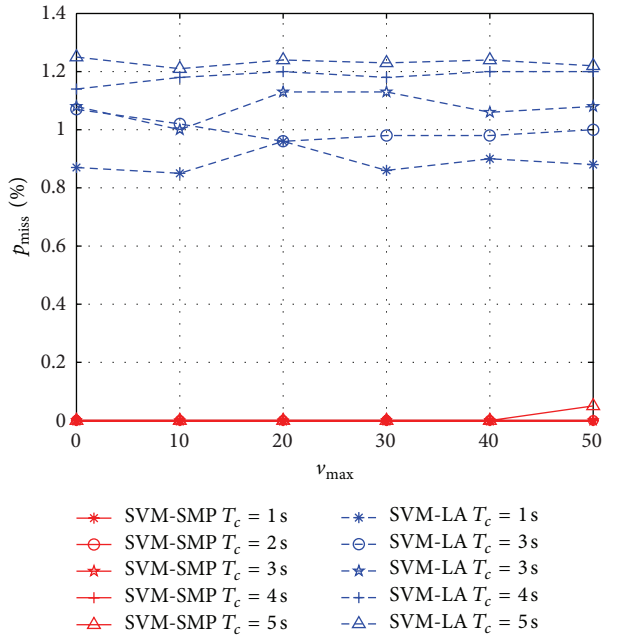


Figure 7: Miss prediction rate p_{miss} for CUs versus v_{max}.

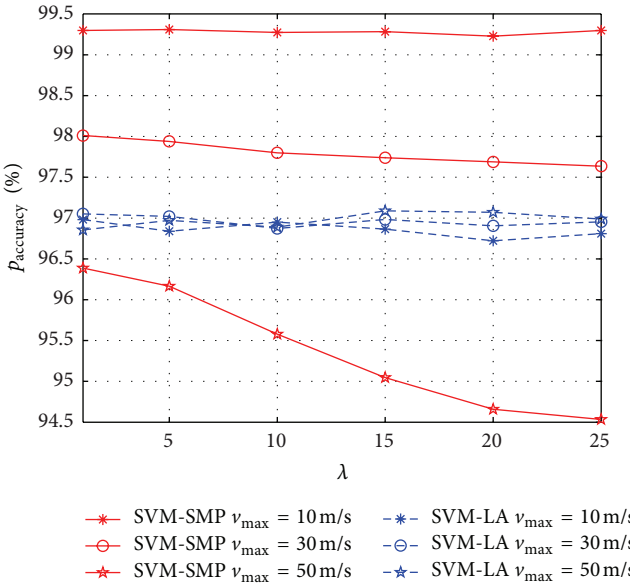


Figure 8: Prediction accuracy rate $p_{accuracy}$ for CUs versus λ.

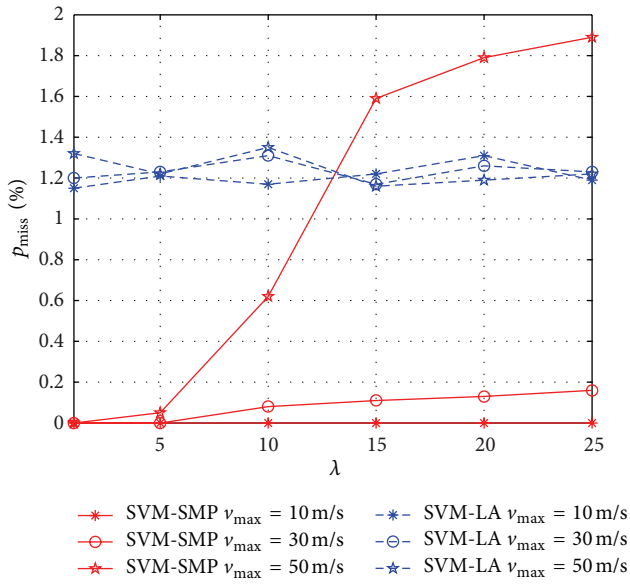


Figure 9: Miss prediction rate p_{miss} for CUs versus λ.

SVM-LA when the prediction duration time is $T_c = 5$ s and the mean idle time is $\beta_p < 4$. In Figure 11, p_{miss} of SVM-SMP is nearly equal to 0, which is much better than SVM-LA. In addition, the prediction performance ($p_{accuracy}$ and p_{miss}) does not improve significantly for the two algorithms with the increasing of β_p when β_p is big enough according to Figures 10 and 11.

Figure 12 shows the impact of ε on the prediction accuracy rate $p_{accuracy}$ for SVM-SMP. When $\lambda = 1$, the maximum of $p_{accuracy}$ occurs at about $\varepsilon = 0$ which is relatively small. With the increasing of λ, the maximum value position of $p_{accuracy}$ moves to the right. When $\lambda = 10$, the maximum of $p_{accuracy}$ occurs at about $\varepsilon = 0.1$. It is because the small probability events happen more times when SUs' randomness movements are strong (λ is big). The bigger the λ is, the bigger the maximum value position of $p_{accuracy}$ occurs for ε. Therefore, we can obtain better prediction performance by adjusting ε for randomness movements of different strength.

As shown in Figure 13, miss prediction rate p_{miss} is studied versus ε for SVM-SMP. When $\lambda = 5$ and $\lambda = 10$, the minimum of p_{miss} occurs at about $\varepsilon = 0.07$ and $\varepsilon = 0.1$, respectively. It shows that different optimal ε corresponds to

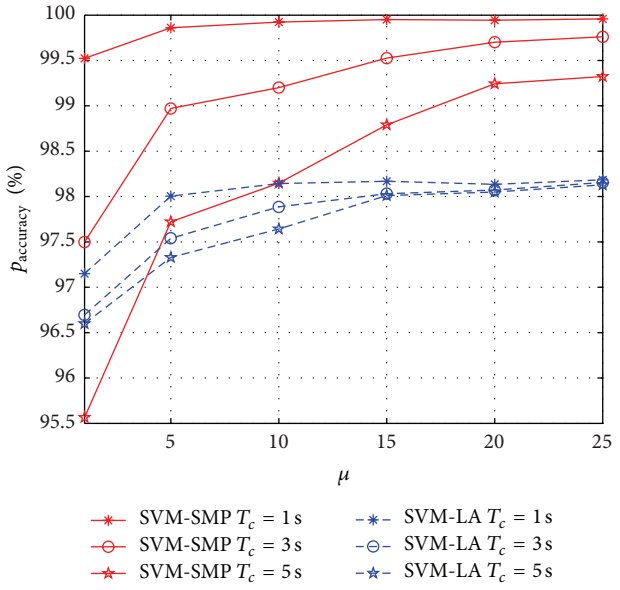


Figure 10: Prediction accuracy rate $p_{accuracy}$ for CUs versus μ.

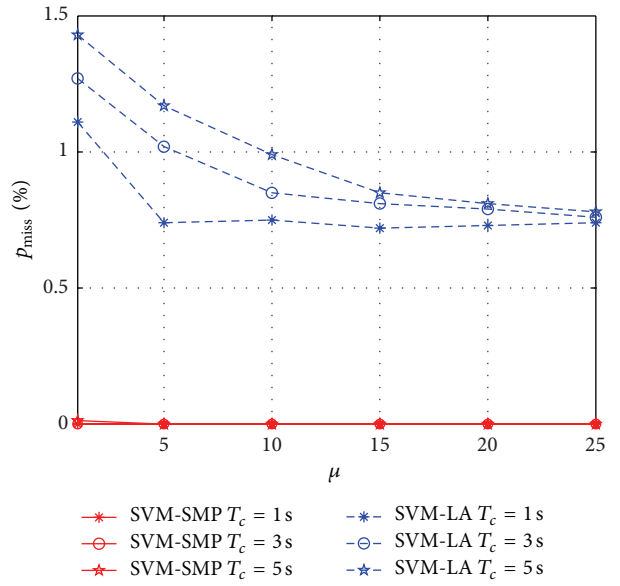


Figure 11: Miss prediction rate p_{miss} for CUs versus μ.

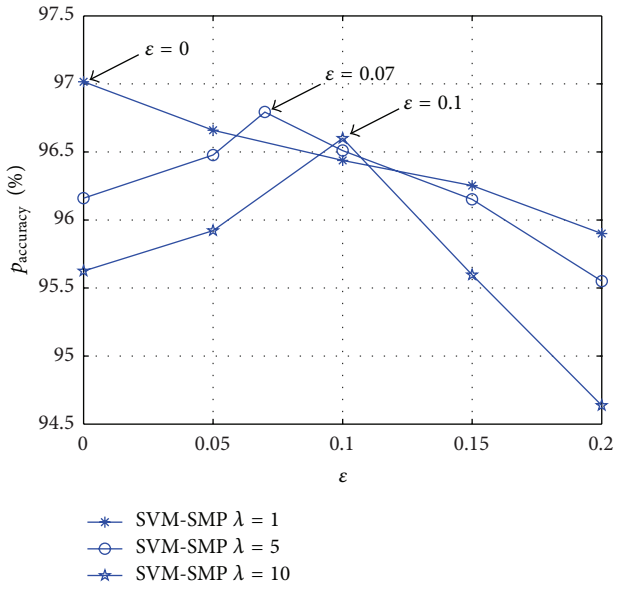


Figure 12: Prediction accuracy rate $p_{accuracy}$ for CUs versus ε.

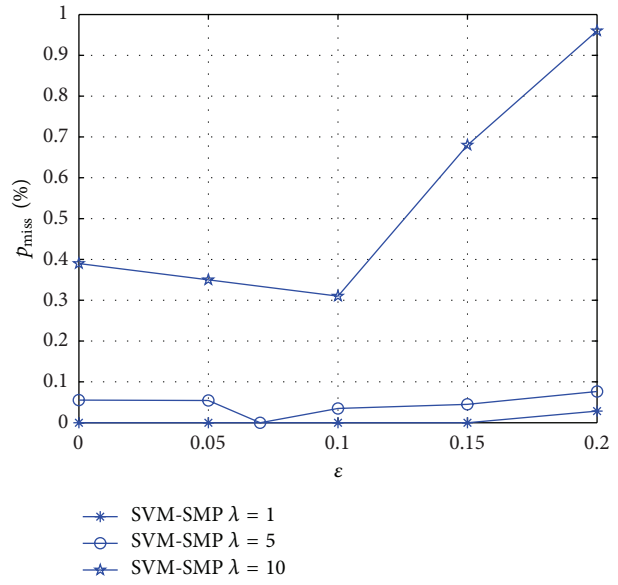


Figure 13: Miss prediction rate p_{miss} for CUs versus ε.

SUs' randomness movements of different strength for p_{miss}. The simulation results further confirm the impact of ε on the prediction accuracy rate in Figure 12.

6. Conclusions

In this paper, a new spectrum mobility prediction algorithm is proposed in mobile CRNs. SVM theory is adopted to improve the spectrum mobility prediction performance, which takes into account time- and space-varying characteristics together. Moreover, new extracted feature vectors based on the theoretical analysis are input into SVM. Simulation results confirm that the convergence speed of our SVM-SMP is faster than SVM-LA and SVM-LSA. Meanwhile, SVM-SMP shows better short-time prediction performance than SVM-LA and SVM-LSA, which is essential to real mobile CRNs. In addition, the prediction performance degradation caused by SUs' high speed and strong randomness movements can be made up by choosing the proper parameters.

As known to us, how to choose the best parameters (C and ξ_i) quickly is still an open problem in SVM. We will further analyze the impact of ε on prediction performance. It leaves us to investigate in the future.

Conflict of Interests

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

Acknowledgments

This research is supported by National Natural Science Foundation and Civil Aviation Administration of China (Grant nos. 61101122 and 61071104), National High Technology Research and Development Program of China (863 Program) (Grant no. 2012AA120802), and Postdoctoral Science-Research Development Foundation of Heilongjiang Province (Grant no. LBH-Q12080).

References

[1] F. Kader, Asaduzzaman, and M. Hoque, "Hybrid spectrum sharing with cooperative secondary user selection in cognitive radio networks," *KSII Transactions on Internet and Information Systems*, vol. 23, no. 2, pp. 2081–2100, 2013.

[2] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[3] S. Sun, Y. Ju, and Y. Yamao, "Overlay cognitive radio OFDM system for 4G cellular networks," *IEEE Wireless Communications*, vol. 20, no. 2, pp. 68–73, 2013.

[4] C. X. Wang, F. Haider, and X. Gao, "Cellular architecture and key technologies for 5G wireless communication networks," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 122–130, 2014.

[5] Q. Zhao and A. Swami, "A decision-theoretic framework for opportunistic spectrum access," *IEEE Wireless Communications*, vol. 14, no. 4, pp. 14–20, 2007.

[6] Y. Zou, Y. Yao, and B. Zheng, "A cooperative sensing based cognitive relay transmission scheme without a dedicated sensing relay channel in cognitive radio networks," *IEEE Transactions on Signal Processing*, vol. 59, no. 2, pp. 854–858, 2011.

[7] K. B. Letaief and W. Zhang, "Cooperative communications for cognitive radio networks," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 878–893, 2009.

[8] I. F. Akyildiz, W. Lee, M. C. Vuran, and S. Mohanty, "A survey on spectrum management in cognitive radio networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 40–48, 2008.

[9] B. Wang, Z. Ji, K. J. R. Liu, and T. C. Clancy, "Primary-prioritized Markov approach for dynamic spectrum allocation," *IEEE Transactions on Wireless Communications*, vol. 8, no. 4, pp. 1854–1865, 2009.

[10] V. Asghari and S. Aissa, "Adaptive rate and power transmission in spectrum-sharing systems," *IEEE Transactions on Wireless Communications*, vol. 9, no. 10, pp. 3272–3280, 2010.

[11] E. Jung and X. Liu, "Opportunistic spectrum access in multiple-primary-user environments under the packet collision constraint," *IEEE/ACM Transactions on Networking*, vol. 20, no. 2, pp. 501–514, 2012.

[12] G. Yuan, R. C. Grammenos, Y. Yang, and W. Wang, "Performance analysis of selective opportunistic spectrum access with traffic prediction," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 1949–1959, 2010.

[13] S. Yin, D. Chen, Q. Zhang, and S. Li, "Prediction-based throughput optimization for dynamic spectrum access," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 3, pp. 1284–1289, 2011.

[14] X. Li and S. A. Zekavat, "Traffic pattern prediction and performance investigation for cognitive radio systems," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '08)*, pp. 894–899, March 2008.

[15] X. Huang, G. Wang, F. Hu, and S. Kumar, "Stability-capacity-adaptive routing for high-mobility multihop cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 6, pp. 2714–2729, 2011.

[16] Y. Song and J. Xie, "Optimal power control for concurrent transmissions of location-aware mobile cognitive radio ad hoc networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '09)*, pp. 1–6, December 2009.

[17] A. C. Talay and D. T. Altilar, "United nodes: cluster-based routing protocol for mobile cognitive radio networks," *IET Communications*, vol. 5, no. 15, pp. 2097–2105, 2011.

[18] L. Zhang, K. Zeng, and P. Mohapatra, "Opportunistic spectrum scheduling for mobile cognitive radio networks in white space," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '11)*, pp. 844–849, March 2011.

[19] R. W. Thomas, L. A. DaSilva, and A. B. MacKenzie, "Cognitive networks," in *Proceedings of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, pp. 352–360, November 2005.

[20] H. Jiang, L. Lai, R. Fan, and H. V. Poor, "Optimal selection of channel sensing order in cognitive radio," *IEEE Transactions on Wireless Communications*, vol. 8, no. 1, pp. 297–307, 2009.

[21] Z. Li, F. R. Yu, and M. Huang, "A distributed consensus-based cooperative spectrum-sensing scheme in cognitive radios," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 383–393, 2010.

[22] T. Yücek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys and Tutorials*, vol. 11, no. 1, pp. 116–130, 2009.

[23] V. K. Tumuluru, P. Wang, and D. Niyato, "A neural network based spectrum prediction scheme for cognitive radio," in *Proceedings of the IEEE International Conference on Communications (ICC '10)*, pp. 1–5, May 2010.

[24] J. Guo, H. Ji, Y. Li, and X. Li, "A novel spectrum handoff management scheme based on SVM in cognitive radio networks," in *Proceedings of the 6th International ICST Conference on Communications and Networking in China (CHINACOM '11)*, pp. 645–649, August 2011.

[25] S. Yarkan and H. Arslan, "Binary time series approach to spectrum prediction for cognitive radio," in *Proceedings of the IEEE 66th Vehicular Technology Conference (VTC '07)*, pp. 1563–1567, October 2007.

[26] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: a POMDP framework," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 3, pp. 589–600, 2007.

[27] W. Lee and I. F. Akyildiz, "Optimal spectrum sensing framework for cognitive radio networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 10, pp. 3845–3857, 2008.

[28] G. Ning and P. Nintanavongsa, "Time prediction based spectrum usage detection in centralized cognitive radio networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '12)*, pp. 300–305, 2012.

[29] A. B. McDonald and T. F. Znati, "Mobility-based framework for adaptive clustering in wireless ad hoc networks," *IEEE Journal*

on Selected Areas in Communications, vol. 17, no. 8, pp. 1466–1487, 1999.

[30] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, USA, 1995.

[31] N. Cristianini and J. S. Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge University Press, Cambridge, UK, 2000.

[32] Y. Liang, M. L. Reyes, and J. D. Lee, "Real-time detection of driver cognitive distraction using support vector machines," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 340–350, 2007.

[33] C. Wu, Q. Yu, and K. Yi, "Least-squares support vector machine-based learning and decision making in cognitive radios," *IET Communications*, vol. 6, no. 17, pp. 2855–2863, 2012.

[34] S. Jiang, D. He, and J. Rao, "A prediction-based link availability estimation for routing metrics in MANETs," *IEEE/ACM Transactions on Networking*, vol. 13, no. 6, pp. 1302–1311, 2005.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 560234, 11 pages
http://dx.doi.org/10.1155/2014/560234

Research Article

Robust H_∞ Filtering for a Class of Complex Networks with Stochastic Packet Dropouts and Time Delays

Jie Zhang,[1] Ming Lyu,[2] Hamid Reza Karimi,[3] Pengfei Guo,[1] and Yuming Bo[1]

[1] *School of Automation, Nanjing University of Science & Technology, Nanjing 210094, China*
[2] *Information Overall Department, North Information Control Group Co., Ltd., Nanjing 211153, China*
[3] *Department of Engineering, Faculty of Engineering and Science, University of Agder, 4898 Grimstad, Norway*

Correspondence should be addressed to Jie Zhang; zhangjie.njust@gmail.com

Received 27 December 2013; Accepted 6 March 2014; Published 27 March 2014

Academic Editors: J. Bajo and Z. Wu

The robust H_∞ filtering problem is investigated for a class of complex network systems which has stochastic packet dropouts and time delays, combined with disturbance inputs. The packet dropout phenomenon occurs in a random way and the occurrence probability for each measurement output node is governed by an individual random variable. Besides, the time delay phenomenon is assumed to occur in a nonlinear vector-valued function. We aim to design a filter such that the estimation error converges to zero exponentially in the mean square, while the disturbance rejection attenuation is constrained to a given level by means of the H_∞ performance index. By constructing the proper Lyapunov-Krasovskii functional, we acquire sufficient conditions to guarantee the stability of the state detection observer for the discrete systems, and the observer gain is also derived by solving linear matrix inequalities. Finally, an illustrative example is provided to show the usefulness and effectiveness of the proposed design method.

1. Introduction

Over the past decades, the H_∞ filtering problem has drawn particular attention, since H_∞ filters are insensitive to the exact knowledge of the statistics of the noise signals. Up to now, a great deal of effort has been devoted to the design issues of various kinds of filters, for example, the Kalman filters [1–3] and H_∞ filters [4–9].

In real-world applications, the measurements may contain missing measurements (or incomplete observations) due to various reasons such as high maneuverability of the tracked targets, sensor temporal failures or network congestion. In the past few years, the filtering problem with missing measurements has received much attention [10–17]. In [10], a model of multiple missing measurements has been presented by using a diagonal matrix to account for the different missing probabilities for individual sensors. The finite-horizon robust filtering problem has been considered in [11] for discrete-time stochastic systems with probabilistic missing measurements subject to norm-bounded parameter uncertainties. A Markovian jumping process has been employed in [12] to reflect the measurement missing problem. Moreover, the optimal filter design problem has been tackled in [13] for systems with multiple packet dropouts by solving a recursive difference equation (RDE).

On the other hand, the complex networks have been gaining increasing research attention from all fields of the basic science and the technological practice. They have applications in many real-world systems such as the Internet, World Wide Web, food webs, electric power grids, cellular and metabolic networks, scientific citation networks, and social networks [18–25]. Due to randomly occurring incomplete phenomenon which occurs in the signal transfer within complex networks, there may be time delays and packet dropouts [26–33]. For instance, over a finite horizon, the synchronization and state estimation problems for an array of coupled discrete time-varying stochastic complex networks have been studied based on the recursive linear matrix inequalities (RLMIs) approach [26]. In [29], one of the first few attempts has been made to address the synchronization problem for stochastic discrete-time complex networks with time delays. Furthermore, in [31], a new array of coupled

delayed complex networks with stochastic nonlinearities, multiple stochastic disturbances, and mixed time delays in the discrete-time domain has been investigated, and the synchronization stability criteria have been derived by utilizing a novel matrix functional, the properties of the Kronecker product, the free-weighting matrix method, and the stochastic techniques.

Summarizing the above discussion, it should be pointed out that, up to now, the general filter results for complex networks with randomly occurring incomplete information have been very few, especially when the networks exhibit both stochastic natures and disturbance inputs. In this paper, we make an attempt to investigate the problems of the robust H_∞ filtering for a class of complex systems with stochastic packet dropouts, time delays, and disturbance inputs. By constructing the proper Lyapunov-Krasovskii functional, we can get sufficient conditions, such that the filter error is exponentially stable in mean-square sense, and acquire gain of the designed observer.

The rest of the paper is organized as follows. In Section 2, the problem of complex networks is formulated and some useful lemmas are introduced. In Section 3, some sufficient conditions are established to make sure the robustly exponential stability of the filtering error dynamics. Besides, the gain of observer is also designed by LMI. An illustrated example is given in Section 4 to demonstrate the effectiveness of the proposed method. Finally, we give our conclusions in Section 5.

Notation. The notation used here is fairly standard except where otherwise stated. $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ denote, respectively, the n dimensional Euclidean space and the set of all $n \times m$ real matrices. I denotes the identity matrix of compatible dimension. The notation $X \geq Y$ (resp., $X > Y$), where X and Y are symmetric matrices, means that $X - Y$ is positive semidefinite (resp., positive definite). A^T represents the transpose of A. $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ denote the maximum and minimum eigenvalue of A, respectively. Sym$\{A\}$ denotes the symmetric matrix $A + A^T$. $\mathbb{E}\{x\}$ stands for the expectation of the stochastic variable x. $\|x\|$ describes the Euclidean norm of a vector x. diag$\{F_1, F_2, \ldots\}$ stands for a block-diagonal matrix whose diagonal blocks are given by $F_1, F_2, \ldots$. The symbol $*$ in a matrix means that the corresponding term of the matrix can be obtained by symmetric property. The symbol $\otimes$ denotes the Kronecker product. In symmetric block matrices, the symbol $*$ is used as an ellipsis for terms induced by symmetry.

2. Problem Formulation

Consider the following discrete-time complex system with time delays and disturbance:

$$\begin{aligned} x_i(k+1) &= f(x_i(k)) + g(x_i(k-d(k))) \\ &\quad + \sum_{j=1}^{N} w_{ij}\Gamma x_j(k) + D_{1i}v_1(k) + h(x_i(k))\omega(k), \\ z_i(k) &= Mx_i(k), \\ x_i(j) &= \varphi_i(j), \quad j = -\overline{d}_M, -\overline{d}_M + 1, \ldots, 0; \\ &\qquad i = 1, 2, \ldots, N, \end{aligned} \tag{1}$$

where $x_i(k) \in \mathbb{R}^n$ is the state vector of the ith node, $z_i(k) \in \mathbb{R}^r$ is the output of the ith node, $d(k)$ denotes time-varying delay, $f(\cdot)$ and $g(\cdot)$ are nonlinear vector-valued functions satisfying certain conditions given later, $v_1(k)$ is the disturbance input belonging to $l_2([0,\infty);\mathbb{R}^q)$, $\omega(k)$ is a zero mean Gaussian white noise sequence, and $h(\cdot)$ is the continuous function quantifying the noise intensity. $\Gamma = \text{diag}\{r_1, r_2, \ldots, r_n\}$ is the matrix linking the jth state variable if $r_j \neq 0$, and $W = (w_{ij})_{N\times N}$ is the coupled configuration matrix of the network with $w_{ij} > 0$ $(i \neq j)$ but not all zero. As usual, the coupling configuration matrix W is symmetric (i.e., $W = W^T$) and satisfies

$$\sum_{j=1}^{N} w_{ij} = \sum_{j=1}^{N} w_{ji} = 0, \quad i = 1, 2, \ldots, N. \tag{2}$$

D_{1i}, F_i, and M are constant matrices with appropriate dimensions, and $\varphi_i(j)$ is a given initial condition sequence.

For the system shown in (2), we make the following assumptions throughout the paper.

Assumption 1. The variable $\omega(k)$ is a scalar Wiener process (Brownian motion) satisfying

$$\begin{aligned} \mathbb{E}\{\omega(k)\} = 0, \qquad \mathbb{E}\left\{\omega^2(k)\right\} = 1, \\ \mathbb{E}\{\omega(k)\omega(j)\} = 0 \quad (k \neq j). \end{aligned} \tag{3}$$

Assumption 2. The variable $d(k)$ denotes the time-varying delay satisfying

$$0 < \overline{d}_m \leq d(k) \leq \overline{d}_M, \tag{4}$$

where $\overline{d}_m$ and $\overline{d}_M$ are constant positive integers representing the lower and upper bounds on the communication delay, respectively.

Assumption 3. $f(\cdot)$ and $g(\cdot)$ are the nonlinear disturbance which satisfies the following sector-bounded conditions:

$$\begin{aligned} &\left[f(x) - f(y) - \phi_1^f(x-y)\right]^T \\ &\qquad \times \left[f(x) - f(y) - \phi_2^f(x-y)\right] \leq 0, \\ &\left[g(x) - g(y) - \phi_1^g(x-y)\right]^T \\ &\qquad \times \left[g(x) - g(y) - \phi_2^g(x-y)\right] \leq 0, \end{aligned} \tag{5}$$

for all $x, y \in \mathbb{R}^n$, where ϕ_1^f, ϕ_2^f, ϕ_1^g, and ϕ_2^g are real matrices of appropriate dimensions and $f(0) = 0$, $g(0) = 0$.

Assumption 4. The continuous function $h(x_i(k))$ satisfies

$$h^T(x_i(k))\,h(x_i(k)) \leq \varrho x_i^T(k)\,x_i(k), \tag{6}$$

where $\varrho > 0$ is known constant scalars.

In this paper, we assume that an unreliable network medium is present between the physical plant and the state detection filter, and this means that the output data is subject to randomly missing phenomenon. The signal received by the state detection filter can be described by

$$y_i(k) = \alpha_i(k) C x_i(k) + D_{2i} v_2(k), \tag{7}$$

where $y_i(k) \in \mathbb{R}^m$ is the measurement output of the ith node and $v_2(k)$ is the disturbance input which belongs to $l_2([0,\infty);\mathbb{R}^p)$. C and D_{2i} are constant matrices with appropriate dimensions. $\alpha_i(k)$ is the Bernoulli distributed white sequences governed by

$$\begin{aligned} &\text{Prob}\{\alpha_i(k) = 1\} = \mathbb{E}\{\alpha_i(k)\} = \alpha_i, \\ &\qquad\qquad \text{when data received;} \\ &\text{Prob}\{\alpha_i(k) = 0\} = 1 - \mathbb{E}\{\alpha_i(k)\} = 1 - \alpha_i, \\ &\qquad\qquad \text{when data missing,} \end{aligned} \tag{8}$$

where $\alpha_i \in [0,1]$ is known constant.

In this paper, we are interested in obtaining $\widehat{z}_i(k)$, the estimate of the signal $z_i(k)$, from the actual measured output $y_i(k)$. We adopt the following filter to be considered for node i:

$$\begin{aligned} \widehat{x}_i(k+1) &= f(\widehat{x}_i(k)) + g(\widehat{x}_i(k - d(k))) \\ &\quad + K_i(y_i(k) - C\widehat{x}_i(k)), \\ \widehat{z}_i(k) &= M\widehat{x}_i(k), \\ \widehat{x}_i(j) &= 0, \quad j = -\overline{d}_M, -\overline{d}_M + 1, \ldots, 0; \\ &\qquad i = 1, 2, \ldots, N, \end{aligned} \tag{9}$$

where $\widehat{x}_i(k) \in \mathbb{R}^n$ is the estimate of the state $x_i(k)$, $\widehat{z}_i(k) \in \mathbb{R}^r$ is the estimate of the output $z_i(k)$, and $K_i \in \mathbb{R}^{n\times m}$ is the estimator gain matrix to be designed.

Let the estimation error be $e(k) = x(k) - \widehat{x}(k)$. By using the Kronecker product, the filtering error system can be obtained from (2), (7), and (9) as follows:

$$\begin{aligned} e_{k+1} &= \widetilde{f}_k + \widetilde{g}_{k-d_k} - K\widetilde{C}e_k + \left(W \otimes \Gamma + K\widetilde{C}\right) x_k \\ &\quad + \widetilde{L}v_k + h_k\omega_k - K\left(\sum_{i=1}^{N} \alpha_i(k) E_i \widetilde{C} x_k\right), \\ \widetilde{z}_k &= \widetilde{M}e_k, \end{aligned} \tag{10}$$

where

$$\begin{aligned} x_k &= \left[x_1^T(k) \;\; x_2^T(k) \;\; \cdots \;\; x_N^T(k)\right]^T, \\ \widehat{x}_k &= \left[\widehat{x}_1^T(k) \;\; \widehat{x}_2^T(k) \;\; \cdots \;\; \widehat{x}_N^T(k)\right]^T, \\ z_k &= \left[z_1^T(k) \;\; z_2^T(k) \;\; \cdots \;\; z_N^T(k)\right]^T, \\ \widehat{z}_k &= \left[\widehat{z}_1^T(k) \;\; \widehat{z}_2^T(k) \;\; \cdots \;\; \widehat{z}_N^T(k)\right]^T, \qquad \widetilde{z}_k = z_k - \widehat{z}_k, \\ v_k &= \left[v_1^T(k) \;\; v_2^T(k)\right]^T, \qquad d_k = d(k), \qquad w_k = w(k), \\ f(x_k) &= \left[f^T(x_1(k)) \;\; f^T(x_2(k)) \;\; \cdots \;\; f^T(x_N(k))\right]^T, \\ g(x_k) &= \left[g^T(x_1(k)) \;\; g^T(x_2(k)) \;\; \cdots \;\; g^T(x_N(k))\right]^T, \\ h(x_k) &= \left[h^T(x_1(k)) \;\; h^T(x_2(k)) \;\; \cdots \;\; h^T(x_N(k))\right]^T, \\ \widetilde{f}_k &= f(x_k) - f(\widehat{x}_k), \qquad \widetilde{g}_k = g(x_k) - g(\widehat{x}_k), \\ K &= \text{diag}\{K_1, K_2, \ldots, K_N\}, \qquad \widetilde{C} = I \otimes C, \\ D_1 &= \left[D_{11}^T \;\; D_{12}^T \;\; \cdots \;\; D_{1N}^T\right]^T, \qquad \widetilde{D} = \left[D_1 \;\; -KD_2\right], \\ D_2 &= \left[D_{21}^T \;\; D_{22}^T \;\; \cdots \;\; D_{2N}^T\right]^T, \qquad \widetilde{M} = I \otimes M, \\ E_i &= \text{diag}\left\{\underbrace{0, \ldots, 0}_{i-1}, I, \underbrace{0, \ldots, 0}_{N-i}\right\}. \end{aligned} \tag{11}$$

Setting $\eta_k = \left[x_k^T \;\; e_k^T\right]^T$, we subsequently obtain an augmented system as follows:

$$\begin{aligned} \eta_{k+1} &= \mathscr{W}\eta_k + \vec{f}_k + \vec{g}_{k-d_k} \\ &\quad + \sum_{i=1}^{N} \left(\alpha_k^i - \alpha_i\right) \mathscr{G}_i \widetilde{C} \mathscr{S} \eta_k + \mathscr{D} v_k + \mathscr{H}\omega_k, \\ \widetilde{z}_k &= \mathscr{M}\eta_k, \end{aligned} \tag{12}$$

where

$$\begin{aligned} \vec{f}_k &= \left[f^T(x_k) \;\; \widetilde{f}_k^T\right]^T, \qquad \vec{g}_k = \left[g^T(x_k) \;\; \widetilde{g}_k^T\right]^T, \\ \alpha_k^i &= \alpha_i(k), \qquad \widetilde{\alpha}^{\Lambda} = \text{diag}\{\alpha_1 I, \alpha_2 I, \ldots, \alpha_N I\}, \\ \mathscr{G}_i &= \left[0 \;\; -E_i^T K^T\right]^T, \\ \mathscr{M} &= \left[0 \;\; \widetilde{M}\right], \qquad \mathscr{S} = \left[I \;\; 0\right], \\ \mathscr{W} &= \begin{bmatrix} W \otimes \Gamma & 0 \\ W \otimes \Gamma + K\left(I - \widetilde{\alpha}^{\Lambda}\right)\widetilde{C} & -K\widetilde{C} \end{bmatrix}, \\ \mathscr{D} &= \begin{bmatrix} D_1 & 0 \\ D_1 & -KD_2 \end{bmatrix}, \qquad \mathscr{H} = \begin{bmatrix} h(x_k) \\ h(x_k) \end{bmatrix}. \end{aligned} \tag{13}$$

Definition 5 (see [34]). The filtering error system (12) is said to be exponentially stable in the mean square if, in case of $v_k = 0$, for any initial conditions, there exist constants $\varepsilon > 0$ and $0 < \kappa < 1$ such that

$$\mathbb{E}\left\{\|\eta_k\|^2\right\} \le \varepsilon\kappa^k \max_{i\in[-\overline{d}_M,\,0]} \mathbb{E}\left\{\|\eta_i\|^2\right\}, \quad k \in \mathbb{N}, \tag{14}$$

where $\eta_i := [\varphi_1^T(i), \varphi_2^T(i), \ldots, \varphi_N^T(i), \varphi_1^T(i), \varphi_2^T(i), \ldots, \varphi_N^T(i)]^T$, for all $i \in [-\overline{d}_M, 0]$.

Our aim in this paper is to develop techniques to deal with the robust H_∞ filtering problem for a class of complex systems with stochastic packet dropouts, time delays, and disturbance inputs. The augmented observer system (12) satisfies the following requirements (Q1) and (Q2), simultaneously:

(Q1) the filter error system (12) with $v_k = 0$ is exponentially stable in the mean square;

(Q2) under the zero initial condition, the filtering error $\widetilde{z}_k$ satisfies

$$\frac{1}{N}\sum_{k=0}^{\infty}\mathbb{E}\left\{\left\|\widetilde{z}_k\right\|^2\right\} \le \gamma^2\sum_{k=0}^{\infty}\left\|v_k\right\|^2 \tag{15}$$

for all nonzero v_k, where $\gamma > 0$ is a given disturbance attenuation level.

Lemma 6 (the Schur complement). *Given constant matrices S_1, S_2, and S_3, where $S_1 = S_1^T$ and $0 < S_2 = S_2^T$, then $S_1 + S_3^T S_2^{-1} S_3 < 0$ if and only if*

$$\begin{bmatrix} S_1 & S_3^T \\ S_3 & -S_2 \end{bmatrix} < 0 \quad or \quad \begin{bmatrix} -S_2 & S_3 \\ S_3^T & S_1 \end{bmatrix} < 0. \tag{16}$$

3. Main Results

In this part, we will construct the Lyapunov-Krasovskii functional and the use of linear matrix inequality to propose sufficient conditions such that the system error model in (12) could be exponentially stable in mean square. Let us first consider the robust exponential stability analysis problem for the filter error system (12) with $v_k = 0$.

Theorem 7. *Consider the system* (2) *and suppose that the estimator parameters K_i $(i = 1, 2, \ldots, N)$ are given. The system augmented error model* (12) *with $v_k = 0$ is said to be exponentially stable in mean square, if there exist positive definite matrices Q_i $(i = 1, 2, 3, 4)$ and positive scalars λ_j $(j = 1, 2, 3)$ satisfying the following inequality:*

$$\Pi_1 = \begin{bmatrix} \Xi_{11} & 0 & \Xi_{13} & \mathscr{W}^T P_1 \\ * & \Xi_{22} & 0 & \lambda_2 \Phi_2^{gT} \\ * & * & \Xi_{33} & P_1 \\ * & * & * & \Xi_{44} \end{bmatrix} < 0, \tag{17}$$

$$P_1 \le \lambda_3 I,$$

where

$$\widetilde{\alpha}_i^* = \alpha_i\left(1-\alpha_i\right), \qquad A_\varrho = \begin{bmatrix} \varrho I & 0 \\ 0 & 0 \end{bmatrix},$$

$$\Phi_1^f = I \otimes \text{Sym}\left\{\frac{1}{2}\phi_1^{fT}\phi_2^f\right\},$$

$$\Phi_2^f = I \otimes \frac{\left(\phi_1^f + \phi_2^f\right)}{2},$$

$$\Phi_1^g = I \otimes \text{Sym}\left\{\frac{1}{2}\phi_1^{gT}\phi_2^g\right\},$$

$$\Phi_2^g = I \otimes \frac{\left(\phi_1^g + \phi_2^g\right)}{2},$$

$$P_1 = \text{diag}\left\{I \otimes Q_1, I \otimes Q_2\right\}, \qquad \Xi_{22} = -P_2 - \lambda_2 \Phi_1^g,$$

$$P_2 = \text{diag}\left\{I \otimes Q_3, I \otimes Q_4\right\}, \qquad \Xi_{33} = P_1 - \lambda_1 I,$$

$$\Xi_{11} = \mathscr{W}^T P_1 \mathscr{W} - P_1 + \left(\overline{d}_M - \overline{d}_m + 1\right) P_2 + \lambda_3 A_\varrho - \lambda_1 \Phi_1^f + \sum_{i=1}^{N} \widetilde{\alpha}_i^* \mathcal{S}^T \widetilde{C}^T \mathcal{G}_i^T P_1 \mathcal{G}_i \widetilde{C} \mathcal{S},$$

$$\Xi_{13} = \mathscr{W}^T P_1 + \lambda_1 \Phi_2^{fT}, \qquad \Xi_{44} = P_1 - \lambda_2 I. \tag{18}$$

Proof. Choose the following Lyapunov functional for system (12):

$$V(k) = V_1(k) + V_2(k) + V_3(k), \tag{19}$$

where

$$V_1(k) = \eta_k^T P_1 \eta_k,$$

$$V_2(k) = \sum_{i=k-d_k}^{k-1} \eta_i^T P_2 \eta_i,$$

$$V_3(k) = \sum_{j=k-\overline{d}_M+1}^{k-\overline{d}_m} \sum_{i=j}^{k-1} \eta_i^T P_2 \eta_i. \tag{20}$$

Then, along the trajectory of system (12) with $v_k = 0$, we have

$$\begin{aligned} \mathbb{E}\left\{\Delta V_1(k)\right\} &= \mathbb{E}\left\{V_1(k+1) - V_1(k)\right\} \\ &= \mathbb{E}\left\{\eta_{k+1}^T P_1 \eta_{k+1} - \eta_k^T P_1 \eta_k\right\} \\ &= \mathbb{E}\left\{\eta_k^T \mathscr{W}^T P_1 \mathscr{W} \eta_k + \vec{f}_k^T P_1 \vec{f}_k \right. \\ &\quad + \vec{g}_{k-d_k}^T P_1 \vec{g}_{k-d_k} + \mathscr{H}^T P_1 \mathscr{H} \\ &\quad + \sum_{i=1}^{N} \widetilde{\alpha}_i^* \eta_k^T \mathcal{S}^T \widetilde{C}^T \mathcal{G}_i^T P_1 \mathcal{G}_i \widetilde{C} \mathcal{S} \eta_k \\ &\quad + 2\eta_k^T \mathscr{W}^T P_1 \vec{f}_k + 2\eta_k^T \mathscr{W}^T P_1 \vec{g}_{k-d_k} \\ &\quad \left. + 2\vec{f}_k^T P_1 \vec{g}_{k-d_k} - \eta_k^T P_1 \eta_k \right\}, \end{aligned} \tag{21}$$

$$\mathbb{E}\left\{\mathscr{H}^T P_1 \mathscr{H}\right\} \le \lambda_3 \eta_k^T A_\varrho \eta_k.$$

Next, it can be derived that

$$\mathbb{E}\left\{\Delta V_2(k)\right\} = \mathbb{E}\left\{V_2(k+1) - V_2(k)\right\}$$

$$= \mathbb{E}\left\{\sum_{i=k-d_{k+1}+1}^{k} \eta_i^T P_2 \eta_i - \sum_{i=k-d_k}^{k-1} \eta_i^T P_2 \eta_i\right\}$$

$$
\begin{aligned}
&= \mathbb{E}\left\{\eta_k^T P_2 \eta_k - \eta_{k-d_k}^T P_2 \eta_{k-d_k} \right.\\
&\qquad \left. + \sum_{i=k-d_{k+1}+1}^{k-1} \eta_i^T P_2 \eta_i - \sum_{i=k-d_k+1}^{k-1} \eta_i^T P_2 \eta_i \right\}\\
&= \mathbb{E}\left\{\eta_k^T P_2 \eta_k - \eta_{k-d_k}^T P_2 \eta_{k-d_k} + \sum_{i=k-\bar{d}_m+1}^{k-1} \eta_i^T P_2 \eta_i \right.\\
&\qquad \left. + \sum_{i=k-d_{k+1}+1}^{k-\bar{d}_m} \eta_i^T P_2 \eta_i - \sum_{i=k-d_k+1}^{k-1} \eta_i^T P_2 \eta_i \right\}\\
&\le \mathbb{E}\left\{\eta_k^T P_2 \eta_k - \eta_{k-d_k}^T P_2 \eta_{k-d_k} \right.\\
&\qquad \left. + \sum_{i=k-\bar{d}_M+1}^{k-\bar{d}_m} \eta_i^T P_2 \eta_i \right\},
\end{aligned}
$$

$$
\begin{aligned}
&\mathbb{E}\{\Delta V_3(k)\}\\
&\quad = \mathbb{E}\{V_3(k+1) - V_3(k)\}\\
&\quad = \mathbb{E}\left\{\sum_{j=k-\bar{d}_M+2}^{k-\bar{d}_m+1} \sum_{i=j}^{k} \eta_i^T P_2 \eta_i - \sum_{j=k-\bar{d}_M+1}^{k-\bar{d}_m} \sum_{i=j}^{k-1} \eta_i^T P_2 \eta_i \right\}\\
&\quad = \mathbb{E}\left\{\sum_{j=k-\bar{d}_M+1}^{k-\bar{d}_m} \sum_{i=j+1}^{k} \eta_i^T P_2 \eta_i - \sum_{j=k-\bar{d}_M+1}^{k-\bar{d}_m} \sum_{i=j}^{k-1} \eta_i^T P_2 \eta_i \right\}\\
&\quad = \mathbb{E}\left\{\sum_{j=k-\bar{d}_M+1}^{k-\bar{d}_m} \left(\eta_k^T P_2 \eta_k - \eta_j^T P_2 \eta_j\right) \right\}\\
&\quad = \mathbb{E}\left\{\left(\bar{d}_M - \bar{d}_m\right) \eta_k^T P_2 \eta_k - \sum_{i=k-\bar{d}_M+1}^{k-\bar{d}_m} \eta_i^T P_2 \eta_i \right\}.
\end{aligned}
\tag{22}
$$

Letting

$$
\xi_k = \left[\eta_k^T \quad \eta_{k-d_k}^T \quad \vec{f}_k^T \quad \vec{g}_{k-d_k}^T\right]^T, \tag{23}
$$

the combination of (21) and (22) results in

$$
\begin{aligned}
\mathbb{E}\{\Delta V(\eta_k)\} &= \mathbb{E}\{V(k+1) - V(k)\}\\
&= \sum_{i=1}^{3} \mathbb{E}\{\Delta V_i(k)\}\\
&\le \mathbb{E}\left\{\xi_k^T \widetilde{\Pi}_1 \xi_k\right\},
\end{aligned}
\tag{24}
$$

where

$$
\begin{aligned}
\widetilde{\Pi}_1 &= \begin{bmatrix} \widetilde{\Xi}_{11} & 0 & \mathscr{W}^T P_1 & \mathscr{W}^T P_1 \\ * & -P_2 & 0 & 0 \\ * & * & P_1 & P_1 \\ * & * & * & P_1 \end{bmatrix},\\
\widetilde{\Xi}_{11} &= \mathscr{W}^T P_1 \mathscr{W} - P_1 + \left(\bar{d}_M - \bar{d}_m + 1\right) P_2\\
&\quad + \lambda_3 A_\varrho + \sum_{i=1}^{N} \widetilde{\alpha}_i^* \mathcal{S}^T \widetilde{C}^T \mathcal{G}_i^T P_1 \mathcal{G}_i \widetilde{C} \mathcal{S},
\end{aligned}
\tag{25}
$$

Notice that (5) implies

$$
\begin{aligned}
\left[\vec{f}_k - \left(I \otimes \phi_1^f\right)\eta_k\right]^T \left[\vec{f}_k - \left(I \otimes \phi_2^f\right)\eta_k\right] \le 0.\\
\left[\vec{g}_k - \left(I \otimes \phi_1^g\right)\eta_k\right]^T \left[\vec{g}_k - \left(I \otimes \phi_2^g\right)\eta_k\right] \le 0.
\end{aligned}
\tag{26}
$$

From (26), it follows that

$$
\begin{aligned}
&\mathbb{E}\{\Delta V(\eta_k)\}\\
&\quad \le \mathbb{E}\left\{\xi_k^T \widetilde{\Pi}_1 \xi_k - \lambda_1 \left[\vec{f}_k - \left(I \otimes \phi_1^f\right)\eta_k\right]^T \right.\\
&\qquad \times \left[\vec{f}_k - \left(I \otimes \phi_2^f\right)\eta_k\right]\\
&\qquad - \lambda_2 \left[\vec{g}_{k-d_k} - \left(I \otimes \phi_1^g\right)\eta_{k-d_k}\right]^T\\
&\qquad \left. \times \left[\vec{g}_{k-d_k} - \left(I \otimes \phi_2^g\right)\eta_{k-d_k}\right] \right\}\\
&\quad \le \mathbb{E}\left\{\xi_k^T \Pi_1 \xi_k\right\}.
\end{aligned}
\tag{27}
$$

According to Theorem 7, we have $\Pi_1 < 0$; there must exist a sufficiently small scalar $\varepsilon_0 > 0$ such that

$$
\Pi_1 + \varepsilon_0 \operatorname{diag}\{I, 0\} < 0. \tag{28}
$$

Then, it is easy to see from (27) and (28) that the following inequality holds:

$$
\mathbb{E}\{\Delta V(\eta_k)\} \le -\varepsilon_0 \mathbb{E}\left\{\|\eta_k\|^2\right\}. \tag{29}
$$

According to the definition of $V(k)$, we can derive that

$$
\mathbb{E}\{V(k)\} \le \rho_1 \mathbb{E}\left\{\|\eta_k\|^2\right\} + \rho_2 \sum_{i=k-\bar{d}_M}^{k-1} \mathbb{E}\left\{\|\eta_i\|^2\right\}, \tag{30}
$$

where $\rho_1 = \lambda_{\max}(P_1)$ and $\rho_2 = (\bar{d}_M - \bar{d}_m + 1)\lambda_{\max}(P_2)$.

For any scalar $\mu > 1$, together with (19), the above inequality implies that

$$
\begin{aligned}
&\mu^{k+1} \mathbb{E}\{V(k+1)\} - \mu^k \mathbb{E}\{V(k)\}\\
&\quad = \mu^{k+1} \mathbb{E}\{\Delta V(k)\} + \mu^k (\mu - 1) \mathbb{E}\{V(k)\}\\
&\quad \le \epsilon_1(\mu) \mu^k \mathbb{E}\left\{\|\eta_k\|^2\right\} + \epsilon_2(\mu) \sum_{i=k-\bar{d}_M}^{k-1} \mu^k \mathbb{E}\left\{\|\eta_i\|^2\right\}
\end{aligned}
\tag{31}
$$

with $\epsilon_1(\mu) = (\mu - 1)\rho_1 - \mu\varepsilon_0$ and $\epsilon_2(\mu) = (\mu - 1)\rho_2$.

In addition, for any integer $m \geq \bar{d}_M + 1$, summing up both sides of (31) from 0 to $m-1$ with respect to k, we have

$$\begin{aligned}
&\mu^m \mathbb{E}\{V(k+1)\} - \mathbb{E}\{V(0)\} \\
&\quad \leq \epsilon_1(\mu) \sum_{k=0}^{m-1} \mu^k \mathbb{E}\left\{\|\eta_k\|^2\right\} \\
&\qquad + \epsilon_2(\mu) \sum_{k=0}^{m-1} \sum_{i=k-\bar{d}_M}^{k-1} \mu^k \mathbb{E}\left\{\|\eta_i\|^2\right\}.
\end{aligned} \tag{32}$$

Due to $\bar{d}_M \geq 1$,

$$\begin{aligned}
&\sum_{k=0}^{m-1} \sum_{i=k-\bar{d}_M}^{k-1} \mu^k \mathbb{E}\left\{\|\eta_i\|^2\right\} \\
&\quad \leq \left(\sum_{i=-\bar{d}_M}^{-1} \sum_{k=0}^{i+\bar{d}_M} + \sum_{i=0}^{m-1-\bar{d}_M} \sum_{k=i+1}^{i+\bar{d}_M} \right. \\
&\qquad \left. + \sum_{i=m-1-\bar{d}_M}^{m-1} \sum_{k=i+1}^{m-1} \right) \mu^k \mathbb{E}\left\{\|\eta_i\|^2\right\} \\
&\quad \leq \bar{d}_M \sum_{i=-\bar{d}_M}^{-1} \mu^{i+\bar{d}_M} \mathbb{E}\left\{\|\eta_i\|^2\right\} \\
&\qquad + \bar{d}_M \sum_{i=0}^{m-1-\bar{d}_M} \mu^{i+\bar{d}_M} \mathbb{E}\left\{\|\eta_i\|^2\right\} \\
&\qquad + \bar{d}_M \sum_{i=m-1-\bar{d}_M}^{m-1} \mu^{i+\bar{d}_M} \mathbb{E}\left\{\|\eta_i\|^2\right\} \\
&\quad \leq \bar{d}_M \mu^{\bar{d}_M} \max_{-\bar{d}_M \leq i \leq 0} \mathbb{E}\left\{\|\eta_i\|^2\right\} \\
&\qquad + \bar{d}_M \mu^{\bar{d}_M} \sum_{i=0}^{m-1} \mu^i \mathbb{E}\left\{\|\eta_i\|^2\right\}.
\end{aligned} \tag{33}$$

So, we can obtain from (32) and (33) the following:

$$\begin{aligned}
\mu^k \mathbb{E}\{V(k)\} &\leq \mathbb{E}\{V(0)\} + \left(\epsilon_1(\mu) + \bar{\epsilon}_2(\mu)\right) \sum_{i=0}^{k-1} \mu^i \mathbb{E}\left\{\|\eta_i\|^2\right\} \\
&\quad + \bar{\epsilon}_2(\mu) \sum_{-\bar{d}_M \leq i \leq 0} \mathbb{E}\left\{\|\eta_i\|^2\right\},
\end{aligned} \tag{34}$$

with $\bar{\epsilon}_2(\mu) = \bar{d}_M \mu^{\bar{d}_M}(\mu - 1)\rho_2$.

Let $\rho_0 = \lambda_{\min}(P_1)$ and $\rho = \max\{\rho_1, \rho_2\}$. It is obvious from (19) that

$$\mathbb{E}\{V(k)\} \geq \rho_0 \mathbb{E}\left\{\|\eta_k\|^2\right\}. \tag{35}$$

Meanwhile, we can find easily from (30) that

$$\mathbb{E}\{V(0)\} \leq \rho\left(2\bar{d}_M + 1\right) \max_{-\bar{d}_M \leq i \leq 0} \mathbb{E}\left\{\|\eta_i\|^2\right\}. \tag{36}$$

It can be verified that there exists a scalar $\mu_0 > 1$ such that

$$\epsilon_1(\mu_0) + \bar{\epsilon}_2(\mu_0) = 0. \tag{37}$$

Therefore, from (34)–(37), it is clear to see that

$$\begin{aligned}
&\mathbb{E}\left\{\|\eta_k\|^2\right\} \\
&\quad \leq \left(\frac{1}{\mu_0}\right)^k \frac{\rho\left(2\bar{d}_M + 1\right) + \bar{d}_M \bar{\epsilon}_2(\mu_0)}{\rho_0} \max_{-\bar{d}_M \leq i \leq 0} \mathbb{E}\left\{\|\eta_i\|^2\right\}.
\end{aligned} \tag{38}$$

The augmented system (12) with $v_k = 0$ is exponentially mean-square stable according to Definition 5. The proof is complete. □

Next, we will analyze the performance of the filtering error system (12).

Theorem 8. *Consider the system* (2) *and suppose that the estimator parameters* K_i $(i = 1, 2, \ldots, N)$ *are given. The system augmented error model* (12) *is said to be exponentially stable in mean square and satisfies the* H_∞ *performance constraint* (15) *for all nonzero* v_k *and* ω_k *under the zero initial condition, if there exist positive definite matrices* Q_i $(i = 1, 2, 3, 4)$ *and positive scalars* λ_j $(j = 1, 2, 3)$ *satisfying the following inequality:*

$$\Pi_2 = \begin{bmatrix}
\Xi_{11}^* & 0 & \Xi_{13} & \mathscr{W}^T P_1 & \mathscr{W}^T P_1 \mathscr{D} \\
* & \Xi_{22} & 0 & \lambda_2 \Phi_2^{gT} & 0 \\
* & * & \Xi_{33} & P_1 & P_1 \mathscr{D} \\
* & * & * & \Xi_{44} & P_1 \mathscr{D} \\
* & * & * & * & \mathscr{D}^T P_1 \mathscr{D} - \gamma^2 I
\end{bmatrix} < 0, \tag{39}$$

$$P_1 \leq \lambda_3 I,$$

where

$$\begin{aligned}
\Xi_{11}^* &= \mathscr{W}^T P_1 \mathscr{W} - P_1 + \left(\bar{d}_M - \bar{d}_m + 1\right) P_2 \\
&\quad - \lambda_1 \Phi_1^f + \lambda_3 A_\varrho \\
&\quad + \frac{1}{N} \mathscr{M}^T \mathscr{M} + \sum_{i=1}^{N} \tilde{\alpha}_i^* \mathcal{S}^T \widetilde{C}^T \mathcal{G}_i^T P_1 \mathcal{G}_i \widetilde{C} \mathcal{S},
\end{aligned} \tag{40}$$

and other parameters are defined as in Theorem 7.

Proof. It is clear that (39) implies (17). According to Theorem 7, the filtering error system (12) with $v_k = 0$ is robustly exponentially stable in the mean square.

Let us now deal with the performance of the system (15). Construct the same Lyapunov-Krasovskii functional as in

Theorem 7. A similar calculation as in the proof of Theorem 7 leads to

$$\begin{aligned}\mathbb{E}\{\Delta V(k)\} \le \mathbb{E}\Big\{&\xi_k^T\Pi_1\xi_k + 2v_k^T\mathscr{D}^T P_1\mathscr{W}\eta_k \\ &+ 2v_k^T\mathscr{D}^T P_1\vec{f}_k + 2v_k^T\mathscr{D}^T P_1\vec{g}_{k-d_k} \\ &+v_k^T\mathscr{D}^T P_1\mathscr{D}v_k\Big\},\end{aligned} \tag{41}$$

where ξ_k and Π_1 are defined previously.

Setting $\widetilde{\xi}_k = \begin{bmatrix}\xi_k^T & v_k^T\end{bmatrix}^T$, inequality (41) can be rewritten as

$$\mathbb{E}\{\Delta V(k)\} \le \mathbb{E}\left\{\widetilde{\xi}_k^T\begin{bmatrix}\Pi_1 & \widetilde{\mathscr{D}}^T \\ * & \mathscr{D}^T P_1\mathscr{D}\end{bmatrix}\widetilde{\xi}_k\right\}, \tag{42}$$

where $\widetilde{\mathscr{D}} = \begin{bmatrix}\mathscr{D}^T P_1\mathscr{W} & 0 & \mathscr{D}^T P_1 & \mathscr{D}^T P_1\end{bmatrix}$.

In order to deal with the H_∞ performance of the filtering system (12), we introduce the following index:

$$\mathscr{J}(s) = \mathbb{E}\sum_{k=0}^{s}\left\{\frac{1}{N}\|\widetilde{z}_k\|^2 - \gamma^2\|v_k\|^2\right\}, \tag{43}$$

where s is nonnegative integer.

Under the zero initial condition, one has

$$\begin{aligned}\mathscr{J}(s) &= \mathbb{E}\sum_{k=0}^{s}\left\{\frac{1}{N}\|\widetilde{z}_k\|^2 - \gamma^2\|v_k\|^2 + \Delta V(k)\right\} \\ &\quad - \mathbb{E}\{V(s+1)\} \\ &\le \mathbb{E}\sum_{k=0}^{s}\left\{\frac{1}{N}\|\widetilde{z}_k\|^2 - \gamma^2\|v_k\|^2 + \Delta V(k)\right\} \\ &\le \mathbb{E}\sum_{k=0}^{s}\left\{\widetilde{\xi}_k^T\Pi_2\widetilde{\xi}_k\right\} < 0.\end{aligned} \tag{44}$$

According to Theorem 8, we have $\mathscr{J}(s) \le 0$. Letting $s \to \infty$, we obtain

$$\frac{1}{N}\sum_{k=0}^{\infty}\mathbb{E}\left\{\|\widetilde{z}_k\|^2\right\} \le \gamma^2\sum_{k=0}^{\infty}\|v_k\|^2, \tag{45}$$

and the proof is now complete. □

We aim at solving the filter design problem for complex network (2). Therefore, we are in a position to consider the H_∞ filter design problem for the complex network (2). The following theorem provides sufficient conditions for the existence of such filters for system (12). The following result can be easily accessible from Theorem 8, and the proof is therefore omitted.

Theorem 9. *Consider the system* (2) *and suppose that the disturbance attenuation level* $\gamma > 0$ *is given. The system augmented error model* (12) *is said to be exponentially stable in mean square and satisfies the* H_∞ *performance constraint* (15) *for all nonzero* v_k *and* ω_k *under the zero initial condition, if there exist positive definite matrices* Q_i $(i = 1, 2, 3, 4)$*, matrices* Y_i $(i = 1, 2, \ldots, N)$*, and positive scalars* λ_j $(j = 1, 2, 3)$ *satisfying the following inequality:*

$$\Pi_3 = \begin{bmatrix}\Pi_{11} & \Pi_{12} & 0 & \Pi_{14} & \Pi_{15} & \Pi_{16} & \Pi_{18} & \Pi_{19} \\ * & \Pi_{22} & 0 & \Pi_{24} & \Pi_{25} & \Pi_{26} & \Pi_{28} & 0 \\ * & * & \Pi_{33} & 0 & \Pi_{35} & 0 & 0 & 0 \\ * & * & * & \Pi_{44} & \Pi_{45} & \Pi_{46} & 0 & 0 \\ * & * & * & * & \Pi_{55} & \Pi_{56} & 0 & 0 \\ * & * & * & * & * & \Pi_{66} & \Pi_{68} & 0 \\ * & * & * & * & * & * & -\mathscr{Q}_2 & 0 \\ * & * & * & * & * & * & * & -\mathscr{Q}_2\end{bmatrix} < 0,$$

$$P_1 \le \lambda_3 I, \tag{46}$$

where

$$\vec{Y} = \operatorname{diag}\{Y_1, Y_2, \ldots, Y_N\}, \qquad \mathscr{Q}_2 = I \otimes Q_2, \tag{47}$$

$$Z = \begin{bmatrix}\sqrt{\widetilde{\alpha}_1^*}E_1\vec{Y}^T & \sqrt{\widetilde{\alpha}_2^*}E_2\vec{Y}^T & \cdots & \sqrt{\widetilde{\alpha}_N^*}E_N\vec{Y}^T\end{bmatrix}^T,$$

$$\Pi_{19} = (I \otimes C)^T Z^T,$$

$$\Pi_{18} = (I \otimes C)^T\left(I - \widetilde{\alpha}^\Lambda\right)^T\vec{Y}^T,$$

$$\Pi_{28} = -(I \otimes C)^T\vec{Y}^T,$$

$$\Pi_{68} = -D_2^T\vec{Y}^T,$$

$$\begin{aligned}\Pi_{11} = &\left(W^T W\right) \otimes \left(\Gamma\left(Q_1 + Q_2\right)\Gamma\right) \\ &+ \operatorname{Sym}\left\{(W \otimes \Gamma)^T\vec{Y}\left(I - \widetilde{\alpha}^\Lambda\right)(I \otimes C)\right\} \\ &- I \otimes \left(Q_1 - \left(\overline{d}_M - \overline{d}_m + 1\right)Q_4\right) \\ &- \lambda_1\Phi_1^f + \lambda_3 A_\varrho,\end{aligned}$$

$$\Pi_{12} = -(W \otimes \Gamma)^T\vec{Y}(I \otimes C),$$

$$\begin{aligned}\Pi_{14} = \Big[&W^T \otimes (\Gamma Q_1) + \lambda_1\Phi_2^{fT}W^T \otimes (\Gamma Q_2) \\ &+(I \otimes C)^T\left(I - \widetilde{\alpha}^\Lambda\right)\vec{Y}^T\Big],\end{aligned}$$

$$\begin{aligned}\Pi_{15} = \Big[&W^T \otimes (\Gamma Q_1)\, W^T \otimes (\Gamma Q_2) \\ &+(I \otimes C)^T\left(I - \widetilde{\alpha}^\Lambda\right)\vec{Y}^T\Big],\end{aligned}$$

$$\begin{aligned}\Pi_{16} = \Big[&W^T \otimes \left(\Gamma\left(Q_1 + Q_2\right)\right)D_1 + (I \otimes C)^T\left(I - \widetilde{\alpha}^\Lambda\right)\vec{Y}^T D_2 \\ &-(W \otimes \Gamma)^T\vec{Y}D_2\Big],\end{aligned}$$

$$\begin{aligned}\Pi_{22} = &-I \otimes \left(Q_2 - \left(\overline{d}_M - \overline{d}_m + 1\right)Q_4\right) \\ &- \lambda_1\Phi_1^f + \frac{1}{N}\left(I \otimes \left(M^T M\right)\right),\end{aligned}$$

$$\Pi_{24} = \begin{bmatrix}0 & (I \otimes C)^T\vec{Y}^T + \lambda_1\Phi_2^{fT}\end{bmatrix},$$

$$\Pi_{25} = \begin{bmatrix}0 & (I \otimes C)^T\vec{Y}^T\end{bmatrix},$$

$$\Pi_{26} = \left[-(I \otimes C)^T \vec{Y}^T D_1 \quad 0\right],$$
$$\Pi_{33} = -\operatorname{diag}\{I \otimes Q_3, I \otimes Q_4\} - \lambda_2 \Phi_1^g,$$
$$\Pi_{35} = \lambda_2 \Phi_2^{gT},$$
$$\Pi_{44} = \operatorname{diag}\{I \otimes Q_1, I \otimes Q_2\} - \lambda_1 I,$$
$$\Pi_{45} = \operatorname{diag}\{I \otimes Q_1, I \otimes Q_2\},$$
$$\Pi_{55} = \operatorname{diag}\{I \otimes Q_1, I \otimes Q_2\} - \lambda_2 I,$$
$$\Pi_{46} = \Pi_{56} = \begin{bmatrix} (I \otimes Q_2) D_1 & 0 \\ (I \otimes Q_2) D_1 & -\vec{Y} D_2 \end{bmatrix},$$
$$\Pi_{66} = \begin{bmatrix} D_1^T (I \otimes (Q_1 + Q_2)) D_1 - \gamma^2 I & -D_1^T \vec{Y} D_2 \\ * & -\gamma^2 I \end{bmatrix}, \tag{48}$$

and other parameters are defined as in Theorem 7. Moreover, if the above inequality is feasible, the desired state estimator gains can be determined by

$$K_i = Q_2^{-1} Y_i. \tag{49}$$

4. Numerical Simulations

In this section, we present an illustrative example to demonstrate the effectiveness of the proposed theorems. Considering the system model (2) with three sensors, the system data are given as follows:

$$W = \begin{bmatrix} -0.4 & 0.4 & 0 \\ 0.4 & -0.6 & 0.2 \\ 0 & 0.2 & -0.2 \end{bmatrix}, \qquad \Gamma = \operatorname{diag}\{0.1, 0.1\},$$
$$D_{11} = \begin{bmatrix} 0.14 \\ -0.15 \end{bmatrix}, \qquad D_{12} = \begin{bmatrix} 0.1 \\ 0.12 \end{bmatrix},$$
$$D_{13} = \begin{bmatrix} 0.1 \\ -0.05 \end{bmatrix}, \qquad M = \begin{bmatrix} 0.5 & 0.7 \end{bmatrix},$$
$$D_{21} = \begin{bmatrix} 0.1 \\ -0.1 \end{bmatrix}, \qquad D_{22} = \begin{bmatrix} -0.1 \\ 0.2 \end{bmatrix},$$
$$D_{23} = \begin{bmatrix} 0.2 \\ -0.15 \end{bmatrix}, \qquad C = \begin{bmatrix} 0.8 & 0.5 \\ 0.9 & -0.3 \end{bmatrix},$$
$$f(x_i(k)) = \begin{bmatrix} -0.6 x_{i1}(k) + 0.3 x_{i2}(k) + \tanh(0.3 x_{i1}(k)) \\ 0.6 x_{i2}(k) - \tanh(0.2 x_{i2}(k)) \end{bmatrix},$$
$$g(x_i(k)) = \begin{bmatrix} 0.02 x_{i1}(k) + 0.06 x_{i2}(k) \\ -0.03 x_{i1}(k) + 0.02 x_{i2}(k) + \tanh(0.01 x_{i1}(k)) \end{bmatrix},$$
$$h(x_i(k)) = 0.15 x_i(k), \qquad d(k) = 2 + \sin\left(\frac{\pi k}{2}\right),$$
$$v_1(k) = 3 \exp(-0.3k) \cos(0.2k),$$
$$v_2(k) = 2 \exp(-0.2k) \sin(0.1k). \tag{50}$$

Then, it is easy to see that the constraint (26) can be met with

$$\phi_1^f = \begin{bmatrix} -0.6 & 0.3 \\ 0 & 0.4 \end{bmatrix}, \qquad \phi_2^f = \begin{bmatrix} -0.3 & 0.3 \\ 0 & 0.6 \end{bmatrix},$$
$$\phi_1^g = \begin{bmatrix} 0.02 & 0.06 \\ -0.03 & 0.02 \end{bmatrix}, \qquad \phi_2^g = \begin{bmatrix} 0.02 & 0.06 \\ -0.02 & 0.02 \end{bmatrix}. \tag{51}$$

Let the disturbance attenuation level be $\gamma = 0.96$. Assume that the initial values $\varphi_i(k)$ $(i = 1, 2, 3;\ k = -3, -2, -1, 0)$ are generated that obey uniform distribution over $[-1.5, 1.5]$, $\alpha_1 = 0.88$, $\alpha_2 = 0.85$, and $\alpha_3 = 0.87$, and the delay parameters are chosen as $\underline{d}_m = 1$ and $\underline{d}_M = 3$.

By applying Theorem 9 with help from MATLAB, we can obtain the desired filter parameters as follows:

$$\lambda_1 = 23.9040, \qquad \lambda_2 = 50.2256, \qquad \lambda_3 = 14.0023,$$
$$Q_1 = \begin{bmatrix} 9.1446 & 3.8908 \\ 3.8908 & 3.5213 \end{bmatrix},$$
$$Q_2 = \begin{bmatrix} 9.0768 & 3.7262 \\ 3.7262 & 6.7169 \end{bmatrix},$$
$$Q_3 = \begin{bmatrix} 0.7093 & -0.1693 \\ -0.1693 & 0.2576 \end{bmatrix}, \tag{52}$$
$$Q_4 = \begin{bmatrix} 0.8390 & -0.3283 \\ -0.3283 & 0.8284 \end{bmatrix},$$
$$Y_1 = \begin{bmatrix} 0.5154 & -0.9270 \\ 0.9371 & -0.6966 \end{bmatrix},$$
$$Y_2 = \begin{bmatrix} 0.6617 & -1.1042 \\ 1.0172 & -0.8178 \end{bmatrix},$$
$$Y_3 = \begin{bmatrix} 0.2943 & -0.7441 \\ 0.8713 & -0.6346 \end{bmatrix}. \tag{53}$$

Then, according to (49), the desired estimator parameters can be designed as

$$K_1 = \begin{bmatrix} -0.0006 & -0.0771 \\ 0.1399 & -0.0609 \end{bmatrix},$$
$$K_2 = \begin{bmatrix} 0.0139 & -0.0928 \\ 0.1437 & -0.0703 \end{bmatrix}, \tag{54}$$
$$K_3 = \begin{bmatrix} -0.0270 & -0.0559 \\ 0.1447 & -0.0635 \end{bmatrix}.$$

Simulation results are shown in Figures 1, 2, 3, and 4, where Figures 1–3 plot the missing measurements and ideal measurements for sensors 1–3, respectively, and Figure 4 depicts the output errors. From those figures, we can confirm the superiority of the designed H_∞ filter.

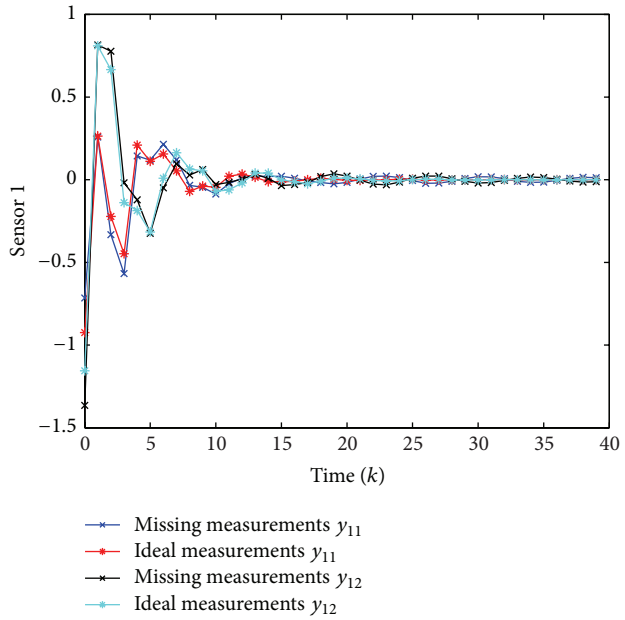


Figure 1: The ideal measurements and the missing measurements of $y_1(k)$.

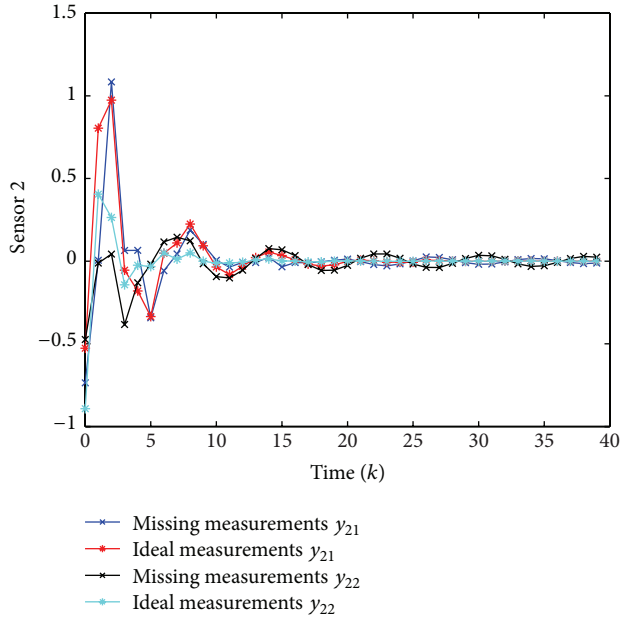


Figure 2: The ideal measurements and the missing measurements of $y_2(k)$.

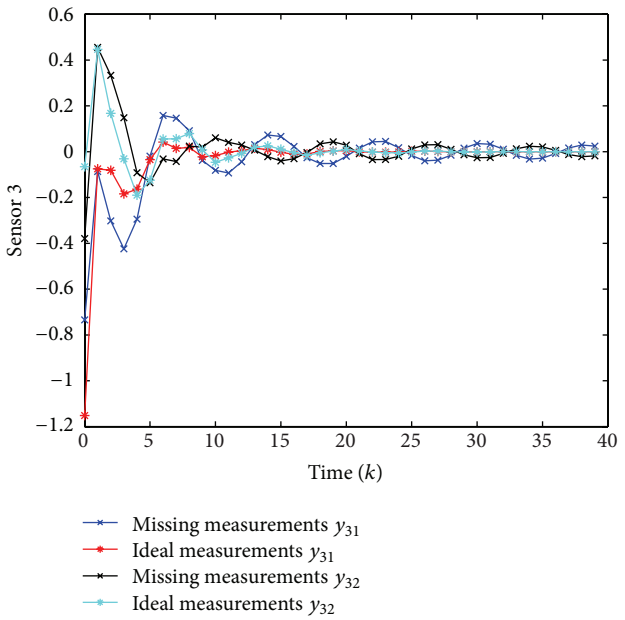


Figure 3: The ideal measurements and the missing measurements of $y_3(k)$.

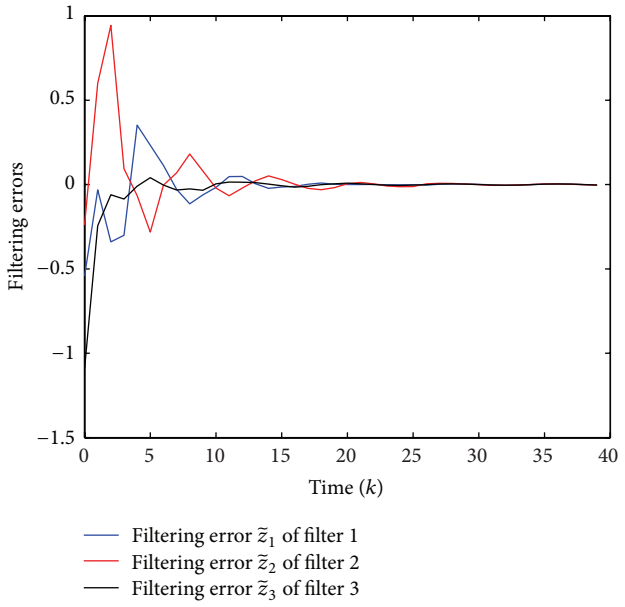


Figure 4: The estimator errors $\tilde{z}_i(k)$ $(i = 1, 2, 3)$.

5. Conclusions

In this paper, we have studied the robust H_∞ filtering problem for a class of complex systems with stochastic packet dropouts, time delays, and disturbance inputs. The discrete-time system under study involves multiplicative noises, time-varying delays, sector-bounded nonlinearities, and stochastic packet dropouts. By means of LMIs, sufficient conditions for the robustly exponential stability of the filtering error dynamics have been obtained and, at the same time, the prescribed disturbance rejection attenuation level has been guaranteed. Then, the explicit expression of the desired filter parameters has been derived. A numerical example has been provided to show the usefulness and effectiveness of the proposed design method.

Conflict of Interests

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

Acknowledgments

This work has been supported by the National Natural Science Foundation of China (Grant no. 61104109), the Natural Science Foundation of Jiangsu Province of China (Grant no. BK2011703), the Support of Science and Technology and Independent Innovation Foundation of Jiangsu Province of China (Grant no. BE2012178), the NUST Outstanding Scholar Supporting Program, and the Doctoral Fund of Ministry of Education of China (Grant no. 20113219110027).

References

[1] J. Hu, Z. Wang, H. Gao, and L. K. Stergioulas, "Extended Kalman filtering with stochastic nonlinearities and multiple missing measurements," *Automatica*, vol. 48, no. 9, pp. 2007–2015, 2012.

[2] Z. Wang, X. Liu, Y. Liu, J. Liang, and V. Vinciotti, "An extended kalman filtering approach to modeling nonlinear dynamic gene regulatory networks via short gene expression time series," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 6, no. 3, pp. 410–419, 2009.

[3] F. Yang, Z. Wang, and Y. S. Hung, "Robust Kalman filtering for discrete time-varying uncertain systems with multiplicative noises," *IEEE Transactions on Automatic Control*, vol. 47, no. 7, pp. 1179–1183, 2002.

[4] L. Wu, P. Shi, H. Gao, and C. Wang, "H∞ filtering for 2D Markovian jump systems," *Automatica*, vol. 44, no. 7, pp. 1849–1858, 2008.

[5] L. Wu and D. W. C. Ho, "Fuzzy filter design for Itô stochastic systems with application to sensor fault detection," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 1, pp. 233–242, 2009.

[6] X. Su, P. Shi, L. Wu, and Y. Song, "A novel approach to filter design for T-S fuzzy discretetime systems with time-varying delay," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 6, pp. 1114–1129, 2012.

[7] X. Su, L. Wu, and P. Shi, "Senor networks with random link failures: distributed filtering for T-S fuzzy systems," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 3, pp. 1739–1750, 2013.

[8] X. Su, P. Shi, L. Wu, and S. K. Nguang, "Induced l2 filtering of fuzzy stochastic systems with time-varying delays," *IEEE Transactions on Cybernetics*, vol. 43, no. 4, pp. 1251–1264, 2013.

[9] X. Yao, L. Wu, and W. X. Zheng, "Fault detection filter design for Markovian jump singular systems with intermittent measurements," *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3099–3109, 2011.

[10] G. Wei, Z. Wang, and H. Shu, "Robust filtering with stochastic nonlinearities and multiple missing measurements," *Automatica*, vol. 45, no. 3, pp. 836–841, 2009.

[11] Z. Wang, F. Yang, D. W. C. Ho, and X. Liu, "Robust finite-horizon filtering for stochastic systems with missing measurements," *IEEE Transactions on Signal Processing*, vol. 12, no. 6, pp. 137–440, 2005.

[12] P. Shi, M. Mahmoud, S. K. Nguang, and A. Ismail, "Robust filtering for jumping systems with mode-dependent delays," *Signal Processing*, vol. 86, no. 1, pp. 140–152, 2006.

[13] S. Sun, L. Xie, and W. Xiao, "Optimal full-order and reduced-order estimators for discrete-time systems with multiple packet dropouts," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 4031–4038, 2008.

[14] H. Dong, Z. Wang, and H. Gao, "Distributed H_∞ filtering for a class of Markovian jump nonlinear time-delay systems over lossy sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 10, pp. 4665–4672, 2013.

[15] H. Dong, Z. Wang, and H. Gao, "Distributed filtering for a class of time varying systems over sensor networks with quantization errors and successive packet dropouts," *IEEE Transactions on Signal Processing*, vol. 60, no. 6, pp. 3164–3173, 2012.

[16] H. Dong, Z. Wang, D. W. C. Ho, and H. Gao, "Robust H∞ filtering for Markovian jump systems with randomly occurring nonlinearities and sensor saturation: the finite-horizon case," *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3048–3057, 2011.

[17] Z. Wang, H. Dong, B. Shen, and H. Gao, "Finite-horizon H_∞ filtering with missing measurements and quantization effects," *IEEE Transactions on Automatic Control*, vol. 58, no. 7, pp. 1707–1718, 2013.

[18] B. Shen, Z. Wang, and X. Liu, "Sampled-data synchronization control of complex dynamical networks with stochastic sampling," *IEEE Transactions on Automatic Control*, vol. 57, no. 10, pp. 2644–2650, 2012.

[19] Y. Tang, Z. Wang, and J.-A. Fang, "Pinning control of fractional-order weighted complex networks," *Chaos*, vol. 19, no. 1, Article ID 013112, 2009.

[20] 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.

[21] C. Lin, Z. Wang, and F. Yang, "Observer-based networked control for continuous-time systems with random sensor delays," *Automatica*, vol. 45, no. 2, pp. 578–584, 2009.

[22] Y. Liu, Z. Wang, J. Liang, and X. Liu, "Synchronization and state estimation for discrete-time complex networks with distributed delays," *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 38, no. 5, pp. 1314–1325, 2008.

[23] 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.

[24] L. Wu, Z. Feng, and W. X. Zheng, "Exponential stability analysis for delayed neural networks with switching parameters: average dwell time approach," *IEEE Transactions on Neural Networks*, vol. 21, no. 9, pp. 1396–1407, 2010.

[25] L. Wu, Z. Feng, and J. Lam, "Stability and synchronization of discrete-time neural networks with switching parameters and time-varying delays," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 12, pp. 1957–1972, 2013.

[26] B. Shen, Z. Wang, and X. Liu, "Bounded H_∞ synchronization and state estimation for discrete time-varying stochastic complex networks over a finite horizon," *IEEE Transactions on Neural Networks*, vol. 22, no. 1, pp. 145–157, 2011.

[27] B. Shen, Z. Wang, D. Ding, and H. Shu, "H_∞ state estimation for complex networks with uncertain inner coupling and incomplete measurements," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 12, pp. 2027–2037, 2013.

[28] D. Ding, Z. Wang, B. Shen, and H. Shu, "H_∞ state estimation for discrete-time complex networks with randomly occurring sensor saturations and randomly varying sensor delays," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 5, pp. 725–736, 2012.

[29] Y. Liu, Z. Wang, J. Liang, and X. Liu, "Synchronization and state estimation for discrete-time complex networks with distributed delays," *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 38, no. 5, pp. 1314–1325, 2008.

[30] Y. Wang, Z. Wang, and J. Liang, "Global synchronization for delayed complex networks with randomly occurring nonlinearities and multiple stochastic disturbances," *Journal of Physics A: Mathematical and Theoretical*, vol. 42, no. 13, Article ID 135101, 2009.

[31] Z. 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.

[32] Y. Liu, Z. Wang, and X. Liu, "Exponential synchronization of complex networks with Markovian jump and mixed delays," *Physics Letters A: General, Atomic and Solid State Physics*, vol. 372, no. 22, pp. 3986–3998, 2008.

[33] Y. Wang, Z. Wang, and J. Liang, "A delay fractioning approach to global synchronization of delayed complex networks with stochastic disturbances," *Physics Letters A: General, Atomic and Solid State Physics*, vol. 372, no. 39, pp. 6066–6073, 2008.

[34] Z. Wang, D. W. C. Ho, Y. Liu, and X. Liu, "Robust H_∞ control for a class of nonlinear discrete time-delay stochastic systems with missing measurements," *Automatica*, vol. 45, no. 3, pp. 684–691, 2009.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 608984, 6 pages
http://dx.doi.org/10.1155/2014/608984

Research Article

Secure and Fair Cluster Head Selection Protocol for Enhancing Security in Mobile Ad Hoc Networks

B. Paramasivan[1] and M. Kaliappan[2]

[1] *Department of Computer Science and Engineering, National Engineering College, Kovilpatti, Tamil Nadu 628503, India*
[2] *Department of Information Technology, National Engineering College, Kovilpatti, Tamil Nadu 628503, India*

Correspondence should be addressed to M. Kaliappan; kalsrajan@yahoo.co.in

Received 27 December 2013; Accepted 6 March 2014; Published 27 March 2014

Academic Editors: Z. Yu and W. Zhang

Mobile ad hoc networks (MANETs) are wireless networks consisting of number of autonomous mobile devices temporarily interconnected into a network by wireless media. MANETs become one of the most prevalent areas of research in the recent years. Resource limitations, energy efficiency, scalability, and security are the great challenging issues in MANETs. Due to its deployment nature, MANETs are more vulnerable to malicious attack. The secure routing protocols perform very basic security related functions which are not sufficient to protect the network. In this paper, a secure and fair cluster head selection protocol (SFCP) is proposed which integrates security factors into the clustering approach for achieving attacker identification and classification. Byzantine agreement based cooperative technique is used for attacker identification and classification to make the network more attack resistant. SFCP used to solve this issue by making the nodes that are totally surrounded by malicious neighbors adjust dynamically their belief and disbelief thresholds. The proposed protocol selects the secure and energy efficient cluster head which acts as a local detector without imposing overhead to the clustering performance. SFCP is simulated in network simulator 2 and compared with two protocols including AODV and CBRP.

1. Introduction

MANETs are formed arbitrarily by a set of mobile devices falling within the transmission range of each other [1]. Routing protocols act as binding force in MANETs and facilitate communication beyond the physical wireless range of the nodes. In MANETs, every node cooperates with other nodes for forwarding packets to their destination by acting as a router. These protocols could operate in either flat or hierarchical network architecture. In the flat architecture, all nodes participate in the routing process. In the hierarchical architecture nodes are divided into a number of clusters each of which is managed by a cluster head that makes control decisions for cluster members. In this architecture, only cluster heads and gateway nodes are participating in the routing. Traditional MANETs routing protocols have no predefined trust exists between communication partners. This may render the network vulnerable to malicious attacks. Selfish nodes do not propagate packets from other nodes, while malicious nodes may perform modification and impersonation attacks against the network traffic. Clustering schemes [2] organized the network into one hop disjoint clusters then elect the most qualified and trustworthy nodes which played the role of cluster heads. Cluster heads are responsible for monitoring all the routing activities within the cluster itself.

Yu and Chong [3] and Bechler et al. [4] have proposed the survey about various clustering schemes for mobile ad hoc networks. In their paper, typical clustering schemes of MANETs are classified into six categories. In Ds based clustering, a set of dominating nodes act as cluster heads to relay routing information and data packets, such a set of nodes are called a dominating set (DS). A DS is called a connected DS (CDS) if all the dominating nodes are directly connected with each other. Low maintenance clustering protocol aimed for providing stable cluster architecture by reducing

the reaffiliation rate and minimizing the reclustering situations which improve the network life time. It causes more communication overhead. The mobility aware clustering provides the cluster architecture based on mobility behavior of nodes. The idea is by grouping mobile nodes with similar speed into the same cluster; the intracluster links become tightly connected. In this approach, the reaffiliation and reclustering rate could be naturally decreased. MOBIC proposed an aggregate local mobility metric for cluster formation in which mobile nodes with low speed relative to their neighbors have more chance to become cluster heads. The energy efficient clustering approach has achieved the less energy consumption among mobile nodes that is also avoiding the node failure. In load balancing clustering approach, an optimum number of nodes are used to form the clusters. It set upper and lower limits on the number of mobile nodes that a cluster can deal with. Reclustering procedures are invoked for cluster maintenance that adjusts the number of nodes in that cluster. The combined metrics based clustering considered number of metrics to cluster configuration including node degree, residual energy capacity, and speed. This category aimed to elect most suitable cluster head in a local area and does not give preference to mobile nodes with certain attributes such as lowest ID or highest node degree. Advantage of this clustering scheme is flexibly adjusting the weighting factors for each metric for different scenarios. In this survey, it is not guaranteed that any one of them is the best for all situations.

Several trust models [5–8] have been proposed for self-organizing networks in distributed paradigm. Jiang and Baras [9] examined the efficiency of trust based reactive routing protocols in the presence of attacks in the networks. This method is considered first-hand information to evaluate other node's trust values to make trustworthiness. Yan et al. [10] proposed a secure AODV based routing protocol for an ad hoc network which is established a secure end-to-end route. The trust values are calculated based on direct observation which is transitive. Pirzada and McDonald [11] enhanced the trust management by considering the confidence level of trust of each node. They have used confidence level as a weight to compute trust value. Ghosh et al. [12] developed a trust model to strengthen the security of MANETs and they dealt with the issues associated with recommendations. Their model was utilized only trusted routes for making effective communication and isolates the malicious nodes based on the evidence obtained from direct interactions and recommendations. Ghosh et al. [13] proposed a mechanism for distinguishing selfish peers from cooperative nodes that is based on local monitoring. In order to distinguish between selfish and cooperative peers, a series of well-known statistical tests are applied for obtaining features from the observed AODV actions.

Noman Mohammed et al [9] proposed a Mechanism design based secure leader election model for encouraging mobile nodes to honestly participate in the election process in order to avoid activities of selfish nodes and balance the energy consumption among all nodes for increasing lifetime of MANETs. The objective of mechanism design [4] is to address problem of designing incentives for nodes to provide truthful information and computing optimal system wide solution for finding the optimal cost efficient leaders. Vickrey, Clarke, and Groves (VCG) model is applied for node incentives to ensure truth telling to be the dominant strategy for any node. They have proposed local election algorithms, namely, cluster-dependent leader election and cluster independent leader election which provided globally optimal election solutions with a low cost. The Nodes with the most remaining energy are elected as the cluster head. This approach makes storage overhead because the cluster head kept an extra service table and each node maintains a reputation table and neighboring nodes list.

Milan et al. proposed a scheme [14], where a game theoretic model is applied to study the impact of collisions on a hop-by-hop reputation based mechanism for regular networks with uniform random traffic. The nodes in MANETs are equipped with different resources and provide discrete services. It did not deal with irregular topologies and nonuniform routing. It also discussed the perception and interaction asymmetries that could impair cooperation between nodes.

Safa et al. presented a cluster based trust aware routing protocol (CBTRP) [15] to ensure secure routing path and established the trust based environment. This mechanism is used to distinguish the trusted nodes from malicious nodes. CBTRP makes use of the weighted clustering algorithm (WCA) [16] to elect cluster heads. The weighted degrees are taken into consideration such as battery power, number of neighbors, transmission power, and mobility of the nodes to form optimal cluster head. CBTRP has also taken security into account to form trusted clusters. It organized the network into 1-hop clusters in which every node is able to elect the most qualified and trustworthy node to be its cluster head. Cluster members forward the packets through the trusted cluster heads. Malicious nodes do not forward the packets to them. In CBTRP model, the trust value is computed based on the information that one node can gather about the other node's vital information including analyzing the received, forwarded, and overheard packets. Analyzing the node's behavior shows that the node is selfish, acting like a black hole, and carrying out a modification attack, fabrication attack and latency delays. This approach provides improved connectivity in MANETs in the presence of malicious nodes and also it ensured the passage of packets through trusted routes only by behavior of each node. Once a malicious node is discovered, it is isolated from the network such that no packet is forwarded from it.

Chatterjee et al. [17] proposed a secure trusted auction oriented clustering based routing protocol (STACRP) to provide trusted structured framework for MANETs. Two auction mechanisms, namely, procurement and Dutch, are used to determine the forwarding cost of one hop. STACRP organized the network into 1-hop clusters and elects the trustworthy nodes as cluster head (CH) by using a secret voting scheme. Each node maintains information of itself and its neighboring nodes for cluster maintenance. The trust model is analyzed using Markov chain which guarantees to selfish node to revoke its status from warned status to normal status by proper forwarding of others packets. This achieved

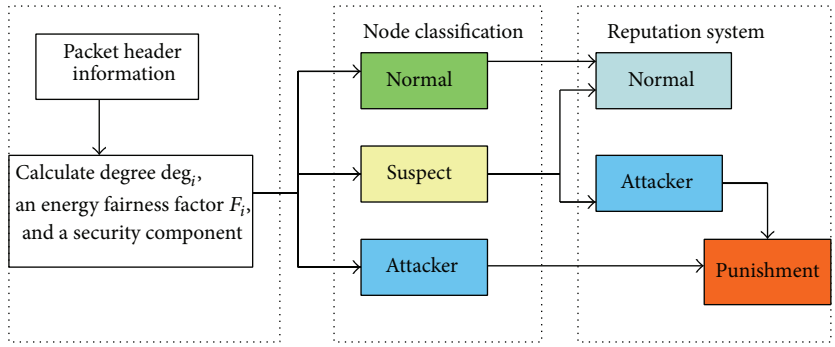


Figure 1: Byzantine agreement classification system.

a secure reliable routing solution. STACRP detected selfish nodes and enforces cooperation between nodes to achieve better throughput and packet delivery ratio with less routing overhead.

2. Materials and Methods

SFCP makes use of the weighted clustering (WCA) [18] to form 1-hop cluster in the networks. In addition to that, SFCP takes security component to form trusted CHs and also considers each node's remaining energy level to elect the cluster head (CH) in all clusters. The network has been partitioned into 1-hop disjoint clusters. It ensures the secure routing. The proposed SFCP elects the trusted CH by the mechanism called fair cluster head selection which distinguishes trusted nodes from malicious nodes. Each node elects most trustworthy node of its 1-hop neighbors to be its CH which should not be a faulty degree claim for election process. In SFCP, each node i takes into account its degree $\deg_i$, an energy fairness factor F_i, and a security component. Equation (1) presents the clustering score of each node v_i

$$v_i = a \times \frac{\deg_i}{d_{\max}} + b \times \frac{F_i}{F_{\max}} + c_t \times \left(\frac{N_f}{\deg_i} - \frac{2}{3} \right) + d \times E, \quad (1)$$

where the coefficients a, b, c_t, and d satisfy the following:

$$a + b + c_t + d = 1. \quad (2)$$

$\deg_i$ is the number of nodes whose Euclidean distance from i is less than the radio range of i that is, the degree of a node in a network is the number of edges the node has to other nodes. N_f is the ratio of the number of neighbors in their neighbors list. F_i defines how many times i has previously served as CH, E is the Remaining energy level of each node i is calculated as follows

$$E = E_i - E_c, \quad (3)$$

where E_i is initial energy and E_c is consumed energy.

The secure cluster head selection algorithm describes the procedure for selecting cluster head (see Algorithm 1).

2.1. Byzantine Agreement. Byzantine agreement mechanism is used to solve the problem of multiple nodes reaching agreement in the presence of malicious node and message failures. This model demands global agreement to be reached for cluster formation and malicious identification. The routing paths are selected after a mutual exchange of control message amongst the neighbouring nodes (Figure 1). The consensus problem is assumed as a set of nodes n. A node is deemed correct if it does not fail and it correctly follows the protocol specification. Otherwise, the node is considered to be faulty. A consensus execution is initiated when every correct node n_i proposes an input value v_i and terminates after every correct node decided on a common value v. Consensus is defined by three properties such as validity, agreement, and termination. In validity, if every correct node proposed the same value v, then any nodes can decide v, In agreement, no two correct nodes decide differently and termination follows after the correct node decides. Agreement ensures the consistency in which some nodes decide a value v, and then no other node can decide a different value. Termination ensures that nodes must decide. These properties can be broken down into two different categories which include safety and liveness. Safety restricts the bad things that can happen in the system. Liveness ensures that good things eventually happen. Both the validity and agreement are safety properties while termination belongs to liveness property.

A fault occurs in a network when a node or link deviates from its correct behaviour. Within this context, faults are classified in two main classes: omissive faults and Byzantine faults. A fault is of omissive class if there is crash of a node and a message loss in a communication link. A fault is of Byzantine faults class if there is a compromised node that acts as malicious which sends incorrect values purposely and it influence other nodes to deviate from its properties of the protocol. More precisely, the nodes must reach an agreement that is admissible in the following sense:

(i) all normal nodes must agree on the same value v;

(ii) in case all the normal nodes have the same initial value, then v must be equal to this value.

It considered the following assumption that there are n nodes in the network in which 30% of nodes are considered

```
Initiate node state
Cluster formation        // assumption one hop clusters
NL = {n1, n2, n3,. . ., nn} // node's Neighbour Lists
Vi= {v1, v2, v3,. . ., vn} // Clustering variable Score
If (NL.size > max-cluster-size) then
      Truncate NL to max-cluster-size
      Sort NL based on Vi;
For each node i in NL:
      if (Nf/degi ≤ 2/3) then
         Neighbour is suspect node
         Calculate Vi with ct < 0 from (1)
         Classify whether normal or attacker by reputation system
      Else if (Nf/degi > 2/3) then
         Node is normal node
         Calculate Vi with ct > 0 from (1)
      Else if (degi > network-size) then
         Node is attacker node
         exclude node from decision
CH = neighbour node with maximum vi
```

Algorithm 1: Secure cluster head selection algorithm.

```
Input: NL = {n1, n2, n3,. . ., nn} // node's Neighbour Lists
Output: Classification of the Node
foreach Member Node in NL do
  if CH checks the Nf over degi is less than or equal to the threshold
     Suspect = node i
       if become good then
          Classified suspect node as normal
       else
         Penalized the suspect node by reducing its score vi
if advertised degree degi is equal to network size and the threshold is exceeded
       Classified node as attacker
       Node Excluded from network
if advertised ratio of degree degi greater than threshold
       Node is classified as normal
```

Algorithm 2: Byzantine agreement node classification algorithm.

as malicious. They make communication through messages. The malicious node can forge messages, send conflicting messages, and masquerade as other nodes. If a message from a normal node is lost or damaged, then the normal node is treated as attacker.

It classifies each node as normal, suspect, or attacker and allocates a different value of c_t for each type of node according to the byzantine agreement requirement to (4) as shown below

$$c_t = \begin{cases} > 0 & \text{if } \dfrac{N_f}{\deg_i} > \dfrac{2}{3}, \text{ 0.4 for dense, 0.2 for sparse} \\ < 0 & \text{if } \dfrac{N_f}{\deg_i} \leq \dfrac{2}{3} \\ = 0 & \text{if } \deg_i > \text{network size, } (a, b, d = 0)\,. \end{cases} \tag{4}$$

When the ratio of the search result N_f over $\deg_i$ is less than or equal to the threshold, the node i is classified as suspect. A suspect node is not immediately excluded from the network, but it is penalized by reducing its score v_i and also this node is motivated not to claim faulty degree. The node i is marked as attacker; if the advertised degree $\deg_i$ is equal to the network size and the threshold is exceeded, then it is immediately excluded from both the clustering and the routing procedures. The node is classified as normal if the advertised ratio of degree $\deg_i$ is greater than the threshold value (see Algorithm 2).

3. Result and Discussion

In this section, the performances of proposed SFCP are evaluated in network simulator (NS-2) [19]. The result of SFCP provides an effective malicious free routing in the presence of attacker nodes. In this implementation models,

Table 1: Simulation parameters.

Parameter	Value
Simulation area	900 m × 900 m
Simulation time	800 Sec
Number of nodes	50, 100, and 150
Transmission range	200 m
Movement model	Random waypoint model
Initial energy	100 joules
Packet size	512 bytes
Pause time	200 Sec

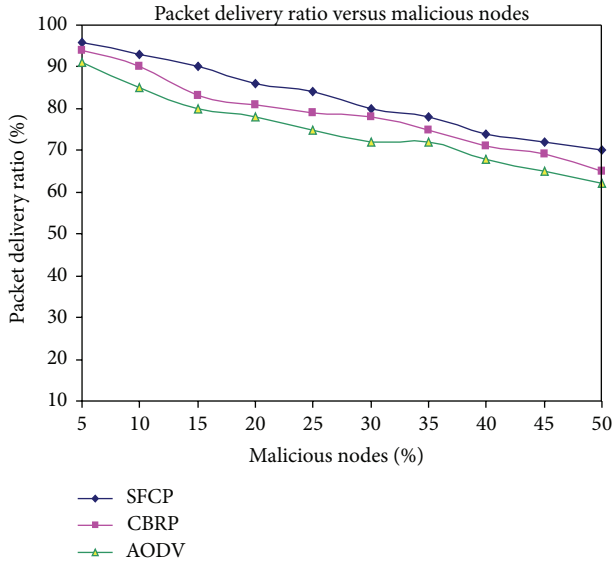


Figure 2: Packet delivery ratio versus malicious nodes.

we used the energy model and initially assigned 60 to 100 joules to each node. Besides, different numbers of nodes that vary from 50 to 150 are deployed in an area of 900 × 900 m. Table 1 summarizes the simulation parameters. The attacker nodes were made to drop the messages during routing phase.

The following sections discussed the results and observations of the proposed SFCP along with AODV and CBRP in terms of packet delivery ratio, routing and cluster overhead, and routing latency.

3.1. Packet Delivery Ratio (PDR) in the Presence of Malicious Nodes. In this simulation, the impacts of malicious nodes are evaluated by measuring the PDR. The numbers of nodes used in this simulation are 50 and malicious nodes are varied from 0 to 40%. In Figure 2, notice that SFCP maintained much higher PDR about 80% of the data packets when 40% of nodes are misbehaving and other protocols decreased the PDR. This is due to the fact that CBRP and AODV do not have a mechanism to detect misbehaving nodes. When numbers of malicious nodes are increased, it becomes harder to find malicious free routes from the source to the destination. SFCP could detect misbehaving or faulty claim nodes continuously changing with network mobility and also

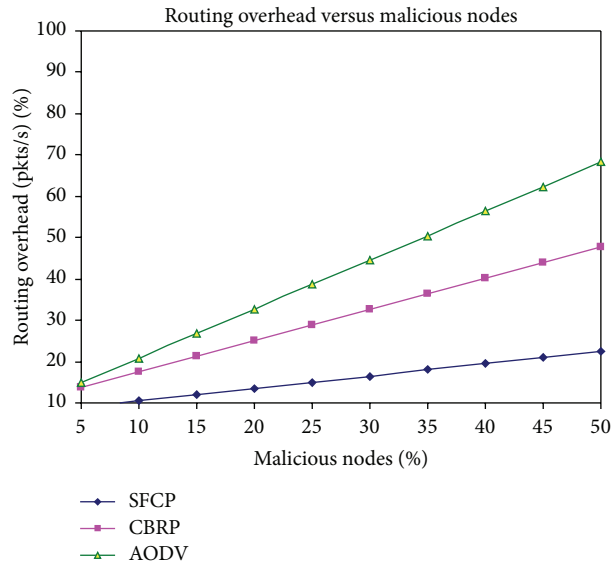


Figure 3: Routing overhead versus malicious nodes.

it elects misbehaving free cluster heads for routing using Byzantine Agreement node classification mechanism.

3.2. Routing and Clustering Overhead. In simulation, clustering and routing overhead are analysed for SFCP, AODV, and CBRP while varying the malicious nodes. The numbers of nodes are set to 50. Figure 3 shows that minimal routing overhead is caused by SFCP and CBRP than AODV. SFCP and CBRP are greatly reducing network traffic because of their cluster based architecture and limited exchange of routing control message. On the other hand, route request of AODV is originated from the source; then, it is flooded throughout the network until destination is reached which cause more routing overhead. In addition to that, the number of malicious node increases, and the overhead is also increased because malicious nodes drop packets. The result shows that SFCP caused minimal routing overhead when there are 40% of malicious nodes present in the network.

3.3. Routing Latency. Figure 4 shows that SFCP achieved higher routing latency than the other two protocols. SFCP is considered the Byzantineagreement mechanism to classify whether the node is normal or attacker. The normal nodes only allow participating in routing and cluster head selection and minimum number of nodes participated in the routing phase. Figure 4 shows that the numbers of malicious nodes are increased; routing latency is decreased for AODV and CBRP because malicious nodes drop the packets.

4. Conclusion

In this paper, the proposed SFCP avoided the impact of compromising nodes by selecting the secure and energy efficient node as a cluster head. SFCP can effectively classify the malicious nodes and prevent these nodes from faulty degree claim in cluster head selection. The SFCP used

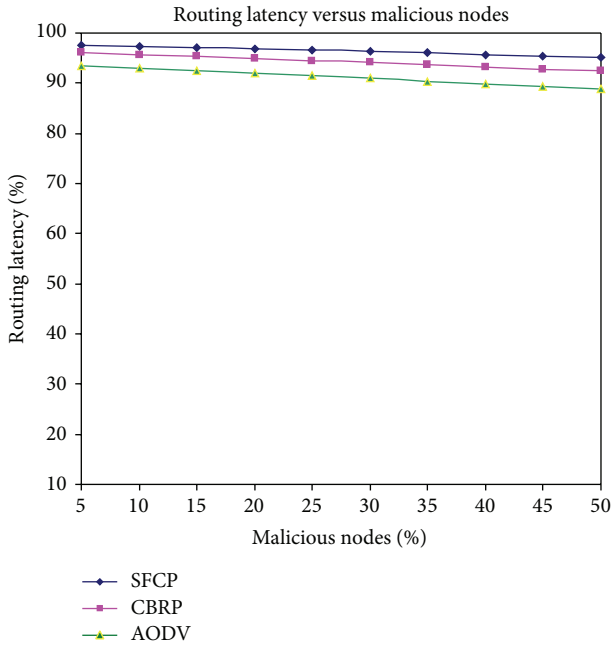


Figure 4: Routing latency versus malicious nodes.

the Byzantineagreement mechanism to mitigate selfishness which made the network more attack resistant. SFCP could detect malicious nodes that get isolated from the networks. Simulation results show that SFCP achieved better packet delivery ratio and routing latency with less routing and cluster overhead than AODV and CBRP.

Conflict of Interests

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

References

[1] I. Chlamtac, M. Conti, and J. J. Liu, "Mobile Ad Hoc networking: imperatives and challenges," *Ad Hoc Networks*, vol. 1, no. 1, pp. 13–64, 2003.

[2] C. Tselikis, S. Mitropoulos, N. Komninos, and C. Douligeris, "Degree-based clustering algorithms for wireless Ad Hoc networks under attack," *IEEE Communications Letters*, vol. 16, no. 5, pp. 619–621, 2012.

[3] J. Y. Yu and P. H. J. Chong, "A survey of clustering schemes for mobile Ad Hoc networks," *IEEE Communications Surveys and Tutorials*, vol. 7, no. 1, pp. 32–48, 2005.

[4] M. Bechler, H. Hof, D. Kraft, F. Pählke, and L. Wolf, "A cluster-based security architecture for Ad Hoc networks," in *Proceedings of the 23rd AnnualJoint Conference of the IEEE Computer and Communications Societies (INFOCOM '04)*, vol. 4, pp. 2393–2403, March 2004.

[5] W. J. Adams and N. J. Davis, "Toward a decentralized trust-based access control system for dynamic collaboration," in *Proceedings of the 6th Annual IEEE System, Man and Cybernetics Information Assurance Workshop (SMC '05)*, pp. 317–324, June 2005.

[6] A. Boukerche and Y. Ren, "A security management scheme using a novel computational reputation model for wireless and mobile Ad Hoc networks," in *Proceedings of the 5th ACM International Symposium on Performance Evaluation of Wireless Ad-Hoc, Sensor, and Ubiquitous Networks (PE-WASUN '08)*, pp. 88–95, October 2008.

[7] R. Li, J. Li, P. Liu, and H. Chen, "An objective trust management framework for mobile Ad Hoc networks," in *Proceedings of the IEEE 65th Vehicular Technology Conference (VTC '07)*, pp. 56–60, April 2007.

[8] G. Theodorakopoulos and J. S. Baras, "Trust evaluation in ad-hoc networks," in *Proceedings of the ACM Workshop on Wireless Security (WiSe '04)*, pp. 1–10, October 2004.

[9] T. Jiang and J. S. Baras, "Ant-based adaptive trust evidence distribution in MANET," in *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops*, pp. 588–593, March 2004.

[10] Z. Yan, P. Zhang, and T. Virtan, "Trust evaluation based security solution in Ad Hoc networks," in *Proceedings of 7th Nordic Workshop on Secure IT Systems*, 2003.

[11] A. A. Pirzada and C. McDonald, "Establishing trust in pure ad-hoc networks," in *Proceedings of the 27th Australasian Conference on Computer Science (ACSC '04)*, vol. 26, pp. 47–54, 2004.

[12] T. Ghosh, N. Pissinou, and K. Makki, "Collaborative trust-based secure routing in multihop Ad Hoc networks," in *NETWORKING 2004. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications: Proceedings of the 3rd International IFIP-TC6 Networking Conference, Athens, Greece, May 9–14, 2004*, vol. 3042 of *Lecture Notes in Computer Science*, pp. 1446–1451, 2004.

[13] T. Ghosh, N. Pissinou, and K. Makki, "Towards designing a trusted routing solution in mobile Ad Hoc networks," *Mobile Networks and Applications*, vol. 10, no. 6, pp. 985–995, 2005.

[14] F. Milan, J. J. Jaramillo, and R. Srikant, "Achieving cooperation in multihop wireless networks of selfish nodes," in *Proceedings of the Workshop on Game Theory for Communications and Networks (GameNets '06)*, October 2006.

[15] H. Safa, H. Artail, and D. Tabet, "A cluster-based trust-aware routing protocol for mobile Ad Hoc networks," *Wireless Networks*, vol. 16, no. 4, pp. 969–984, 2010.

[16] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: a weighted clusrering algorithm for mobile Ad Hoc networks," *Journal of Cluster Computing*, vol. 5, no. 2, pp. 193–204, 2002.

[17] P. Chatterjee, I. Sengupta, and S. K. Ghosh, "STACRP: a secure trusted auction oriented clustering based routing protocol for MANET," *Cluster Computing*, vol. 15, pp. 303–320, 2012.

[18] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: a weighted clustering algorithm for mobile Ad Hoc networks," *Journal of Cluster Computing*, vol. 5, no. 2, pp. 193–204, 2002.

[19] NS-2 Simulator, 2006, http://www.isi.edu/nsnam/ns.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 810782, 6 pages
http://dx.doi.org/10.1155/2014/810782

Research Article

Assessment on Knowledge Network Sharing Capability of Industrial Cluster Based on Dempster-Shafer Theory of Evidence

Shengli Dai[1,2] and Hailin Zhang[3]

[1] *School of Public Management, Central China Normal University, 152 Luoyu Road, Wuhan 430079, China*
[2] *Local Governance and Local Development Research Centre of Hubei, Central China Normal University, 152 Luoyu Road, Wuhan 430079, China*
[3] *School of Urban and Environmental Sciences, Central China Normal University, 152 Luoyu Road, Wuhan 430079, China*

Correspondence should be addressed to Hailin Zhang; hailzhang@sina.com

Received 22 December 2013; Accepted 25 February 2014; Published 24 March 2014

Academic Editors: H. R. Karimi, X. Yang, Z. Yu, and W. Zhang

Based on Theory of Evidence and reviewing research papers concerned, a concept model of knowledge sharing network among industrial cluster firms, which can be applied to assess knowledge sharing capacity, has been built. Next, the authors create a set of assessment index systems including twelve subindexes under four principle indexes. In this study, ten experts in the same field were invited to score all the indexes of knowledge sharing capacity concerning one certain industrial cluster. The research result shows relatively high knowledge network sharing capacity among the certain industrial cluster firms. Another conclusion is that the assessment method with Theory of Evidence is feasible to conduct such a research.

1. Introduction

In recent years, it has become increasingly apparent that cluster economy, accompanied by industrial cluster boom, is one important form of the world economy development and presents a state that develops quickly. In the study field of industrial cluster, operation of knowledge network is focused on by some researchers.

The relationship of cooperation between enterprises and knowledge exchange attracts more and more attention. In the study of process of knowledge transfer, emphasis is widely put on knowledge characteristic (e.g., recessive, tacit-codified, individual-social, appropriable-exclusive, and divisible-indivisible) [1]. There is also much research, which contributes to knowledge network of industrial cluster. The Framework of Knowledge Networks comprises the following components: actors individuals, groups, and organizations; relationships between actors, which can be categorized by form, content, and intensity; resources which may be used by actors within their relationships, and institutional properties, including structural and cultural dimensions such as control mechanisms, standard operating procedures, norms and rules, and communication patterns [2]. Regional clusters depend on the networks that arise from reciprocal linkages among colocated organizations, while physical proximity among firms can alter the nature of information and resource flows through networks [3]. In all industries, knowledge can flow through the linkage border and the effect of knowledge flow is enhanced by spatial proximity between the actors taking part in the exchange [4]. The focal firms and their organized networks within industrial districts, being the driving force of cluster innovation, play a crucial role in the process of knowledge creation and transfer [5]. Knowledge alliance, one kind of strategic alliance similar to knowledge network, means a group of firms entering into voluntary arrangements that involve knowledge creation, transfer and exchange [6]. Alliances may serve different purposes from knowledge alliance formation. However, rather than using alliances to acquire capabilities, scholars suggest that firms use interfirm collaboration to gain access to other firms' capabilities, supporting more focused, intensive exploitation of existing capabilities [7]. Based on heterogeneity of knowledge acquired from accumulative research, different firms can develop various capabilities [8]. Some researches concerning

knowledge transfer and sharing have been done. The study of Ahuja and Giuliani shows the existing relation between structural properties of social networks and learning and innovation output [9, 10]. Ahuja's study shows that direct and indirect ties between a firm's partners both have a positive impact on innovation but that the impact of indirect ties is moderated by the number of a firm's direct ties. More brokerage opportunities in network firms relate to better innovative performance, because actors in a network rich in structural holes will be able to access novel information from remote parts of the network and exploit that information to their advantage. Hansen's research argues that indirect ties in network firms are conducive to knowledge transfer and focal firms usually are the bridge of the connections [11]. In the study of the Boston Biotechnology Community innovative activities, Owen-Smith and Powell emphasize that focal firms are the organizer of the network knowledge and fundamentally alter the flow of information through a network [12, 13].

Though there are many insights on explaining the process of knowledge transfer and innovation, more emphasis was put on structural character of knowledge network. Because of much uncertainty, many methods have been applied to evaluate the capacity of knowledge networks sharing, for example, Analytic Hierarchy Process, Fuzzy Theory, Grey Theory, Neutral Network Analysis, Data Envelop Analyze, Factor Analysis, and Bayesian Network. In our view, the existing researches on assessing the capacity of knowledge networks sharing have a number of important weaknesses and constraints, especially on assessment methodologies and unclear attribute of the evaluation objects. In this research, to begin with, one new evaluation index system concerning the evaluation object will be established. We attempt to assess the capacity of knowledge networks sharing among cluster industries based on another powerful method—Dempster-Shafer Theory of Evidence.

There are quite a few models concerning knowledge networks sharing among cluster industries. Szulanski argues that knowledge transfer process includes four stages—initiation, implementation, ramp-up, and integration [14]. In Xiao's opinion, there are several categories of Knowledge Flows in Knowledge Networks: technical corporation, talent flow, new technology transfer, transfer of patent right, market research, and informal exchange [15]. Zhang constructs a cluster model of enterprise knowledge networks based on social network tools and argues that knowledge base, recessive knowledge center, knowledge transfer center, knowledge innovation center, and knowledge expansion center exist among cluster enterprises [16]. Above, researches present a reasonable analysis on operation mechanism of knowledge network sharing among cluster industries and draw some valuable conclusions. In authors' view, much efficient discussion on the knowledge sharing process is required. In the knowledge sharing process among cluster industries, knowledge from knowledge source, combined with knowledge transfer carrier, finally transfers to knowledge accepter via certain transfer path. The knowledge source includes universities, institutions, financial institutions, administration consulting company, administration and industry association, upstream firms, downstream firms, and competitive firms as shown in Figure 1.

2. Computational Methods for Evidence Theory

Dempster-Shafer theory of evidence was put forward on early 1976 by Dempster and developed by Shafer [17]. Generally, evidences are conclusions of one specific research and observation, which is one part of experience and knowledge. Based on evidences, initial allocations are created, which means to calculate the degrees of support on each proposition. In actual study, evidences available can be used to calculate collective contribution to every proposition and this method can effectively solve the problem of uncertainty in knowledge network sharing assessment.

Generally, suppose that one sampling space is called a frame of discernment θ, one proposition $H_q(q = 1, 2, \ldots, z)$ is subset of set θ, and H_q is individually independent. In the study, some definitions of belief allocation, belief function, and focal elements are brought forward. Evidence combination rules are introduced and knowledge network sharing capability of cluster industries is assessed.

Given a question of interest, let θ be a finite set of possible answers to the question, called a frame of discernment, and let 2^θ be the set of all subsets of θ:

$$2^\theta = \{A \mid A \subseteq \theta\}. \tag{1}$$

The subset A includes as special cases the empty set φ and the full set θ. It represents a statement or proposition that the truth lies in A. A real function over the subsets Bel: $2^\theta \rightarrow [0, 1]$ is called a belief function if and only if it satisfies the following three axioms:

$$\begin{aligned} &\text{(a)}\ \ m(f) = 0 \\ &\text{(b)}\ \ \sum_{A \subset \theta} m(A) = 1, \quad \forall A \subseteq \theta, \end{aligned} \tag{2}$$

where m is defined as basic probability on frame of discernment θ *and* $m(A)$ *is called Proposition A's basic probability number.* The vacuous belief function is a simple support function with focus θ. If Bel is a simple support function with focus $F = \theta$, then $m(F) = s, m(\theta) = 1-s$, and m is 0 elsewhere. Thus, a simple support function invests all of our committed belief on the disjunction represented by its focus, F, and all our uncommitted belief on θ. Consider

$$\text{Bel}(A) = \sum_{B \subset A} m(B) \quad (\forall A \subset \theta). \tag{3}$$

The subsets A of θ such that $m(A) > 0$ are called the focal elements of the belief function, and their union is called its core.

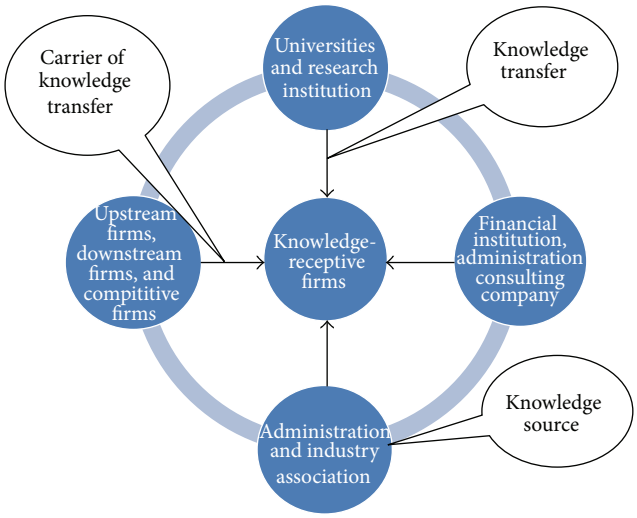


Figure 1: Flowchart of knowledge network sharing among cluster industries.

Based on the definitions above, evidence combination rules can be derived. Let m_1 and m_2 be basic probability assignments on the same frame θ. The function is defined as

$$m_{12}(f) = 0,$$
$$m_{12}(A) = \frac{1}{1-k}\sum_{X\cap Y=A} m_1(X)\, m_2(Y), \quad A \neq f, \tag{4}$$

where m_{12} is a set function over the frame of discernment θ and $K = (1/(1-k))\sum_{X\cap Y=f} m_1(X)m_2(Y)$ is conflict factor.

This formulation (4), called Dempster's rule of combination, namely, $m_{12} = m_1 \cdot m_2$, represents the degrees of support by two independent evidences.

In the assessment process, firstly, hierarchy factors' values of model are scored by experts and decision makers and basic probabilities are obtained. Secondly, the assessment is completed based on the combination rules of evidence. There are five steps to assess knowledge network sharing capability among cluster industries via Evidence Theory. Consider the following.

(a) Based on analytic hierarchy process (AHP), the numerical weight is derived for each element of any hierarchy. Corresponding basic probability ($\beta_{H_q(e^i_{jk})}$) of the lowest hierarchy elements (e^i_{jk}) over set H_q ($q = 1, 2, \ldots, z$) is acquired by experts and decision makers.

(b) Basic probability of each element of each hierarchy is computed. The elements with greatest weight is called core element and their probability over set H_q is $m(H_q \mid e^i_{jk}) = \alpha_{jk}\beta_{H_q}(e^i_{jk})$, while probability of the lowest elements over set H_q is $\beta_{H_q(e^i_{jk})}$, where α_{jk} is preference coefficient reflecting experts' attitude to core elements. α_{jk}'s value ranges over $0.9 \le \alpha_{jk} \le 1$. The greater is the α_{jk}'s value, the more important is the core element. As to noncore elements, their basic probability over set H_q is expressed as $m(H_q \mid e^i_{jk}) = (w_{jk}/w_{jm})\alpha_{jk}\beta_{H_q}(e^i_{jk})$, where w_{jm} is the weight of core element and $\alpha_{jk}(w_{jk}/w_{jm})$ is the element e^i_{jk}'s normalized weight. And the basic probability of the element e^i_{jk} under completely uncertain is expressed as $m(H_q \mid e^i_{jk}) = 1 - \sum_{q=1}^{z} m(H_q \mid e^i_{jk})$.

(c) Suppose that there is an element set $I_{(n)} = \{e_{i1}, e_{i2}, \ldots, e_{in}\}$ within subsystem $I_{(n)} = \{e_{i1}, e_{i2}, \ldots, e_{in}\}$. Based on the combination rules, complex set function is generated as $m^q_{I_{(n)}} = m(H_q \mid I_{(n)})$ and $m^\theta_{I_{(n)}} = m(H_q \mid I_{(n)})$, where $m^q_{I_{(n)}}$ represents the basic probability of all elements in the set $I_{(n)}$ over set H_q and $m^\theta_{I_{(n)}}$ represents the basic probability of the elements in the set $I_{(n)}$ under completely uncertainty. When $n = 2$, $I_{(2)} = \{e_{i1}, e_{i2}\}$, equations will be induced by (4) as

$$m^q_{I_{(2)}} = k_{I_{(2)}}\left[m^q_1 m^q_2 + m^q_1 m^\theta_2 + m^\theta_1 m^q_2\right], \quad q = 1, 2, \ldots, z$$
$$m^\theta_{I_{(2)}} = k_{I_{(2)}} m^\theta_1 m^\theta_2 \tag{5}$$
$$k_{I_{(2)}} = \left(1 - \sum_{t=1}^{z}\sum_{q\neq t} m^t_1 m^q_2\right)^{-1}.$$

Since $m^q_{I_{(1)}} = m^q_1 (q = 1, 2, \ldots, z)$ and $m^\theta_{I_{(1)}} = m^\theta_1$, general term formula of recurrent sequence of $I_{(p+1)} = \{e_{i1}, e_{i2}, \ldots, e_{ip}, e_{i(p+1)}\}$ $(p = 1, 2, \ldots, z-1)$ is obtained through comprehensive induction as formulation (5) as

$$m^q_{I_{(p+1)}} = k_{I_{(p+1)}}\left[m^q_{I_{(p)}} m^q_{p+1} + m^q_{I_{(p)}} m^\theta_{p+1} + m^\theta_{I_{(p)}} m^q_{p+1}\right], \quad q = 1, 2, \ldots, z, \tag{6}$$
$$m^\theta_{I_{(p+1)}} = k_{I_{(p+1)}} m^\theta_{I_{(p)}} m^\theta_{p+1},$$
$$k_{I_{(p+1)}} = \left(1 - \sum_{t=1}^{z}\sum_{q\neq t} m^t_{I_{(p)}} m^q_{p+1}\right)^{-1}. \tag{7}$$

In the formulation (7), when $p = n-1$, the set function of all the lowest hierarchy elements in subsystem e_i is obtained

$$m\left(\{H_q\}\right) = m^q_{I_{(n)}}, \quad q = 1, 2, \ldots, z,$$
$$m(H) = m^\theta_{I_{(n)}} \tag{8}$$

And $m(A) = 0$, for all $A \neq \{H_q\}$, $q = 1, 2, \ldots, z$, $A \neq H$.

(d) In the same way, the whole assessment system's set function can be obtained from formulations (6), (7), and (8); namely, the belief assignment of the whole system over set assessment set $H = \{H_1, H_2, \ldots, H_z\}$ $(q = 1, 2, \ldots, z)$ is m^q $(q = 1, 2, \ldots, z)$.

(e) Finally, formulation $S = \sum_{q=1}^{z} m^q p(H_q)$ is used to compute knowledge network sharing capability among cluster industries. The output is the result of evidential reasoning [18].

3. Construction of Assessment Indicator

In authors' view, there are several direct factors that affect the capability of knowledge network sharing among cluster industries. According to the model in the flowchart above, knowledge source, the pathway of knowledge transfer, carrier of knowledge transfer, and knowledge recipient are four prominent factors upon the capability of knowledge network sharing among cluster industries. Generally, the knowledge sharing capability of knowledge source is affected by vigor of knowledge, knowledge quantity, interest motive of knowledge sharing, and knowledge compatibility. The pathway of knowledge transfer is affected by remunerative transfer of new technology, talent flow, and technical corporation. Carrier of knowledge transfer is affected by competency of staff and knowledge sharing technology. Knowledge recipient is affected by knowledge deficiency, knowledge absorptive capacity, and willingness of receiving knowledge. Based on the factors above, the assessment index system is built. V_i represents the indicator and w_i represents the corresponding indicator's weight. Ten experts are invited to score each indicator and scores are listed in Table 1.

4. Weight of Assessment Indicator

In the aspect of weight, fuzzy clustering analysis is applied to avoid experts' subjective factors. Essentially, fuzzy clustering analysis is one classification method, which divides data elements into classes or clusters so that items in the same class are as similar as possible, and items in different classes are as dissimilar as possible. Under different supposed thresholds λ, different weights can be calculated as below.

4.1. Data Normalization. Given a universe $U = \{u_1, u_2, \ldots, u_m\}$, any index u_i is expressed by membership degree of n samples; namely, $u_i = \{X_{i1} = \mu_1(u_i), X_{i2} = \mu_2(u_i), \ldots X_{ij} = \mu_j(u_i), \ldots X_{in} = \mu_n(u_i)\}$. Then, the initial assessment matrix can be obtained,

$$X = \left\{X_{ij} = \mu_j\left(u_i\right)\right\} \quad (i = 1, 2, \ldots m; j = 1, 2, \ldots n). \quad (9)$$

And the matrix of normalized data (Y) is derived, $Y = \{Y_{ij}\}_{m\times n}$, where $Y_{ij} = X_{ij}/\sum_{j=1}^{n} X_{ij}$ is normalization data of x_{ij} and x_{ij} is the simple value of the variable i measured in the level j.

4.2. Computation of Fuzzy Similar Matrix. Angle cosine method is applied to obtain fuzzy similar matrix $R = (r_{ij})$, where r_{ij} represents relation coefficient between any two observed variables,

$$r_{ij} = \frac{\sum_{k=1}^{m} X_{ik} X_{jk}}{\sqrt{\sum_{k=1}^{m} \left(X_{ik}\right)^2 \left(X_{jk}\right)^2}}, \quad (10)$$

where X_{ik} and X_{jk}, respectively, represent the average scores assessed by ten experts, which range over $0 \le X_{jk} \le 1$.

4.3. Construction of Fuzzy Equivalence Matrix. Convolution computation on fuzzy similar matrix is conducted after iterations until $R^n R = R^n$ and $R^k = R2^k$ appear, and then fuzzy classification matrix is obtained.

4.4. Clustering. Given different threshold λ values, Boolean matrixes, which relate to the significance of the indictor, are obtained by computation of fuzzy classification matrix. Generally, the classification categories rise with the increase of λ value. In order to distinguish the significance of the assessment indicators, greater λ value should be chosen so that sequence and classification can be well done. Based on the different grades of λ values, weights of the assessment indicators are determined after data normalization.

5. Empirical Study

As described above, ten experts anticipate to score each assessment indicator and scores are listed in Table 1. Fuzzy clustering analysis is applied to determine the weight of each assessment indicator as $W = (W_1, W_2, W_3, W_4, W_5) = (0.4, 0.2, 0.1, 0.3, 0)$, $W_1 = (W_{11}, W_{12}, W_{13}, W_{14}) = (0.35, 0.25, 0.2, 0.2)$, $W_2 = (W_{21}, W_{22}, W_{23}) = (0.4, 0.3, 0.3)$, $W_3 = (W_{31}, W_{32}) = (0.6, 0.4)$, and $W_4 = (W_{41}, W_{42}, W_{43}) = (0.3, 0.5, 0.2)$. *Let* preference coefficient be $\alpha = 0.9$; then mass function about the indicator V_{11} over the fuzzy set H is computed. Vigor of knowledge is the core indicator of knowledge source and outputs are $m(H_1/V_{11}) = 0.9 \times 0.2 = 0.18$, $m(H_2/V_{11}) = 0.9 \times 0.4 = 0.36$, $m(H_3/V_{11}) = 0.9 \times 0.1 = 0.09$, $m(H_4/V_{11}) = 0.9 \times 0.3 = 0.27$, and $m(H_5/V_{11}) = 0.9 \times 0 = 0$, while the nonassigned confidence is $m(H/V_{11}) = 0.1$. In the same way, mass function values about noncore indicator V_{12} over the fuzzy set H are $m(H_1/V_{12}) = 0.9 \times 0.25/0.35 \times 0.3 = 0.1929$, $m(H_2/V_{12}) = 0.9 \times 0.25/0.35 \times 0.2 = 0.1286$, $m(H_3/V_{12}) = 0.9 \times 0.25/0.35 \times 0.3 = 0.1929$, $m(H_4/V_{12}) = 0.9 \times 0.25/0.35 \times 0.1 = 0.0643$, $m(H_5/V_{12}) = 0.9 \times 0.25/0.35 \times 0.1 = 0.0643$, and $m(H/V_{12}) = 0.357$. Mass function about other indicators over the fuzzy set H can be calculated as well. Finally, mass matrix about the indicator over the fuzzy set H is expressed as

$$\begin{bmatrix} 0.1800 & 0.3600 & 0.0900 & 0.2700 & 0 & 0.1000 \\ 0.1929 & 0.1286 & 0.1929 & 0.0643 & 0.0643 & 0.3570 \\ 0.2057 & 0.1029 & 0.0514 & 0.1543 & 0 & 0.4857 \\ 0.0514 & 0.1543 & 0.2057 & 0.1029 & 0 & 0.4857 \end{bmatrix}. \quad (11)$$

Complex mass matrix about knowledge source V_1 is derived from evidence combination computation by following computational steps (c) and (d). In the same way, mass matrix about V_2, V_3, and V_4 over the fuzzy set can create a new mass matrix as

$$\begin{bmatrix} 0.25 & 0.27 & 0.23 & 0.23 & 0.02 \\ 0.27 & 0.26 & 0.20 & 0.13 & 0.07 \\ 0.45 & 0.25 & 0.20 & 0.10 & 0 \\ 0.23 & 0.37 & 0.23 & 0.10 & 0.06 \end{bmatrix}. \quad (12)$$

Table 1: Assessment indicator system of knowledge sharing capability in knowledge network.

Principle tier	Subprinciple tier	Assessment class				
		High	Relatively high	Moderate	Relatively low	Low
Assessment index system of knowledge sharing capability in knowledge network (V)						
Knowledge Source (V_1, W_1)	Vigor of knowledge (V_{12}, W_{12})	2	4	1	3	0
	Knowledge quantity (V_{12}, W_{12})	3	2	3	1	1
	Interest motive of knowledge sharing (V_{12}, W_{12})	4	2	1	3	0
	Knowledge compatibility (V_{12}, W_{12})	1	3	4	2	0
Path of knowledge transfer (V_2, W_2)	Remunerative transfer of new technology (V_{21}, W_{21})	3	2	3	2	0
	Talent flow (V_{22}, W_{22})	1	4	2	2	1
	Technical corporation (V_{23}, W_{23})	4	2	1	2	1
Carrier of knowledge transfer (V_3, W_3)	Competency of staff (V_{31}, W_{31})	5	2	3	0	0
	Knowledge sharing technology (V_{32}, W_{32})	4	3	1	2	0
Knowledge recipient (V_4, W_4)	Knowledge deficiency (V_{41}, W_{41})	3	3	3	1	0
	Knowledge absorptive capacity (V_{42}, W_{42})	3	2	4	0	1
	Willingness of receiving knowledge (V_{43}, W_{43})	1	6	0	2	1

Mass matrix about V is also obtained by the following computational steps (c) and (d): $m^w(H_1/V) = 0.27$, $m^w(H_2/V) = 0.30$, $m^w(H_3/V) = 0.20$, $m^w(H_4/V) = 0.16$, $m^w(H_5/V) = 0.04$, and $m^w(H_5/V) = 0.03$.

Studies' results show that experts' high support degree on knowledge network sharing capability among cluster industries is 27%, and relatively high support degree, moderate support degree, relatively low support degree, and low support degree are, respectively, 30%, 20%, 16%, and 4%. The nonassigned confidence is 3%. Based on the maximum membership degree principle, knowledge network sharing capability belongs to relatively high degree.

6. Conclusion

Due to asymmetric information, less objective data, and cognitive biases of experts, systematic assessment methods have not dealt with uncertain matters. Evidence Theory is an effective mathematical tool to solve inaccurate and uncertain problems while distinguishing uncertain and unknown facts. Under Dempster's evidence combination rules, experts' opinions with different probability distribution can be operatively synthesized, which bring a new insight to carry out specific assessment.

Conflict of Interests

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

References

[1] A. Amin and P. Cohendet, *Architectures of Knowledge: Firms, Capabilities, and Communities*, Oxford University Press, Oxford, UK, 2004.

[2] A. Seufert, G. von Krogh, and A. Bach, "Towards knowledge networking," *Journal of Knowledge Management*, vol. 3, no. 3, pp. 180–190, 2002.

[3] K. B. Whittington, J. Owen-Smith, and W. W. Powell, "Networks, propinquityand innovation in knowledge-intensive industries," *Administrative Science Quarterly*, vol. 54, no. 3, pp. 90–122, 2009.

[4] P. Maskell and A. Malmberg, "Localized learning and industrial competitiveness," *Cambridge Journal of Economics*, vol. 23, no. 2, pp. 167–186, 1999.

[5] C. Boari and A. Lipparini, "Networks within industrial districts: organising knowledge creation and transfer by means of moderate hierarchies," *Journal of Management and Governance*, vol. 3, no. 4, pp. 339–360, 1999.

[6] A. C. Inkpen and E. W. K. Tsang, "Social capital networks, and knowledge transfer," *Academy of Management Review*, vol. 30, no. 1, pp. 146–165, 2005.

[7] D. C. Mowery, J. E. Oxley, and B. S. Silverman, "Strategic alliances and inter-firm knowledge transfer," *Strategic Management Journal*, vol. 17, pp. 77–91, 1996.

[8] G. Dosi, R. R. Nelson, and S. G. Winter, *The Nature and Dynamics of Organizational Capabilities*, Oxford University Press, Oxford, UK, 2000.

[9] G. Ahuja, "Collaboration networks, structural holes and innovation: a longitudinal study," *Administrative Science Quarterly*, vol. 45, no. 3, pp. 425–456, 2000.

[10] E. Giuliani, "The selective nature of knowledge networks in clusters: evidence from the wine industry," *Journal of Economic Geography*, vol. 7, no. 2, pp. 139–168, 2007.

[11] M. T. Hansen and B. von Oetinger, "Introducing T-shaped managers. Knowledge management's next generation," *Harvard Business Review*, vol. 79, no. 3, pp. 106–116, 2001.

[12] L. Orsenigo, F. Pammolli, and M. Riccaboni, "Technological change and network dynamics: lessons from the pharmaceutical industry," *Research Policy*, vol. 30, no. 3, pp. 485–508, 2001.

[13] J. Owen-Smith and W. W. Powell, "Knowledge networks as channels and conduits: the effects of spillovers in the boston biotechnology community," *Organization Science*, vol. 15, no. 1, pp. 5–21, 2004.

[14] G. Szulanski, "The process of knowledge transfer: a diachronic analysis of stickiness," *Organizational Behavior and Human Decision Processes*, vol. 82, no. 1, pp. 9–27, 2000.

[15] D. Xiao, X. Gu, and X. Peng, "On knowledge flows in knowledge networks: a perspective of embeddedness," *Journal of Intelligence*, no. 8, pp. 116–125, 2009.

[16] H. Zhang, "Construction of knowledge networks among cluster enterprises based on the social networks tools," *Science and Technology Management Research*, no. 2, pp. 193–196, 2011.

[17] D.-L. Zheng and C. Du, "Evidence theory application in performance review of aviation equipment," *Journal of Traffic and Transportation Engineering*, vol. 4, no. 3, pp. 114–116, 2004.

[18] S. Ding, R. Shi, and H. Shi, "Airline safety system risk assessment based on evidence theory," *Journal of Transportation Systems Engineering and Information Technology*, no. 4, pp. 77–82, 2007.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 914907, 8 pages
http://dx.doi.org/10.1155/2014/914907

Research Article

Modeling of Information Diffusion in Twitter-Like Social Networks under Information Overload

Pei Li, Wei Li, Hui Wang, and Xin Zhang

College of Information Systems and Management, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Pei Li; peili@nudt.edu.cn

Received 29 December 2013; Accepted 24 February 2014; Published 23 March 2014

Academic Editors: H. R. Karimi, X. Yang, Z. Yu, and W. Zhang

Due to the existence of information overload in social networks, it becomes increasingly difficult for users to find useful information according to their interests. This paper takes Twitter-like social networks into account and proposes models to characterize the process of information diffusion under information overload. Users are classified into different types according to their in-degrees and out-degrees, and user behaviors are generalized into two categories: generating and forwarding. View scope is introduced to model the user information-processing capability under information overload, and the average number of times a message appears in view scopes after it is generated by a given type user is adopted to characterize the information diffusion efficiency, which is calculated theoretically. To verify the accuracy of theoretical analysis results, we conduct simulations and provide the simulation results, which are consistent with the theoretical analysis results perfectly. These results are of importance to understand the diffusion dynamics in social networks, and this analysis framework can be extended to consider more realistic situations.

1. Introduction

Research on social networks has received remarkable attention in the past decade, since social networks provide numerous features to encourage information sharing among users. Among the existing social networks, microblogging services (e.g., Twitter and Facebook) have impressively become more and more popular, which provide new communication methods for people to stay connected with their friends. The use of microblogging for lightweight communication makes it important candidate media for informal communication.

Twitter is arguably one of the most well-known microblogging platforms currently available, which is used by hundreds of millions of people all over the world. Twitter users update their daily life activities by computers or mobile phones, so as to broadcast things that happen in their daily lives, such as what they are reading, thinking, and experiencing. Users declare the persons they are interested in by the action *following*. For the case when user A follows user B, we say user A is one of user B's *followers*, and user B is one of user A's *followees*. Twitter users are allowed to post short messages (up to 140 characters) which are so-called *tweets* and also forward messages which are so-called *retweets*. Each user has a web form, where all his/her followees' new messages (both tweets and retweets) are arranged in a reverse-chronological order. So after logging in, a user will get noticed if his/her followees have posted new messages.

Essentially, relationships in Twitter are asymmetric, since a user who is followed by another user does not necessarily have to reciprocate by following him/her back. Some social networks adopt symmetric relationships. For example, in Facebook, a relationship is established when a request for friendship is accepted by a user, which adds both on each other's contact lists. If one user removes the other, the relationship is broken. Therefore, an important difference between these two social networks is that the network of Twitter is directed, while that of Facebook is undirected. Having noticed the increasing popularity of Twitter, we take Twitter-like social networks into account in this paper.

Compared with traditional media such as newspapers and television, social networks allow creation and exchange of user-generated contents, while every user can produce and distribute messages. This results in an explosively growing amount of information and makes many social networks become increasingly information saturated. Besides, due to the potential for marketing and advertising, Twitter and

some other social networks are considered to be efficient approaches to stimulate the awareness and adoption of products or services. One important benefit of these social networks is that the costs of generating and transmitting information are almost negligible, so advertising messages can reach wide audiences within a short period of time [1]. This also leads to a large volume of advertising information. However, due to the limitation of information-processing capability, if the messages arrive in numbers larger than what users can process, some messages will be lost without catching users' attention, where *information overload* occurs [2]. Under information overload, users will find it difficult to find useful messages according to their personal interests, which actually has a serious negative impact on the user experience. Therefore, to understand and then address the information overload issue arising in social networks, it is of importance to model and analyze the process of information diffusion under information overload, which is the focus of this paper.

Most research on diffusion dynamics in social networks has focused on the spread of one phenomenon at a time, for example, diffusion models for disease [3], influence [4], knowledge [5], and cooperation [6]. Recently, some researchers have begun to study competitive diffusion, which models the process that multiple competitive epidemics [7], influences [8], or phenomena [9] diffuse through a complex or social network. These problems are somewhat similar to the one considered in this paper, but they fail to characterize the information overload phenomenon in social networks, where every user can generate new messages. In our previous work [10, 11], we study the process of information diffusion under information overload in Facebook-like social networks. We know that the network structure of Twitter is very different from that of Facebook. Besides, considerable effort has been devoted to alleviate the information overload syndrome, where filter-based or cost-based approaches are usually adopted [12–14]. However, to the best of our knowledge, there is no prior work which seeks to model and analyze the process of information diffusion under information overload for Twitter-like social networks.

The remainder of this paper is organized as follows. We describe the models in Section 2 and analyze the process of information diffusion under information overload in Section 3. To verify the accuracy of theoretical analysis results, we conduct simulations and provide the simulation results in Section 4. Finally, we conclude this paper in Section 5.

2. Model Descriptions

In this section, we propose models to capture the characters of Twitter-like social networks, such as network, user behaviors, and information diffusion under information overload. Based on these models, we can analyze the process of information diffusion under information overload theoretically.

2.1. Network. We consider Twitter-like social network as a directed network, where nodes represent typical users and links represent the relationships between pairs of users. Note that a user who is followed by another user does not necessarily have to reciprocate by following him/her back. We let the direction of a link be the same as the direction of information diffusion. For example, in Figure 1, user A is followed by users B, C, D, and E, where the update messages of user A can be received by users B, C, D, and E, and user A can only receive the update messages of user C.

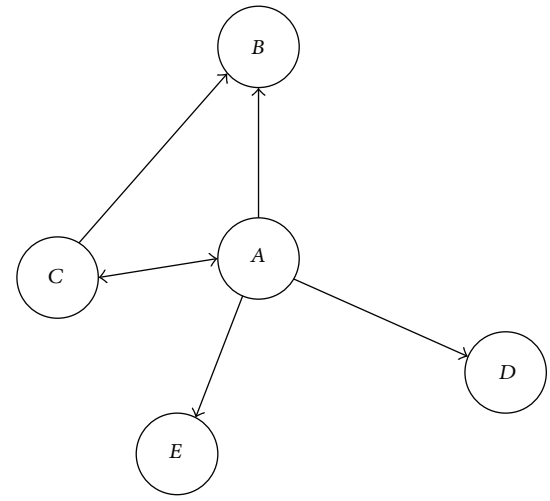


Figure 1: Network for Twitter-like social networks.

Since isolated users never get involved in the process of information diffusion, we neglect all the isolated users and classify the rest of users into different types according to their in-degrees and out-degrees; that is, a user with in-degree i and out-degree j is of type (i, j), where $i + j \geq 1$. For type (i, j) users, we define $e_{k,l}^{i,j}$ to be the probability that a randomly chosen follower is of type (k, l). Then, we have $j \geq 1$, $k \geq 1$, and

$$\sum_{k,l} e_{k,l}^{i,j} = 1. \tag{1}$$

We further define $q_{i,j}$ to be the fraction of type (i, j) users in the network, and we get

$$\sum_{i,j} q_{i,j} = 1. \tag{2}$$

Consider the ensemble of networks in which the distributions $\{e_{k,l}^{i,j}\}$ and $\{q_{i,j}\}$ take specified values. This defines a random graph model similar to the random graphs defined in [15, 16]. That is to say, the network is drawn uniformly at random from the ensemble of all possible networks with the distributions $\{e_{k,l}^{i,j}\}$ and $\{q_{i,j}\}$. For users, we denote by M the maximum number of in-degrees and by N the maximum number of out-degrees. Then, this network can be characterized by the $(M + 1) \times N \times M \times (N + 1)$ tensor $\{e_{k,l}^{i,j}\}$ and the $(M + 1) \times (N + 1)$ matrix $\{q_{i,j}\}$. Note that in a Twitter-like social network, users usually have moderate numbers of followees due to attention limitation. So, we usually have $M \ll N$.

2.2. User Behaviors. In Twitter-like social networks, different functions are adopted to diffuse information. After logging in, users can post tweets to broadcast things which happen in their daily lives. There are also other functions such as reply and retweet which allow users to interact with their friends. In this paper, we generalize these behaviors into two categories: *generating* and *forwarding*; that is, users can generate new messages or forward messages generated by other users. Note that forwarded messages can still be forwarded.

To model the user ability of message processing under information overload, we introduce the term *view scope*, which indicates the messages a user can process at a time. Note that for users in Twitter-like social networks, messages are listed in a reverse-chronological order. So for a user with view scope number S, if information overload occurs, he/she can process (i.e., browse) the latest S messages after logging in, while the former ones are lost. In this paper, we assume homogeneous view scope number, which is S, for all users.

To model user behaviors, we make the same assumptions as [10].

(i) The process of user login follows a Poisson process with rate λ.

(ii) After logging in and browsing the messages, a user may choose to log off or react to these messages (i.e., generate or forward a message), while the reacting probability is p_1.

(iii) Among the reacting actions, users may choose to forward a randomly chosen browsed message with probability p_2 or generate a new message with probability $1 - p_2$.

Actually, user online activities may be bursty, and users may generate or forward multiple messages at a time. However, we make these assumptions to simplify the analysis here and plan to extend this analysis framework to more realistic situations in our future work.

2.3. Information Diffusion under Information Overload. Under information overload, messages are arriving in numbers larger than what users can process, and some messages are lost without catching users' attention. We use Figure 2 to illustrate the evolvement of view scopes under information overload. Suppose user A is followed by other users, such as users B, C, and D. The view scopes of these users are depicted in Figure 2(a). After user A processes the messages in his/her view scope (i.e., $M_{A,1}$, $M_{A,2}$,..., and $M_{A,S}$), he/she may generate a new message or just forward a message in his/her view scope. No matter which action is chosen, this message (say $M_{A,0}$) will be placed at the top of all his/her followers' view scopes, and the messages at the bottom of his/her followers' view scopes (i.e., $M_{B,S}$, $M_{C,S}$, and $M_{D,S}$) will be discarded due to the information overload effect, which are depicted in Figure 2(b).

One may argue that the view scope of user A should be cleared after he/she has processed all the messages. However, for simplicity we assume memoryless users here. That is to say, processed messages can still be processed as long as they are in the view scope. We will model the behaviors of users with memories in our future work.

3. Performance Analysis

In this section, we analyze the process of information diffusion under information overload based on the proposed models. Specifically, we are interested in the information diffusion efficiency, which is characterized by the average number of times a message appears in view scopes after it is generated by a type (i, j) user (say $u_{i,j}$). To achieve this goal, we first calculate the average number of times a message is forwarded by a type (k, l) user after it arrives in this user's view scope (say $v_{k,l}$).

3.1. Calculation of $v_{k,l}$. Since users log in following a Poisson process with rate λ, we know that the probability that a user logs in and then generates or forwards a message within a time slot, which is of length Δ, is $\lambda\Delta p_1$. Consider a type (k, l) user (say user B). Note that he/she is memoryless and he/she may choose to forward a randomly chosen message in his/her view scope with probability p_2 after he/she decides to react to the browsed messages. So if a message (say M_0) is in his/her view scope, the average number of times that he/she will forward this message in t time slots is

$$f(t) = t\lambda\Delta p_1 p_2 \frac{1}{S}. \tag{3}$$

The followees of user B will generate or forward messages, which will be placed at the top of his/her view scope. Let $\Delta \to 0$, and then, for user B, the probability that multiple followees generate or forward messages in the same time slot can be neglected. So the probability that a new message arrives in user B's view scope in a time slot is $k\lambda\Delta p_1$. Note that message M_0 will be discarded after S new messages arrive. Then the probability that message M_0 will stay in user B's view scope for t time slots is

$$g(t) = \binom{t-1}{S-1}(1 - k\lambda\Delta p_1)^{t-S}(k\lambda\Delta p_1)^S. \tag{4}$$

Therefore, the average number of times that user B will forward message M_0 is

$$\begin{aligned} v_{k,l} &= \sum_{t\geq 0} f(t)\, g(t) \\ &= \sum_{t\geq 0} \frac{t\lambda\Delta p_1 p_2}{S} \binom{t-1}{S-1}(1 - k\lambda\Delta p_1)^{t-S}(k\lambda\Delta p_1)^S \\ &= \lambda\Delta p_1 p_2 (k\lambda\Delta p_1)^S \sum_{t\geq 0} \binom{t}{S}(1 - k\lambda\Delta p_1)^{t-S} \\ &= \lambda\Delta p_1 p_2 (k\lambda\Delta p_1)^S \sum_{t\geq 0} \binom{t+S}{S}(1 - k\lambda\Delta p_1)^{t}. \end{aligned} \tag{5}$$

From (5.56) at [17, page 199], we get

$$\sum_{t\geq 0} \binom{t+S}{S}(1 - k\lambda\Delta p_1)^t = \frac{1}{(k\lambda\Delta p_1)^{S+1}}. \tag{6}$$

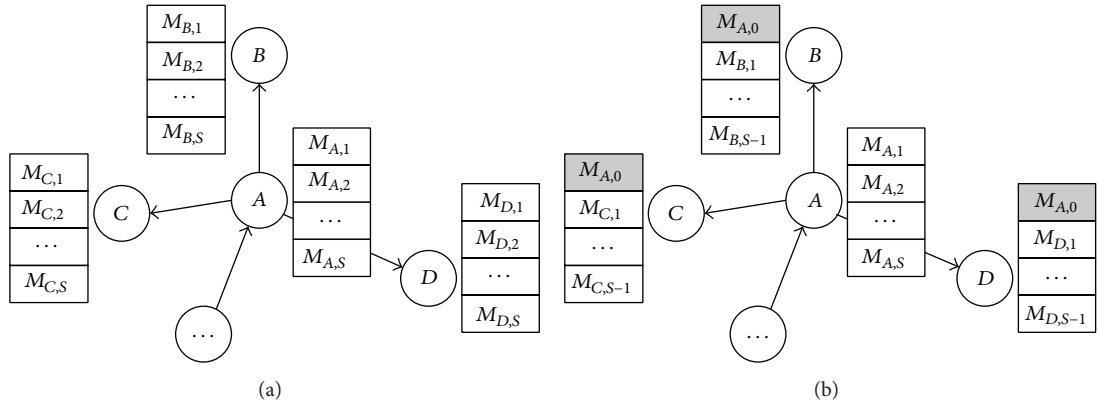


Figure 2: Evolvement of view scopes under information overload.

So, we have

$$v_{k,l} = \lambda \Delta p_1 p_2 (k \lambda \Delta p_1)^S \frac{1}{(k \lambda \Delta p_1)^{S+1}} = \frac{p_2}{k}. \tag{7}$$

Remark 1. Intuitively, the larger the view scope number S is, the longer a message stays in the view scope and the more this message is forwarded. However, from (7) we find that $v_{k,l}$ is unrelated to S. This is because larger S will lead to more messages stored in the view scope, which reduces the probability that a given message is chosen to be forwarded in a time slot.

3.2. Calculation of $u_{i,j}$. Consider a type (i, j) user (say user A) and suppose his/her followers are divided into some partition $\{r_{1,0}, r_{1,1}, \ldots, r_{M,N}\}$, where $r_{k,l}$ is the number of type (k, l) followers and

$$\sum_{k,l} r_{k,l} = j. \tag{8}$$

The probability that the partition takes a particular value $\{r_{k,l}\}$ is given by the multinomial distribution [16]

$$P\left(j, \{r_{k,l}\}\right) = j! \prod_{k,l} \frac{1}{r_{k,l}!} \left(e_{k,l}^{i,j}\right)^{r_{k,l}}. \tag{9}$$

We define the generating function $G_{i,j}(z)$ as the distribution of the number of times a message appears in view scopes after it is generated by a type (i, j) user and the generating function $H_{k,l}(z)$ as the distribution of the number of times a message appears in view scopes after it arrives in the view scope of a type (k, l) user. Then

$$G_{i,j}(z) = \sum_{\{r_{k,l}\}} \delta\left(j, \sum_{k,l} r_{k,l}\right) \times P\left(j, \{r_{k,l}\}\right) H_{1,0}(z)^{r_{1,0}} \cdots H_{M,N}(z)^{r_{M,N}}, \tag{10}$$

$$H_{k,l}(z) = z G_{k,l}(z)^{p_2/k}, \tag{11}$$

where δ is the Kronecker delta function, and

$$\delta\left(j, \sum_{k,l} r_{k,l}\right) = \begin{cases} 1, & \text{if } j = \sum_{k,l} r_{k,l}, \\ 0, & \text{if } j \neq \sum_{k,l} r_{k,l}. \end{cases} \tag{12}$$

By submitting (11) into (10), we get

$$G_{i,j}(z) = z^j \sum_{\{r_{k,l}\}} \delta\left(j, \sum_{k,l} r_{k,l}\right) \times P\left(j, \{r_{k,l}\}\right) G_{1,0}(z)^{p_2 r_{1,0}} \cdots G_{M,N}(z)^{p_2 r_{M,N}/M}. \tag{13}$$

Then, by submitting (9) into (13) and performing the sum over $\{r_{k,l}\}$, we have

$$G_{i,j}(z) = z^j \left(\sum_{k,l} e_{k,l}^{i,j} G_{k,l}(z)^{p_2/k} \right)^j. \tag{14}$$

By solving this equation, we can derive the distribution of the number of times a message appears in view scopes after it is generated by a type (i, j) user. However, here we just calculate the average number of times, which is

$$u_{i,j} = G'_{i,j}(1) = j + j p_2 \sum_{k,l} \frac{e_{k,l}^{i,j}}{k} G'_{k,l}(1). \tag{15}$$

We know that users with out-degree 0 can generate or forward messages, but no one can receive them. So, $G_{i,0}(z) = 1$ and $G'_{i,0}(1) = 0$. We further know that users with in-degree 0 never forward messages. That is to say, $G'_{0,j}(1)$ never contributes to the right of (15). So we can first calculate $G'_{i,j}(1)$ where $i, j \geq 1$ and then get $G'_{0,j}(1)$ from (15).

We know that $\{e_{k,l}^{i,j}\}$ becomes an $M \times N \times M \times N$ tensor for $i, j, k, l \geq 1$, which is still hard to handle. We rearrange the elements of this tensor so that they form a matrix, which is called *matricizing* [18]. Specifically, we let

$$x = (i-1) N + j, \qquad y = (k-1) N + l. \tag{16}$$

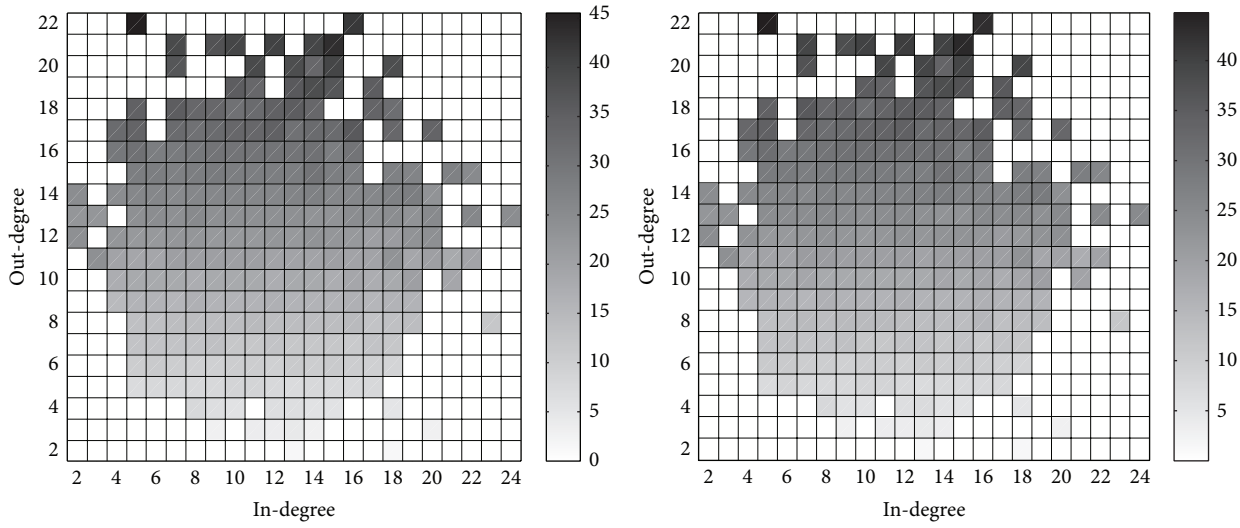


Figure 3: Results for $u_{i,j}$ from (a) simulations and (b) theoretical analysis.

Then $1 \le x, y \le MN$ and

$$G'_x(1) = j + jp_2 \sum_y \frac{e_{x,y}}{k} G'_y(1). \tag{17}$$

We can write (17) in matrix form and get

$$\mathbf{G'}(1) = \mathbf{X1} + p_2\, \mathbf{XEY}^{-1}\mathbf{G'}(1), \tag{18}$$

where

$$\mathbf{1} = (1, 1, \ldots, 1)^T,$$

$$\mathbf{X} = \begin{pmatrix} \mathbf{X_0} & 0 & \cdots & 0 \\ 0 & \mathbf{X_0} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{X_0} \end{pmatrix}_{M\times M},$$

$$\mathbf{X_0} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & N \end{pmatrix},$$

$$\mathbf{Y} = \begin{pmatrix} \mathbf{Y_1} & 0 & \cdots & 0 \\ 0 & \mathbf{Y_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{Y_M} \end{pmatrix}, \tag{19}$$

$$\mathbf{Y_m} = \begin{pmatrix} m & 0 & \cdots & 0 \\ 0 & m & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & m \end{pmatrix}_{N\times N}.$$

So we get

$$\mathbf{G'}(1) = \left(\mathbf{I} - p_2 \mathbf{XEY}^{-1}\right)^{-1} \mathbf{X1}. \tag{20}$$

Remark 2. From (20), we observe that $u_{i,j}$ is determined by p_2 and $\{e_{k,l}^{i,j}\}$ and is unrelated to other factors such as S, p_1, and $\{q_{i,j}\}$.

4. Simulations

To verify the accuracy of theoretical analysis results, we conduct simulations and provide the simulation results in this section. We first take into account a directed ER network and then a growing network model, which generates directed and degree-correlated networks.

The simulations are conducted in a discrete fashion. Specifically, time is slotted, and in each time slot a random user is selected to generate or forward a message. Denoting by K the user number, each simulation is run KT time slots, where $T = 10000$. That is to say, each user will be selected T times on average to generate or forward messages. We further set $S = 10$ and $p_2 = 0.5$.

4.1. Directed ER Network. In the directed ER network, we let the user number $K = 2001$ and the average user in-degree (or out-degree) $\alpha = 10$. That is to say, each link is included in the network with probability $p = \alpha/(K-1) = 0.005$.

The results for $u_{i,j}$ from simulations and theoretical analysis (i.e., (20)) are depicted in Figure 3. To quantify the gap between these results, we plot the differences in Figure 4, from which we know that the theoretical analysis results coincide very well with the simulation results.

4.2. Growing Network Model. Degree correlations among nodes in a network essentially characterize the network structure, while many real-world networks show degree correlations [19–21]. In particular, social networks show assortative mixing, that is, a preference of high-degree nodes to be connected to other high-degree nodes [15, 16].

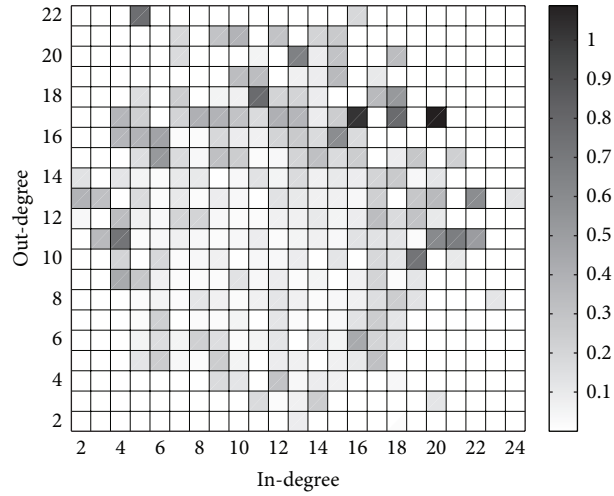


Figure 4: Differences between the results for $u_{i,j}$ from simulations and theoretical analysis.

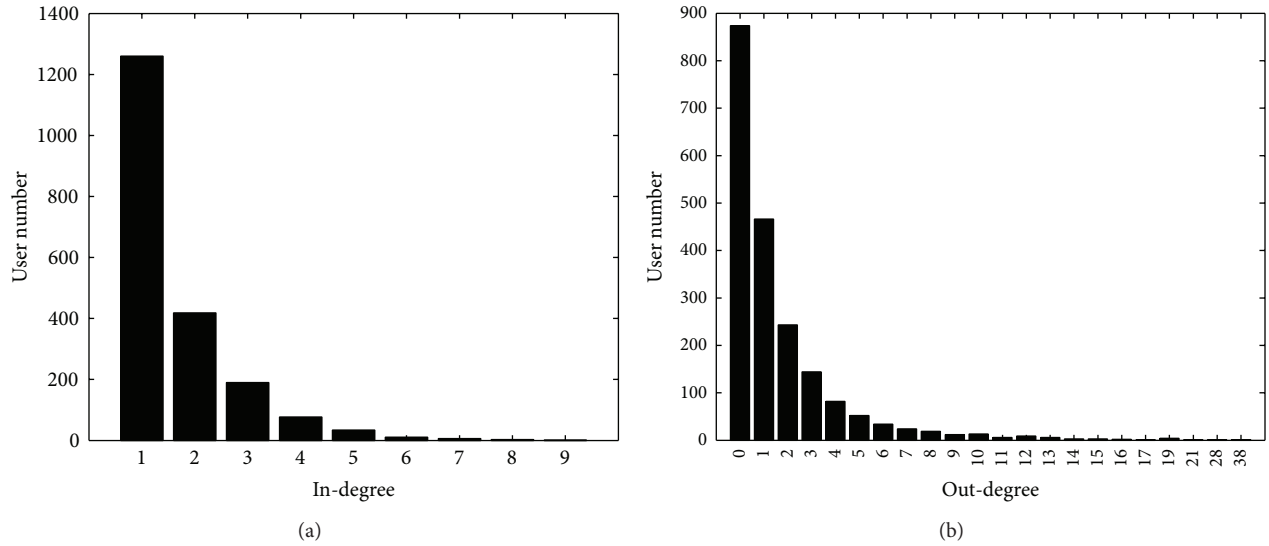


Figure 5: Distributions of (a) in-degrees and (b) out-degrees.

To generate directed and degree-correlated networks, we adopt the growing network model proposed in [22], where in each step the probability of adding a new node and creating a link from one of the earlier nodes (say A) is

$$q\frac{d_A^{\text{out}}+\beta}{\sum_{C\in V}\left(d_C^{\text{out}}+\beta\right)}, \tag{21}$$

and the probability of adding a new link and connecting two old nonlinked nodes (say from A to B) is

$$(1-q)\,\frac{d_A^{\text{out}}+\beta}{\sum_{C\in V}\left(d_C^{\text{out}}+\beta\right)}\frac{d_B^{\text{in}}+\gamma}{\sum_{C\in V}\left(d_C^{\text{in}}+\gamma\right)}, \tag{22}$$

where V is the node set, d_*^{out} (d_*^{in}) is the out-degree (in-degree) of node $*$, and parameters β, γ must obey the constraints $\beta > 0$ and $\gamma > -1$ to ensure that each node will be chosen with positive probability.

Here, we set $K = 2000$, $q = 0.6$, and $\beta = \gamma = 5$ to generate a degree-correlated network. The distributions of in-degrees and out-degrees are depicted in Figure 5, from which we know that in-degrees and out-degrees follow power law distributions. The degree correlations are depicted in Figure 6, from which we know that degree correlations at users are evident, while high in-degree users usually have high out-degrees. However, the degree correlations at links are not so obvious.

Simulation results for $u_{i,j}$ are depicted in Figure 7(a), while the theoretical analysis results are depicted in Figure 7(b). We also plot the differences in Figure 8, from which we know that the theoretical analysis results are quite consistent with the simulation results, especially for users with low in-degrees and out-degrees. However, even for the only user who is of type $(5, 39)$, the value of difference is about 6, which is very small compared to the value of $u_{5,39}$.

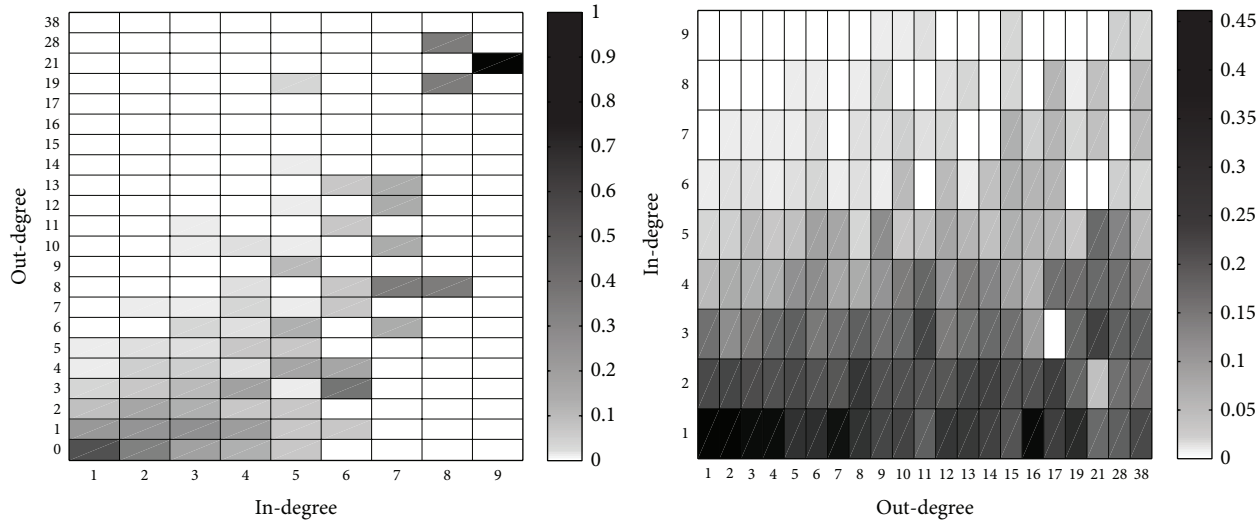


Figure 6: Degree correlations. (a) Degree correlations at users. Columns indicate the distributions of out-degrees for users with given in-degrees. (b) Degree correlations at links. Columns indicate the distributions of followers' in-degrees for users with given out-degrees.

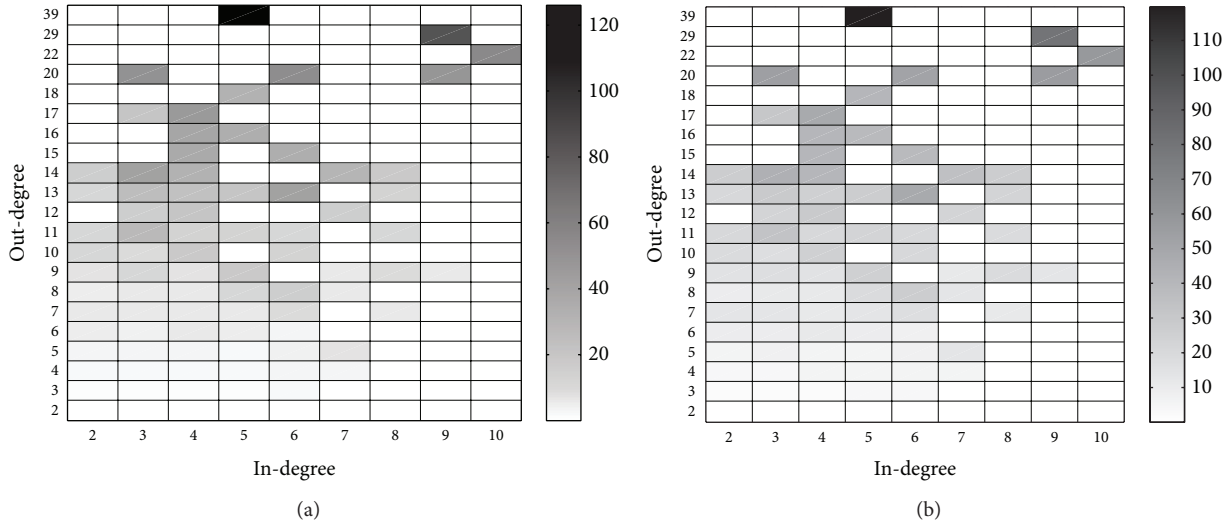


Figure 7: Results for $u_{i,j}$ from (a) simulations and (b) theoretical analysis.

5. Conclusion

Having noticed the increasing popularity of Twitter and negative influence of information overload, we take Twitter-like social networks into account and propose models to capture the characters such as network, user behaviors, and information diffusion under information overload. Based on these models, we analyze the process of information diffusion under information overload theoretically, and the accuracy of theoretical analysis results is verified by simulations. These results are of importance to understand the diffusion dynamics in social networks and of use for advertisers in viral marketing. However, to simplify the analysis, we make some assumptions such as Poisson arrival and memoryless users, which seem to be unrealistic. We seek to extend these models to characterize more realistic situations and validate the theoretical analysis results by empirical evidence in our future work. Besides, the impact of degree correlations on spreading dynamics appears to be nontrivial [23], and it is demonstrated that degree correlations strongly influence information diffusion [24]. Another future work of this paper is to analyze the impact of degree correlations on the information diffusion under information overload.

Conflict of Interests

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

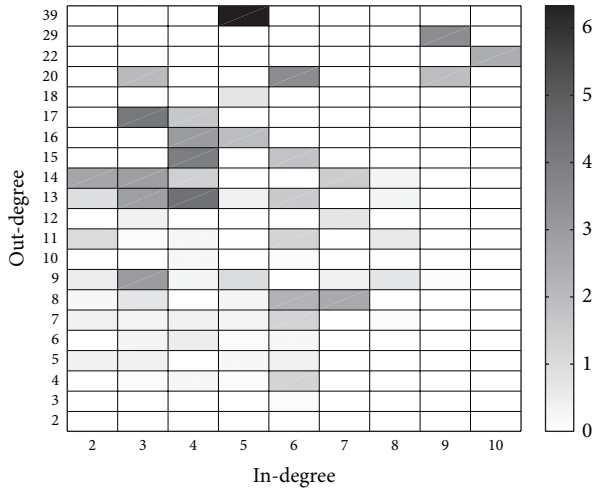


Figure 8: Differences between the results for $u_{i,j}$ from simulations and theoretical analysis.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants nos. 61105124 and 71331008, the Research Fund for the Doctoral Program of Higher Education of China under Grant no. 20114307120023, and the China Scholarship Council under Grant no. 2011611534.

References

[1] J. Leskovec, L. A. Adamic, and B. A. Huberman, "The dynamics of viral marketing," *ACM Transactions on the Web*, vol. 1, no. 1, 2007.

[2] K. Koroleva, H. Krasnova, and O. Gunther, "'stop spamming me!'-exploring information overload on facebook," in *Proceedings of the 16th Americas Conference on Information Systems (AMCIS '10)*, Lima, Peru, August 2010.

[3] R. M. Anderson and R. M. May, "Population biology of infectious diseases: part I," *Nature*, vol. 280, no. 5721, pp. 361–367, 1979.

[4] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '03)*, pp. 137–146, New York, NY, USA, August 2003.

[5] R. Cowan and N. Jonard, "Network structure and the diffusion of knowledge," *Journal of Economic Dynamics and Control*, vol. 28, no. 8, pp. 1557–1575, 2004.

[6] F. C. Santos, J. M. Pacheco, and T. Lenaerts, "Evolutionary dynamics of social dilemmas in structured heterogeneous populations," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 9, pp. 3490–3494, 2006.

[7] B. Karrer and M. E. J. Newman, "Competing epidemics on complex networks," *Physical Review E-Statistical, Nonlinear, and Soft Matter Physics*, vol. 84, no. 3, Article ID 036106, 2011.

[8] A. Borodin, Y. Filmus, and J. Oren, "Threshold models for competitive influence in social networks," in *Proceedings of the 6th International Conference on Internet and Network Economics (WINE '10)*, pp. 539–550, Stanford, Calif, USA, 2010.

[9] M. Broecheler, P. Shakarian, and V. S. Subrahmanian, "A scalable framework for modeling competitive diffusion in social networks," in *Proceedings of the 2nd IEEE International Conference on Social Computing (SocialCom '10), and the 2nd IEEE International Conference on Privacy, Security, Risk and Trust (PASSAT '10)*, pp. 295–302, August 2010.

[10] P. Li, K. Xing, D. Wang, X. Zhang, and H. Wang, "Information diffusion in facebook- like social networks under information overload," *International Journal of Modern Physics C*, vol. 24, no. 7, Article ID 1350047, 2013.

[11] P. Li, Y. Sun, Y. Chen, and Z. Tian, "Estimating user influence in online social networks subject to information overload," *International Journal of Modern Physics B*, vol. 28, no. 3, Article ID 1450004, 2014.

[12] E. Solan and E. Reshef, "The effect of filters on spam mail," Discussion Papers 1402, Northwestern University, Center for Mathematical Studies in Economics and Management Science, Evanston, Ill, USA, 2005.

[13] R. E. Kraut, S. Sunder, J. Morris, R. Telang, D. Filer, and M. Cronin, "Markets for attention: will postage for email help?" in *Proceedings of the 8th 2002 ACM Conference on Computer Supported Cooperative Work (CSCW '02)*, pp. 206–215, New Orleans, La, USA, November 2002.

[14] J. Cheng, A. Sun, and D. Zeng, "Information overload and viral marketing: countermeasures and strategies," in *Lecture Notes in Computer Science*, vol. 6007, pp. 108–117, 2010.

[15] M. E. J. Newman, "Assortative mixing in networks," *Physical Review Letters*, vol. 89, no. 20, Article ID 208701, 2002.

[16] M. E. J. Newman, "Mixing patterns in networks," *Physical Review E*, vol. 67, Article ID 026126, 2003.

[17] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics*, Addison Wesley, Reading, Mass, USA, 1994.

[18] B. W. Bader and T. G. Kolda, "Algorithm 862: MATLAB tensor classes for fast algorithm prototyping," *ACM Transactions on Mathematical Software*, vol. 32, no. 4, pp. 635–653, 2006.

[19] R. Pastor-Satorras, A. Vázquez, and A. Vespignani, "Dynamical and correlation properties of the internet," *Physical Review Letters*, vol. 87, no. 25, Article ID 258701, pp. 1–4, 2001.

[20] A. Vázquez, R. Pastor-Satorras, and A. Vespignani, "Large-scale topological and dynamical properties of the Internet," *Physical Review E-Statistical, Nonlinear, and Soft Matter Physics*, vol. 65, no. 6, Article ID 066130, pp. 1–12, 2002.

[21] M. E. J. Newman and J. Park, "Why social networks are different from other types of networks," *Physical Review E-Statistical, Nonlinear, and Soft Matter Physics*, vol. 68, no. 3, Article ID 036122, pp. 1–18, 2003.

[22] P. L. Krapivsky, G. J. Rodgers, and S. Redner, "Degree distributions of growing networks," *Physical Review Letters*, vol. 86, no. 23, pp. 5401–5404, 2001.

[23] J. L. Payne, P. S. Dodds, and M. J. Eppstein, "Information cascades on degree-correlated random networks," *Physical Review E-Statistical, Nonlinear, and Soft Matter Physics*, vol. 80, no. 2, Article ID 026125, 2009.

[24] M. Karsai, M. Kivelä, R. K. Pan et al., "Small but slow world: how network topology and burstiness slow down spreading," *Physical Review E-Statistical, Nonlinear, and Soft Matter Physics*, vol. 83, no. 2, Article ID 025102, 2011.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 572969, 9 pages
http://dx.doi.org/10.1155/2014/572969

Research Article

An Improved Proportionate Normalized Least-Mean-Square Algorithm for Broadband Multipath Channel Estimation

Yingsong Li and Masanori Hamamura

Graduate School of Engineering, Kochi University of Technology, Kami-shi 782-8502, Japan

Correspondence should be addressed to Yingsong Li; liyingsong@ieee.org

Received 27 December 2013; Accepted 19 February 2014; Published 20 March 2014

Academic Editors: H. R. Karimi, X. Yang, Z. Yu, and W. Zhang

To make use of the sparsity property of broadband multipath wireless communication channels, we mathematically propose an l_p-norm-constrained proportionate normalized least-mean-square (LP-PNLMS) sparse channel estimation algorithm. A general l_p-norm is weighted by the gain matrix and is incorporated into the cost function of the proportionate normalized least-mean-square (PNLMS) algorithm. This integration is equivalent to adding a zero attractor to the iterations, by which the convergence speed and steady-state performance of the inactive taps are significantly improved. Our simulation results demonstrate that the proposed algorithm can effectively improve the estimation performance of the PNLMS-based algorithm for sparse channel estimation applications.

1. Introduction

Broadband signal transmission is becoming a commonly used high-data-rate technique for next-generation wireless communication systems, such as 3 GPP long-term evolution (LTE) and worldwide interoperability for microwave access (WiMAX) [1]. The transmission performance of coherent detection for such broadband communication systems strongly depends on the quality of channel estimation [2–5]. Fortunately, broadband multipath channels can be accurately estimated using adaptive filter techniques [6–10] such as the normalized least-mean-square (NLMS) algorithm, which has low complexity and can be easily implemented at the receiver. On the other hand, channel measurements have shown that broadband wireless multipath channels can often be described by only a small number of propagation paths with long delays [4, 11, 12]. Thus, a broadband multipath channel can be regarded as a sparse channel with only a few active dominant taps, while the other inactive taps are zero or close to zero. This inherent sparsity of the channel impulse response (CIR) can be exploited to improve the quality of channel estimation. However, such classical NLMS algorithms with a uniform step size across all filter coefficients have slow convergence when estimating sparse impulse response signals such as those in broadband sparse wireless multipath channels [11]. Consequently, corresponding algorithms have recently received significant attention in the context of compressed sensing (CS) [5, 12–14] and were already considered for channel estimation prior to the CS era [5, 12]. However, these CS channel estimation algorithms are sensitive to the noise in wireless multipath channels.

Inspired by the CS theory [12–14], several zero-attracting (ZA) algorithms have been proposed and investigated by combining the CS theory and the standard least-mean-square (LMS) algorithm for echo cancellation and system identification, which are known as the zero-attracting LMS (ZA-LMS) and reweighted ZA-LMS (RZA-LMS) algorithms, respectively [15]. Recently, this technique has been expanded to the NLMS algorithm and other adaptive filter algorithms to improve their convergence speed in a sparse environment [9, 16–18]. However, these approaches are mainly designed for nonproportionate adaptive algorithms. On the other hand, to utilize the advantages of the NLMS algorithm, such as stable performance and low complexity, the proportionate

normalized least-mean-square (PNLMS) algorithm has been proposed and studied to exploit the sparsity in nature [19] and has been applied to echo cancellation in telephone networks. Although the PNLMS algorithm can utilize the sparsity characteristics of a sparse signal and obtain faster convergence at the initial stage by assigning independent magnitudes to the active taps, the convergence speed is reduced by even more than that of the NLMS algorithm for the inactive taps after the active taps converge. Consequently, several algorithms have been proposed to improve the convergence speed of the PNLMS algorithm [20–27], which include the use of the l_1-norm technique and a variable step size. Although these algorithms have significantly improved the convergence speed of the PNLMS algorithm, they still converge slowly after the active taps converge. In addition, some of them are inferior to the NLMS and PNLMS algorithms in terms of the steady-state error when the sparsity decreases. From these previously proposed sparse signal estimation algorithms, we know that the ZA algorithms mainly exert a penalty on the inactive channel taps through the integration of the l_1-norm constraint into the cost function of the standard LMS algorithms to achieve better estimation performance, while the PNLMS algorithm updates each filter coefficient with an independent step size, which improves the convergence of the active taps.

Motivated by the CS theory [13, 14] and ZA technique [15–18], we propose an l_p-norm-constrained PNLMS (LP-PNLMS) algorithm that incorporates the l_p-norm into the cost function of the PNLMS algorithm, resulting in an improved proportionate adaptive algorithm. The difference between the proposed LP-PNLMS algorithm and the ZA algorithms is that the gain-matrix-weighted l_p-norm is used in our proposed LP-PNLMS algorithm instead of the general l_1-norm to expand the application of ZA algorithms [15]. Also, this integration is equivalent to adding a zero attractor in the iterations of the PNLMS algorithm to obtain the benefits of both the PNLMS and ZA algorithms. Thus, our proposed LP-PNLMS algorithm can achieve fast convergence at the initial stage for the active taps. After the convergence of these active taps, the ZA technique in the LP-PNLMS algorithm acts as another force to attract the inactive taps to zero to arrest the slow convergence of the PNLMS algorithm. Furthermore, our proposed LP-PNLMS algorithm achieves a lower mean square error than the PNLMS algorithm and its related improved algorithms, such as the improved PNLMS (IPNLMS) [20] and μ-law PNLMS (MPNLMS) [21] algorithms. In this study, our proposed LP-PNLMS algorithm is verified over a sparse multipath channel by comparison with the NLMS, PNLMS, IPNLMS, and MPNLMS algorithms. The simulation results demonstrate that the LP-PNLMS algorithm achieves better channel estimation performance in terms of both convergence speed and steady-state behavior for sparse channel estimation.

The remainder of this paper is organized as follows. Section 2 briefly reviews the standard NLMS, PNLMS, and improved PNLMS algorithms, including the IPNLMS and MPNLMS algorithms. In Section 3, we describe in detail the proposed LP-PNLMS algorithm, which employs the Lagrange multiplier method. In Section 4, the estimation performance of the proposed LP-PNLMS algorithm is verified over sparse channels and compared with other commonly used algorithms. Finally, this paper is concluded in Section 5.

2. Related Channel Estimation Algorithms

2.1. Normalized Least-Mean-Square Algorithm. In this section, we first consider the sparse multipath communication system shown in Figure 1 to discuss the channel estimation algorithms. The input signal $\mathbf{x}(n) = [x(n), x(n-1), \ldots, x(n-N+1)]^T$ containing the N most recent samples is transmitted over a finite impulse response (FIR) channel with channel impulse response (CIR) $\mathbf{h} = [h_0, h_1, \ldots, h_{N-1}]^T$, where $(\cdot)^T$ denotes the transposition operation. Then the output signal of the channel is written as follows:

$$y(n) = \mathbf{h}^T \mathbf{x}(n), \tag{1}$$

where $\mathbf{h}$ is a sparse channel vector with K dominant active taps whose magnitudes are larger than zero and $(N - K)$ inactive taps whose magnitudes are zero or close to zero with $K \ll N$. To estimate the unknown sparse channel $\mathbf{h}$, an NLMS algorithm uses the input signal $\mathbf{x}(n)$, the output signal $y(n)$, and the instantaneous estimation error $e(n)$, which is given by

$$e(n) = d(n) - \widehat{\mathbf{h}}^T(n)\,\mathbf{x}(n), \tag{2}$$

where $\widehat{\mathbf{h}}(n)$ is the NLMS adaptive channel estimator at instant n, $d(n) = y(n) + v(n)$, and $v(n)$ is an additive noise at the receiver. The update function of the NLMS channel estimation algorithm is expressed as

$$\widehat{\mathbf{h}}(n+1) = \widehat{\mathbf{h}}(n) + \mu_{\text{NLMS}} \frac{e(n)\,\mathbf{x}(n)}{\mathbf{x}^T(n)\,\mathbf{x}(n) + \delta_{\text{NLMS}}}, \tag{3}$$

where μ_{NLMS} is the step size with $0 < \mu_{\text{NLMS}} < 2$ and δ_{NLMS} is a small positive constant used to avoid division by zero.

2.2. Proportionate Normalized Least-Mean-Square Algorithm. The PNLMS algorithm, which is an NLMS algorithm improved by the use of a proportionate technique, has been proposed for sparse system identification and echo cancellation. In this algorithm, each tap is assigned an individual step size, which is obtained from the previous estimation of the filter coefficient. According to the gain allocation rule in this algorithm, the greater the magnitude of the tap, the larger the step size assigned to it, and hence the active taps converge quickly. The update function of the PNLMS algorithm [19] is described by the following equation with reference to Figure 1:

$$\widehat{\mathbf{h}}(n+1) = \widehat{\mathbf{h}}(n) + \mu_{\text{PNLMS}} \frac{e(n)\,\mathbf{G}(n)\,\mathbf{x}(n)}{\mathbf{x}^T(n)\,\mathbf{G}(n)\,\mathbf{x}(n) + \delta_{\text{PNLMS}}}. \tag{4}$$

Here, $\mathbf{G}(n)$, which denotes as the gain matrix, is a diagonal matrix that modifies the step size of each tap, μ_{PNLMS} is the

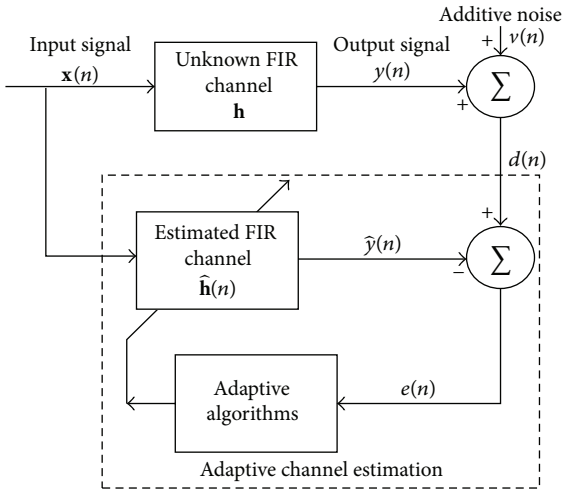


Figure 1: Typical sparse multipath communication system.

global step size of the PNLMS algorithm, and $\delta_{\text{PNLMS}} = \delta_x^2/N$ is a regularization parameter to prevent division by zero at the initialization stage, where δ_x^2 is the power of the input signal $\mathbf{x}(n)$. In the PNLMS algorithm, the gain matrix $\mathbf{G}(n)$ is given by

$$\mathbf{G}(n) = \operatorname{diag}\left(g_0(n), g_1(n), \ldots, g_{N-1}(n)\right), \tag{5}$$

where the individual gain $g_i(n)$ is defined as

$$g_i(n) = \frac{\gamma_i(n)}{\sum_{i=0}^{N-1}\gamma_i(n)}, \quad 0 \le i \le N-1 \tag{6}$$

with

$$\gamma_i(n) = \max\left[\rho_g \max\left[\delta_p, \left|\hat{h}_0(n)\right|, \left|\hat{h}_1(n)\right|, \ldots, \left|\hat{h}_{N-1}(n)\right|\right], \left|\hat{h}_i(n)\right|\right], \tag{7}$$

where the parameters δ_p and ρ_g are positive constants with typical values of $\delta_p = 0.01$ and $\rho_g = 5/N$. δ_p is used to regularize the updating at the initial stage when all the taps are initialized to zero, and ρ_g is used to prevent $\hat{h}_i(n)$ from stalling when it is much smaller than the largest coefficient.

2.3. Improved Proportionate Normalized Least-Mean-Square Algorithms

2.3.1. IPNLMS Algorithm. The IPNLMS algorithm is a type of PNLMS algorithm used to improve the convergence speed of the PNLMS algorithm. It is a combination of the PNLMS and NLMS algorithms with the relative significance of each coefficient controlled by a factor α. The IPNLMS algorithm [20] adopts the l_1-norm to enable the smooth selection of (7), and the update equation of the IPNLMS algorithm is expressed as

$$\hat{\mathbf{h}}(n+1) = \hat{\mathbf{h}}(n) - \mu_{\text{IPNLMS}} \frac{e(n)\mathbf{K}(n)\mathbf{x}(n)}{\mathbf{x}^T(n)\mathbf{K}(n)\mathbf{x}(n) + \delta_{\text{IPNLMS}}}, \tag{8}$$

where $\mathbf{K}(n) = \operatorname{diag}(k_0(n), k_1(n), \ldots, k_{N-1}(n))$ is a diagonal matrix used to adjust the step size of the IPNLMS algorithm, where

$$k_j(n) = \frac{1-\alpha}{2N} + (1+\alpha)\frac{\left|\hat{h}_j(n)\right|}{2\left\|\hat{\mathbf{h}}(n)\right\|_1 + \varepsilon}, \quad 0 \le j \le N-1 \tag{9}$$

for a small positive constant ε and $-1 \le \alpha \le 1$. At the initial stage, the step size is multiplied by $(1-\alpha)/2N$, since all the filter coefficients are initialized to zero. Thus, in the IPNLMS algorithm, a regularization parameter δ_{IPNLMS} is introduced, which is given by

$$\delta_{\text{IPNLMS}} = \frac{1-\alpha}{2N}\delta_{\text{NLMS}}. \tag{10}$$

We can see that the IPNLMS is identical to the NLMS algorithm for $\alpha = -1$, while the IPNLMS behaves identically to the PNLMS algorithm when $\alpha = 1$. In practical engineering applications, a suitable value for α is 0 or -0.5.

2.3.2. MPNLMS Algorithm. The μ-law PNLMS algorithm (MPNLMS) is another enhancement of the PNLMS algorithm that utilizes the logarithm of the magnitudes of the filter coefficients instead of using the magnitudes directly in the PNLMS algorithm [21]. The update equation is the same as that in the PNLMS algorithm given by (4). In the MPNLMS algorithm,

$$\gamma_i(n) = \max\left[\rho_g \max\left[\delta_p F\left(\left|\hat{h}_0(n)\right|\right), F\left(\left|\hat{h}_1(n)\right|\right), \ldots, F\left(\left|\hat{h}_{N-1}(n)\right|\right)\right], F\left(\left|\hat{h}_i(n)\right|\right)\right], \tag{11}$$

where

$$F\left(\left|\hat{h}_i(n)\right|\right) = \log\left(1 + \vartheta\left|\hat{h}_i(n)\right|\right), \tag{12}$$

where ϑ is a large positive constant related to the estimation accuracy requirement, typically $\vartheta = 1000$.

3. Proposed LP-PNLMS Algorithm

In this section, we propose an LP-PNLMS algorithm by incorporating the l_p-norm into the cost function of the PNLMS algorithm to create a zero attractor, making it a type of ZA algorithm. The difference between the LP-PNLMS algorithm and general ZA algorithms is that the gain-matrix-weighted l_p-norm is taken into account in designing the zero attractor. On the other hand, the proposed LP-PNLMS algorithm is based on the commonly used PNLMS algorithm, which is also a sparse channel estimation algorithm and can improve the convergence for the active taps. Regarding

channel estimation, the purpose of the LP-PNLMS algorithm is to minimize

$$\begin{aligned}&\left(\widehat{\mathbf{h}}(n+1)-\widehat{\mathbf{h}}(n)\right)^{T}\mathbf{G}^{-1}(n)\\&\quad\times\left(\widehat{\mathbf{h}}(n+1)-\widehat{\mathbf{h}}(n)\right)+\gamma_{\mathrm{LP}}\left\|\mathbf{G}^{-1}(n)\widehat{\mathbf{h}}(n+1)\right\|_{p}\\&\text{subject to}\\&d(n)-\widehat{\mathbf{h}}^{T}(n+1)\,\mathbf{x}(n)=0,\end{aligned}\tag{13}$$

where $\mathbf{G}^{-1}(n)$ is the inverse of the gain matrix $\mathbf{G}(n)$ in the PNLMS algorithm, $\gamma_{\mathrm{LP}} > 0$ is a very small constant used to balance the estimation error and the sparse l_p-norm penalty of $\widehat{\mathbf{h}}(n+1)$, $\|\cdot\|_p$ is the p-norm defined as $\|\widehat{\mathbf{h}}\|_p = (\sum_i \widehat{h}_i^p)^{1/p}$, and $0 \le p \le 1$. Note that in (13), we introduce an l_p-norm penalty to $\widehat{\mathbf{h}}(n+1)$ after scaling the gain matrix by $\mathbf{G}^{-1}(n)$, which is different from the previously proposed ZA LMS algorithms.

To minimize (13), the Lagrange multiplier method is adopted, and the cost function $J_{\mathrm{LP}}(n+1)$ of the proposed LP-PNLMS algorithm is expressed as

$$\begin{aligned}J_{\mathrm{LP}}(n+1)=\ &\left(\widehat{\mathbf{h}}(n+1)-\widehat{\mathbf{h}}(n)\right)^{T}\mathbf{G}^{-1}(n)\\&\times\left(\widehat{\mathbf{h}}(n+1)-\widehat{\mathbf{h}}(n)\right)+\gamma_{\mathrm{LP}}\left\|\mathbf{G}^{-1}\widehat{\mathbf{h}}(n+1)\right\|_{p}\\&+\lambda\left(d(n)-\widehat{\mathbf{h}}^{T}(n+1)\,\mathbf{x}(n)\right),\end{aligned}\tag{14}$$

where λ is the Lagrange multiplier.

By calculating the gradient of the cost function $J_{\mathrm{LP}}(n+1)$ of the LP-PNLMS algorithm and assuming $\widehat{\mathbf{h}}(n+1) = \widehat{\mathbf{h}}(n)$ in the steady stage, we have

$$\frac{\partial J_{\mathrm{LP}}(n+1)}{\partial\widehat{\mathbf{h}}(n+1)}=\mathbf{0},\qquad\frac{\partial J_{\mathrm{LP}}(n+1)}{\partial\lambda}=0,\tag{15}$$

$$\widehat{\mathbf{h}}(n+1)=\widehat{\mathbf{h}}(n)+\lambda\mathbf{G}(n)\,\mathbf{x}(n)-\gamma_{\mathrm{LP}}\frac{\left\|\widehat{\mathbf{h}}(n)\right\|_{p}^{1-p}\mathrm{sgn}\left(\widehat{\mathbf{h}}(n)\right)}{\left|\widehat{\mathbf{h}}(n)\right|^{1-p}}.\tag{16}$$

In practice, we need to introduce a small positive constant into the final term in (16) to cope with the situation that an entry of $\widehat{\mathbf{h}}(n)$ approaches zero, which is the case for a sparse CIR at initialization. Then the update equation (16) of the LP-PNLMS algorithm is modified to

$$\widehat{\mathbf{h}}(n+1)=\widehat{\mathbf{h}}(n)+\lambda\mathbf{G}(n)\,\mathbf{x}(n)-\gamma_{\mathrm{LP}}\frac{\left\|\widehat{\mathbf{h}}(n)\right\|_{p}^{1-p}\mathrm{sgn}\left(\widehat{\mathbf{h}}(n)\right)}{\left|\widehat{\mathbf{h}}(n)\right|^{1-p}+\varepsilon_{p}},\tag{17}$$

where ε_p is a small value to prevent division by zero. By multiplying both sides of (17) by $\mathbf{x}^T(n)$, we obtain

$$\begin{aligned}\mathbf{x}^{T}(n)\,\widehat{\mathbf{h}}(n+1)=\ &\mathbf{x}^{T}(n)\,\widehat{\mathbf{h}}(n)+\lambda\mathbf{x}^{T}(n)\,\mathbf{G}(n)\,\mathbf{x}(n)\\&-\gamma_{\mathrm{LP}}\frac{\mathbf{x}^{T}(n)\left\|\widehat{\mathbf{h}}(n)\right\|_{p}^{1-p}\mathrm{sgn}\left(\widehat{\mathbf{h}}(n)\right)}{\left|\widehat{\mathbf{h}}(n)\right|^{1-p}+\varepsilon_{p}}.\end{aligned}\tag{18}$$

From (2), (15), and (17), we obtain

$$\begin{aligned}e(n)=\ &-\gamma_{\mathrm{LP}}\frac{\mathbf{x}^{T}(n)\left\|\widehat{\mathbf{h}}(n)\right\|_{p}^{1-p}\mathrm{sgn}\left(\widehat{\mathbf{h}}(n)\right)}{\left|\widehat{\mathbf{h}}(n)\right|^{1-p}+\varepsilon_{p}}\\&+\lambda\mathbf{x}^{T}(n)\,\mathbf{G}(n)\,\mathbf{x}(n).\end{aligned}\tag{19}$$

Then, the Lagrange multiplier λ is given as follows by solving (19):

$$\begin{aligned}\lambda=\ &\left(e(n)+\gamma_{\mathrm{LP}}\frac{\mathbf{x}^{T}(n)\left\|\widehat{\mathbf{h}}(n)\right\|_{p}^{1-p}\mathrm{sgn}\left(\widehat{\mathbf{h}}(n)\right)}{\left|\widehat{\mathbf{h}}(n)\right|^{1-p}+\varepsilon_{p}}\right)\\&\times\left(\mathbf{x}^{T}(n)\,\mathbf{G}(n)\,\mathbf{x}(n)\right)^{-1}.\end{aligned}\tag{20}$$

Substituting (20) into (17), we have

$$\begin{aligned}\widehat{\mathbf{h}}(n+1)=\ &\widehat{\mathbf{h}}(n)-\gamma_{\mathrm{LP}}\frac{\left\|\widehat{\mathbf{h}}(n)\right\|_{p}^{1-p}\mathrm{sgn}\left(\widehat{\mathbf{h}}(n)\right)}{\left|\widehat{\mathbf{h}}(n)\right|^{1-p}+\varepsilon_{p}}\\&+\left(\left(e(n)+\gamma_{\mathrm{LP}}\frac{\mathbf{x}^{T}(n)\left\|\widehat{\mathbf{h}}(n)\right\|_{p}^{1-p}\mathrm{sgn}\left(\widehat{\mathbf{h}}(n)\right)}{\left|\widehat{\mathbf{h}}(n)\right|^{1-p}+\varepsilon_{p}}\right)\right.\\&\left.\times\left(\mathbf{x}^{T}(n)\mathbf{G}(n)\mathbf{x}(n)\right)^{-1}\right)\mathbf{G}(n)\,\mathbf{x}(n)\\=\ &\widehat{\mathbf{h}}(n)+\frac{e(n)\,\mathbf{G}(n)\,\mathbf{x}(n)}{\mathbf{x}^{T}(n)\,\mathbf{G}(n)\,\mathbf{x}(n)}\\&-\gamma_{\mathrm{LP}}\left\{\mathbf{I}-\frac{\mathbf{G}(n)\,\mathbf{x}(n)\,\mathbf{x}^{T}(n)}{\mathbf{x}^{T}(n)\,\mathbf{G}(n)\,\mathbf{x}(n)}\right\}\\&\times\frac{\left\|\widehat{\mathbf{h}}(n)\right\|_{p}^{1-p}\mathrm{sgn}\left(\widehat{\mathbf{h}}(n)\right)}{\left|\widehat{\mathbf{h}}(n)\right|^{1-p}+\varepsilon_{p}}.\end{aligned}\tag{21}$$

It was found that the magnitudes of the elements in the matrix $\mathbf{G}(n)\mathbf{x}(n)\mathbf{x}^T(n)\{\mathbf{x}^T(n)\mathbf{G}(n)\mathbf{x}(n)\}^{-1}$ are much smaller

than 1 for broadband multipath channel estimation. Therefore, the update equation (21) of the proposed LP-PNLMS algorithm is rewritten as

$$\widehat{\mathbf{h}}(n+1) = \widehat{\mathbf{h}}(n) + \frac{e(n)\,\mathbf{G}(n)\,\mathbf{x}(n)}{\mathbf{x}^T(n)\,\mathbf{G}(n)\,\mathbf{x}(n)} - \gamma_{\mathrm{LP}} \frac{\left\|\widehat{\mathbf{h}}(n)\right\|_p^{1-p} \operatorname{sgn}\left(\widehat{\mathbf{h}}(n)\right)}{\left|\widehat{\mathbf{h}}(n)\right|^{1-p} + \varepsilon_p}. \tag{22}$$

Here, we neglect the effects of the matrix $\mathbf{G}(n)\mathbf{x}(n)\mathbf{x}^T(n)\{\mathbf{x}^T(n)\mathbf{G}(n)\mathbf{x}(n)\}^{-1}$ and assume that the filter order is large. Similarly to the PNLMS algorithm, a step size μ_{LP} is introduced to balance the convergence speed and the steady-state error of the proposed LP-PNLMS algorithm, and a small positive constant $\varepsilon_{\mathrm{LP}} = \delta_x^2/N$ is employed to prevent division by zero. Thus, the update function (22) can be modified to

$$\begin{aligned}\widehat{\mathbf{h}}(n+1) &= \widehat{\mathbf{h}}(n) + \mu_{\mathrm{LP}} \frac{e(n)\,\mathbf{G}(n)\,\mathbf{x}(n)}{\mathbf{x}^T(n)\,\mathbf{G}(n)\,\mathbf{x}(n) + \varepsilon_{\mathrm{LP}}} \\ &\quad - \rho_{\mathrm{LP}} \frac{\left\|\widehat{\mathbf{h}}(n)\right\|_p^{1-p} \operatorname{sgn}\left(\widehat{\mathbf{h}}(n)\right)}{\left|\widehat{\mathbf{h}}(n)\right|^{1-p} + \varepsilon_p} \\ &= \widehat{\mathbf{h}}(n) + \mu_{\mathrm{LP}} \frac{e(n)\,\mathbf{G}(n)\,\mathbf{x}(n)}{\mathbf{x}^T(n)\,\mathbf{G}(n)\,\mathbf{x}(n) + \varepsilon_{\mathrm{LP}}} - \rho_{\mathrm{LP}}\mathbf{T}(n),\end{aligned} \tag{23}$$

where $\rho_{\mathrm{LP}} = \mu_{\mathrm{LP}}\gamma_{\mathrm{LP}}$ and $\mathbf{T}(n) = \|\widehat{\mathbf{h}}(n)\|_p^{1-p} \operatorname{sgn}(\widehat{\mathbf{h}}(n))\{|\widehat{\mathbf{h}}(n)|^{1-p} + \varepsilon_p\}^{-1}$. Comparing the update function (23) of the proposed LP-PNLMS algorithm with the update function (4) of the PNLMS algorithm, we see that our proposed LP-PNLMS algorithm has the additional term $\gamma_{\mathrm{LP}}\mathbf{T}(n)$, also defined as the zero attractor, which attracts the small channel taps to zero with high probability. Moreover, the ZA strength of this zero attractor is controlled by ρ_{LP}. In other words, in our proposed LP-PNLMS algorithm, the gain matrix $\mathbf{G}(n)$ assigns a large step size to the active channel taps of the sparse channel, while the zero attractor mainly exerts the l_p-penalty on the inactive taps whose taps are zero or close to zero. Thus, our proposed LP-PNLMS algorithm can further improve the convergence speed of the PNLMS algorithm after the convergence of the large active taps.

4. Results and Discussions

In this section, we present the results of computer simulations carried out to illustrate the channel estimation performance of the proposed LP-PNLMS algorithm over a sparse multipath communication channel and compare it with those of the previously proposed IPNLMS, MPNLMS, PNLMS, and NLMS algorithms. Here, we consider a sparse channel $\mathbf{h}$ whose length N is 64 or 128 and whose number of dominant active taps K is set to three different sparsity levels, namely, $K = 2, 4$ and 8, similar to previous studies [6, 22, 25, 26]. The dominant active channel taps are obtained from a Gaussian distribution with $\|\mathbf{h}\|_2^2 = 1$, and the positions of the dominant channel taps are randomly spaced along the length of the channel. The input signal $\mathbf{x}(n)$ of the channel is a Gaussian random signal while the output of the channel is corrupted by an independent white Gaussian noise $v(n)$. An example of a typical sparse multipath channel with a channel length of $N = 64$ and a sparsity level of $K = 3$ is shown in Figure 2. In the simulations, the power of the received signal is $E_b = 1$, while the noise power is given by δ_v^2 and the signal-to-noise ratio is defined as SNR $= 10\log(E_b/\delta_v^2)$. In all the simulations, the difference between the actual and estimated channels based on the sparsity-aware algorithms and the sparse channel mentioned above is evaluated by the MSE defined as follows:

$$\mathrm{MSE}(n) = 10\log_{10}\mathrm{E}\left\{\left\|\mathbf{h} - \widehat{\mathbf{h}}(n)\right\|_2^2\right\}(\mathrm{dB}). \tag{24}$$

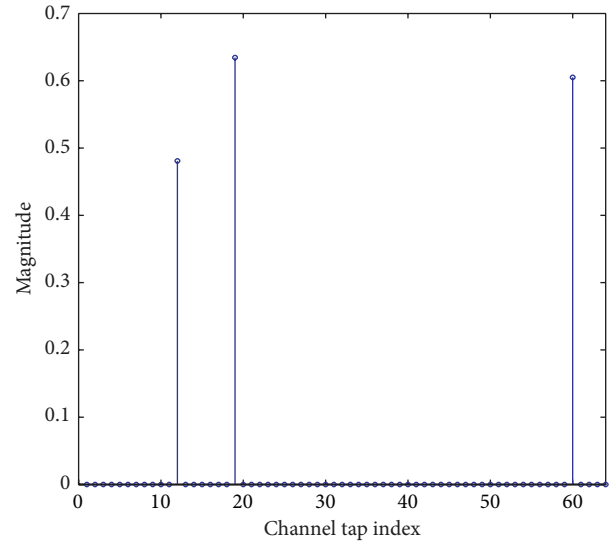
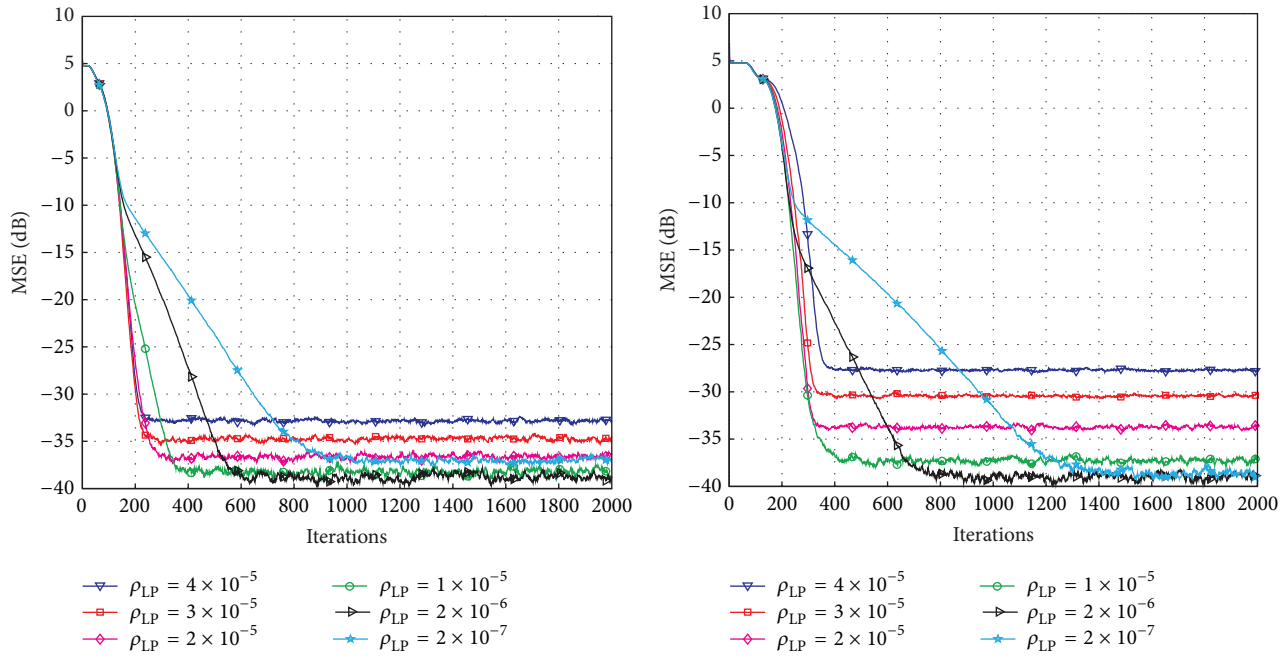


Figure 2: Typical sparse multipath channel.

In these simulations, the simulation parameters are chosen to be $\mu_{\mathrm{NLMS}} = \mu_{\mathrm{PNLMS}} = \mu_{\mathrm{IPNLMS}} = \mu_{\mathrm{LP}} = 0.5$, $\delta_{\mathrm{NLMS}} = 0.01$, $\varepsilon = 0.001$, $\alpha = 0$, $\varepsilon_p = 0.05$, $\rho_{\mathrm{LP}} = 1 \times 10^{-5}$, $\delta_p = 0.01$, $\rho_g = 5/N$, $\vartheta = 1000$, $p = 0.5$, and SNR $= 30$ dB. When we change one of these parameters, the other parameters remain constant.

4.1. Estimation Performance of the Proposed LP-PNLMS Algorithm

4.1.1. Effects of Parameters on the Proposed LP-PNLMS Algorithm. In the proposed LP-PNLMS algorithm, there are two extra parameters, p and ρ_{LP}, compared with the PNLMS algorithm, which are introduced to design the zero attractor. Next, we show how these two parameters affect the proposed LP-PNLMS algorithm over a sparse channel with $N = 64$ or 128 and $K = 4$. The simulation results for different values of ρ_{LP} and p are shown in Figures 3 and 4, respectively. From Figure 3(a), we can see that the steady-state error of the LP-PNLMS algorithm decreases with decreasing ρ_{LP} when



(a) $N = 64$ (b) $N = 128$

FIGURE 3: Effects of ρ_{LP} on the proposed LP-PNLMS algorithm.

$\rho_{LP} \geq 2 \times 10^{-6}$, while it increases again when ρ_{LP} is less than 2×10^{-6}. Furthermore, the convergence speed of the LP-PNLMS algorithm rapidly decreases when ρ_{LP} is less than 1×10^{-5}. This is because a small ρ_{LP} results in a low ZA strength, which consequently reduces the convergence speed. In the case of $N = 128$ shown in Figure 3(b), we observe that both the convergence speed and the steady-state performance are improved with decreasing ρ_{LP} for $\rho_{LP} \geq 1 \times 10^{-5}$. When $\rho_{LP} < 1 \times 10^{-5}$, the convergence speed of the LP-PNLMS algorithm decreases while the steady-state error remains constant.

Figure 4 demonstrates the effects of the parameter p. We can see from Figure 4(a) that the convergence speed of the proposed LP-PNLMS algorithm rapidly decreases with increasing p for $N = 64$. Moreover, the steady-state error is reduced with p ranging from 0.45 to 0.5, while it remains constant for $p = 0.6$, 0.7, and 0.8. However, the steady-state performance for $p = 1$ is inferior to that for $p = 0.8$. This is because the proposed LP-PNLMS algorithm is an l_1-norm-penalized PNLMS algorithm, which cannot distinguish between active taps and inactive taps, reducing its convergence speed and steady-state performance. When $N = 128$, as shown in Figure 4(b), the steady-state performance is improved as p increases from 0.45 to 0.6. Thus, we should carefully select the parameters ρ_{LP} and p to balance the convergence speed and steady-state performance for the proposed LP-PNLMS algorithm.

4.1.2. Effects of Sparsity Level on the Proposed LP-PNLMS Algorithm. On the basis of the results discussed in Section 4.1.1 for our proposed LP-PNLMS algorithm, we choose $p = 0.5$ and $\rho_{LP} = 1 \times 10^{-5}$ to evaluate the channel estimation performance of the LP-PNLMS algorithm over a sparse channel with different channel lengths of $N = 64$ and 128, for which the obtained simulation results are given in Figures 5 and 6, respectively. From Figure 5, we see that our proposed LP-PNLMS algorithm has the same convergence speed as the PNLMS algorithm at the initial stage. The proposed LP-PNLMS algorithm converges faster than the PNLMS algorithm as well as the IPNLMS and NLMS algorithms for all sparsity levels K, while its convergence is slightly slower than that of the MPNLMS algorithm before it reaches a steady stage. However, the proposed LP-PNLMS algorithm has the smallest steady-state error for $N = 64$. When $N = 128$, we see from Figure 6 that our proposed LP-PNLMS algorithm not only has the highest convergence speed but also possesses the best steady-state performance. This is because with increasing sparsity, our proposed LP-PNLMS algorithm attracts the inactive taps to zero quickly and hence the convergence speed is significantly improved, while the previously proposed PNLMS algorithms mainly adjust the step size of the active taps and thus they only impact on the convergence speed at the early iteration stage. Additionally, we see from Figures 5 and 6 that both the convergence speed and the steady-state performance of all the PNLMS algorithms deteriorate when the sparsity level K increases for both $N = 64$ and 128. In particular, when $K = 8$, the convergence speeds of the PNLMS and IPNLMS algorithms are greater than that of the NLMS algorithm at the early iteration stage, while after this fast initial convergence, their convergence speeds decrease to less than that of the NLMS algorithm before reaching a steady

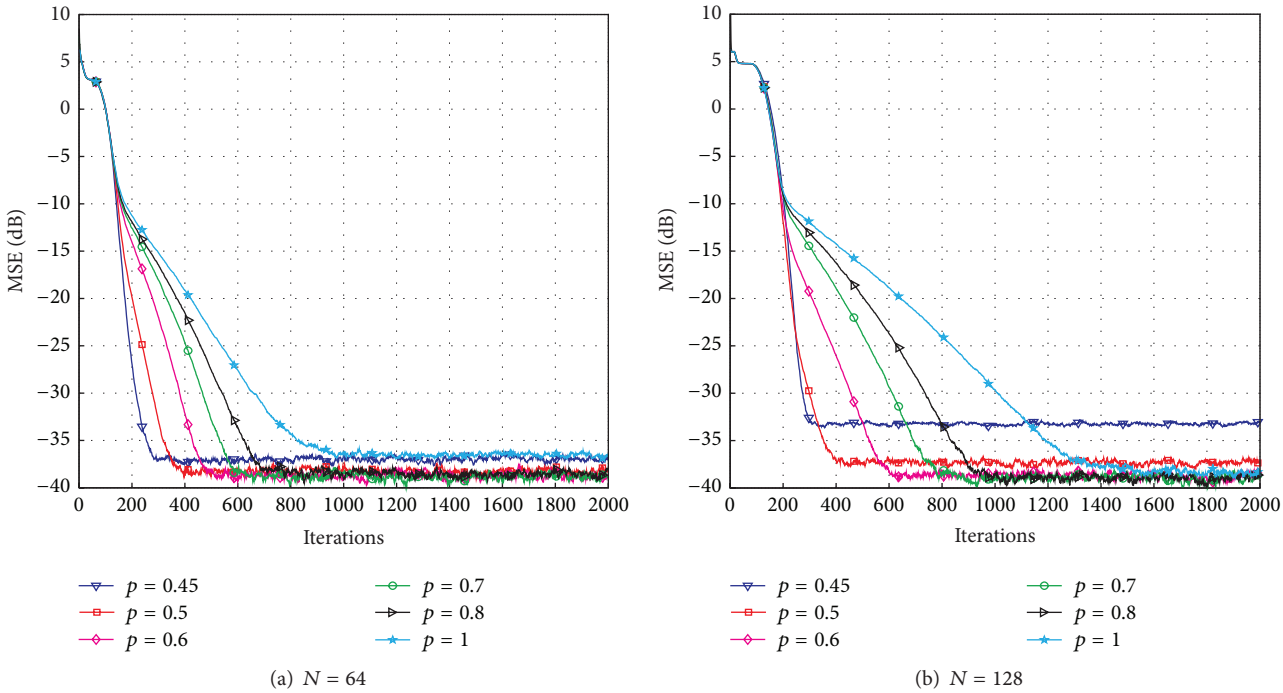


Figure 4: Effects of p on the proposed LP-PNLMS algorithm.

stage. Furthermore, we observe that the MPNLMS algorithm is sensitive to the length N of the channel, and its convergence speed for $N = 128$ is less than that for $N = 64$ at the same sparsity level K and less than that of the proposed LP-PNLMS algorithm. Thus, we conclude that our proposed LP-PNLMS algorithm is superior to the previously proposed PNLMS algorithms in terms of both the convergence speed and the steady-state performance with the appropriate selection of the related parameters p and ρ_{LP}. From the above discussion, we believe that the gain-matrix-weighted l_p-norm method in the LP-PNLMS algorithm can be used to further improve the channel estimation performance of the IPNLMS and MPNLMS algorithms.

4.2. Computational Complexity. Finally, we discuss the computational complexity of the proposed LP-PNLMS algorithm and compare it with those of the NLMS, PNLMS, IPNLMS, and MPNLMS algorithms. Here, the computational complexity is the arithmetic complexity, which includes additions, multiplications, and divisions. The computational complexities of the proposed LP-PNLMS algorithm and the related PNLMS and NLMS algorithms are shown in Table 1.

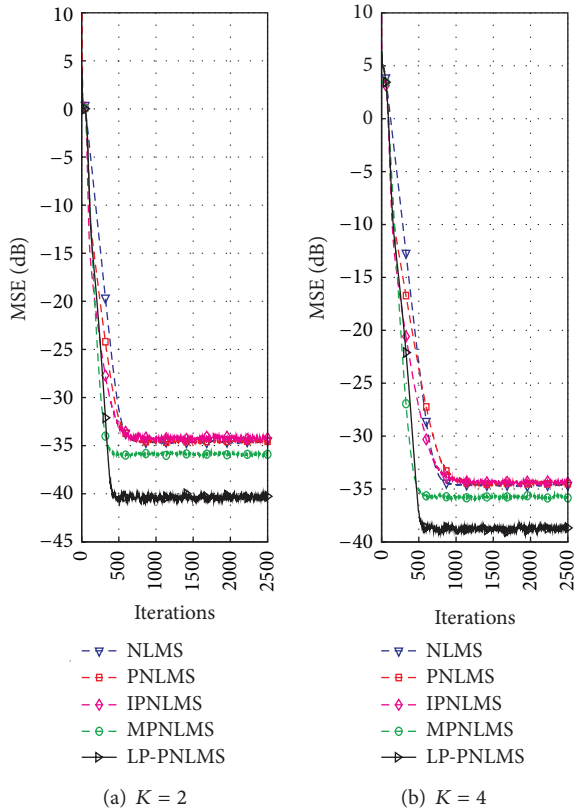
From Table 1, we see that the computational complexity of our proposed LP-PNLMS algorithm is slightly higher than those of the MPNLMS and PNLMS algorithms, which is due to the calculation of the gradient of the l_p-norm. Furthermore, the MPNLMS algorithm has an additional logarithm operation, which increases its complexity but is not included in Table 1. However, the LP-PNLMS algorithm noticeably increases the convergence speed and significantly improves the steady-state performance of the PNLMS algorithm. In addition, it also has a higher convergence speed and lower steady-state error than the IPNLMS and MPNLMS algorithms when the channel length is large.

Table 1: Computational complexity.

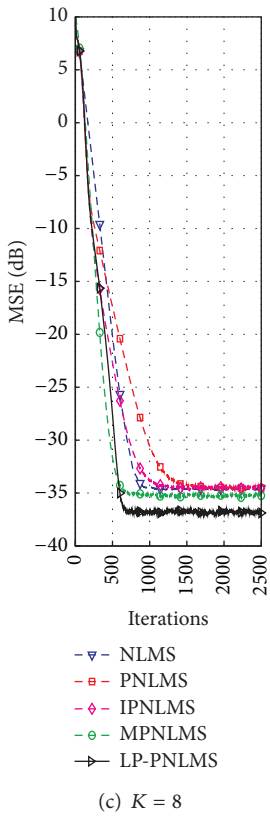
Algorithms	Additions	Multiplications	Divisions
NLMS	$3N$	$3N + 1$	1
PNLMS	$4N + 3$	$6N + 3$	$N + 2$
IPNLMS	$4N + 7$	$5N + 5$	$N + 2$
MPNLMS	$5N + 3$	$7N + 3$	$N + 3$
LP-PNLMS	$4N + 4$	$9N + 4$	$2N + 2$

5. Conclusion

In this paper, we have proposed an LP-PNLMS algorithm to exploit the sparsity of broadband multipath channels and to improve both the convergence speed and steady-state performance of the PNLMS algorithm. This algorithm was mainly developed by incorporating the gain-matrix-weighted l_p-norm into the cost function of the PNLMS algorithm, which significantly improves its convergence speed and steady-state performance. The simulation results demonstrated that our proposed LP-PNLMS algorithm, which has an acceptable increase in computational complexity, increases the convergence speed and reduces the steady-state error compared with the previously proposed PNLMS algorithms.

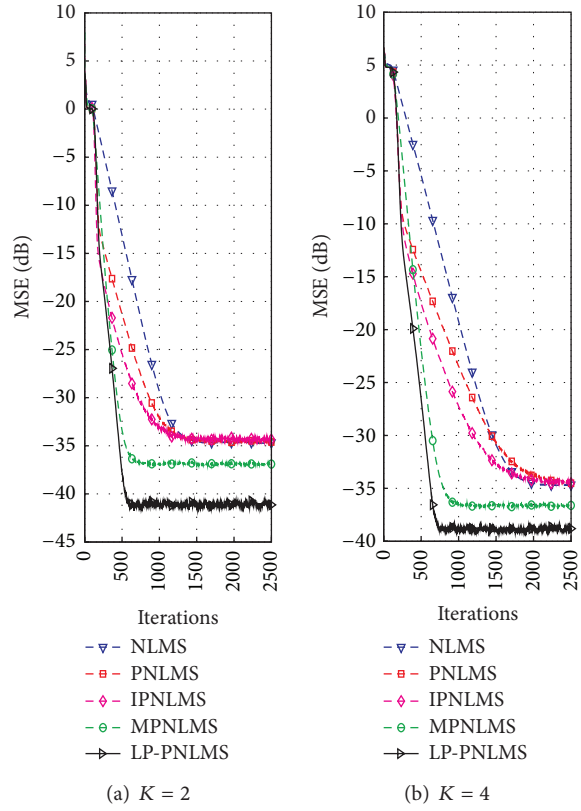


(a) $K = 2$ (b) $K = 4$

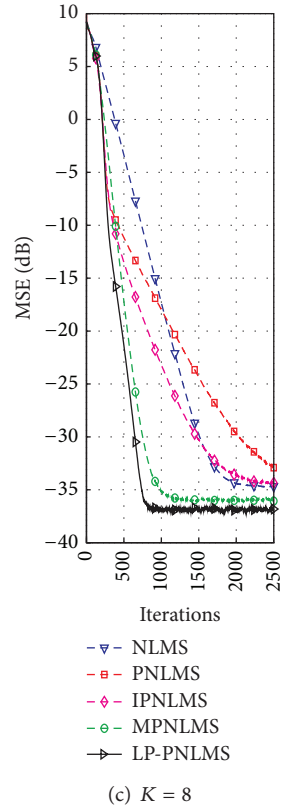


(c) $K = 8$

Figure 5: Effects of sparsity on the proposed LP-PNLMS algorithm for $N = 64$.



(a) $K = 2$ (b) $K = 4$



(c) $K = 8$

Figure 6: Effects of sparsity on the proposed LP-PNLMS algorithm for $N = 128$.

Conflict of Interests

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

References

[1] L. Korowajczuk, *LTE, WiMAX and WLAN Network Design, Optimization and Performance Analysis*, John Wiley & Sons, New York, NY, USA, 2011.

[2] J. G. Proakis, *Digital Communications*, McGraw-Hill, 4th edition, 2001.

[3] F. Adachi, D. Grag, S. Takaoka, and K. Takeda, "New direction of broadband CDMA techniques," *Wireless Communications and Mobile Computing*, vol. 7, no. 8, pp. 969–983, 2007.

[4] S. F. Cotter and B. D. Rao, "Sparse channel estimation via matching pursuit with application to equalization," *IEEE Transactions on Communications*, vol. 50, no. 3, pp. 374–377, 2002.

[5] W. U. Bajwa, J. Haupt, A. M. Sayeed, and R. Nowak, "Compressed channel sensing: a new approach to estimating sparse multipath channels," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1058–1076, 2010.

[6] H. Zayyani, M. Babaie-Zadeh, and C. Jutten, "Compressed sensing block MAP-LMS adaptive filter for sparse channel estimation and a Bayesian Cramer-Rao bound," in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP'09)*, pp. 1–6, Grenoble, France, September 2009.

[7] Md. M. Rana, J. Kim, and W.-K. Cho, "LMS based channel estimation of LTE uplink using variable step size and phase information," *Radioengineering*, vol. 19, no. 4, pp. 678–688, 2010.

[8] J. Homer, I. Mareels, and C. Hoang, "Enhanced detection-guided NLMS estimation of sparse FIR-modeled signal channels," *IEEE Transactions on Circuits and Systems I*, vol. 53, no. 8, pp. 1783–1791, 2006.

[9] B. Babadi, N. Kalouptsidis, and V. Tarokh, "SPARLS: the sparse RLS algorithm," *IEEE Transactions on Signal Processing*, vol. 58, no. 8, pp. 4013–4025, 2010.

[10] A. H. Sayed, *Adaptive Filters*, John Wiley & Sons, 2008.

[11] G. Gui and F. Adachi, "Improved least mean square algorithm with application to adaptive sparse channel estimation," *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, article 204, 2013.

[12] P. Maechler, P. Greisen, B. Sporrer, S. Steiner, N. Felber, and A. Burg, "Implementation of greedy algorithms for LTE sparse channel estimation," in *Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers (ASILOMAR '10)*, pp. 400–405, Pacific Grove, Calif, USA, November 2010.

[13] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society B*, vol. 58, no. 1, pp. 267–288, 1996.

[14] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[15] Y. Chen, Y. Gu, and A. O. Hero III, "Sparse LMS for system identification," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09)*, pp. 3125–3128, Taipei, Taiwan, April 2009.

[16] O. Taheri and S. A. Vorobyov, "Sparse channel estimation with L_P-norm and reweighted L_1 -norm penalized least mean squares," in *Proceedings of the 36th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '11)*, pp. 2864–2867, Prague, Czech Republic, May 2011.

[17] Y. Gu, J. Jin, and S. Mei, "$\ell 0$ norm constraint LMS algorithm for sparse system identification," *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 774–777, 2009.

[18] R. L. Das and M. Chakraborty, "A zero attracting proportionate normalized least mean square algorithm," in *Proceedings of the 2nd Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC '12)*, pp. 1–4, Hollywood, Calif, USA, December 2012.

[19] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 5, pp. 508–518, 2000.

[20] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in *Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing*, pp. 1881–1884, Orlando, Fla, USA, May 2002.

[21] H. Deng and M. Doroslovački, "Improving convergence of the PNLMS algorithm for sparse impulse response identification," *IEEE Signal Processing Letters*, vol. 12, no. 3, pp. 181–184, 2005.

[22] A. W. H. Khong and P. A. Naylor, "Efficient use of sparse adaptive filters," in *Proceedings of the 40th Asilomar Conference on Signals, Systems, and Computers (ACSSC '06)*, pp. 1375–1379, Pacific Grove, Calif, USA, November 2006.

[23] L. Liu, M. Fukumoto, S. Saiki, and S. Zhang, "A variable step-size proportionate NLMS algorithm for identification of sparse impulse response," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 93, no. 1, pp. 233–242, 2010.

[24] M. Nekuii and M. Atarodi, "A fast converging algorithm for network echo cancelation," *IEEE Signal Processing Letters*, vol. 11, no. 4, pp. 427–430, 2004.

[25] L. Liu, M. Fukumoto, and S. Saikfi, "An improved mu-law proportionate NLMS algorithm," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '08)*, pp. 3797–3800, Las Vegas, Nev, USA, April 2008.

[26] C. Paleologu, J. Benesty, and S. Ciochină, "Sparse adaptive filters for echo cancellation," *Synthesis Lectures on Speech and Audio Processing*, vol. 6, pp. 1–124, 2010.

[27] K. Dogancay, *Partial-Update Adaptive Signal Processing: Design, Analysis and Implementation*, Academic Press, Oxford, UK, 2008.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 738206, 11 pages
http://dx.doi.org/10.1155/2014/738206

Research Article

A Fast Overlapping Community Detection Algorithm with Self-Correcting Ability

Laizhong Cui, Lei Qin, and Nan Lu

College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China

Correspondence should be addressed to Laizhong Cui; cuilaizhong@gmail.com

Received 27 December 2013; Accepted 4 February 2014; Published 13 March 2014

Academic Editors: H. R. Karimi, X. Yang, Z. Yu, and W. Zhang

Due to the defects of all kinds of modularity, this paper defines a weighted modularity based on the density and cohesion as the new evaluation measurement. Since the proportion of the overlapping nodes in network is very low, the number of the nodes' repeat visits can be reduced by signing the vertices with the overlapping attributes. In this paper, we propose three test conditions for overlapping nodes and present a fast overlapping community detection algorithm with self-correcting ability, which is decomposed into two processes. Under the control of overlapping properties, the complexity of the algorithm tends to be approximate linear. And we also give a new understanding on membership vector. Moreover, we improve the bridgeness function which evaluates the extent of overlapping nodes. Finally, we conduct the experiments on three networks with well known community structures and the results verify the feasibility and effectiveness of our algorithm.

1. Introduction

Community structure is an important field in complex networks research. In the traditional social network, Newman et al. discover the community structure [1–4], which is a group of nodes with dense internal links and sparse connections between groups [3–6]. Later, from the metabolic networks [7] to the large-scale WWW webpage links [8], they are all community structures.

In the exploration of community structure, the crisp division [9, 10] is put forth first by scholars; that is, a node belongs to only one community. In reality, networks are built in different relations, and nodes can be shared by many communities. For example, in human relationship, the relationship of two people may be family, friend, and colleague. When Palla et al. point out the overlap feature of the community [3], a number of soft division algorithms are designed to detect the overlapping community structure, and two main effective means are clique [3, 11–14] and optimization theory [14–16]. The methods based on clique have the high accuracy, but the process is complex. While the optimization algorithms choose the appropriate object function and get a lower complexity. But when and how to finish are ambiguous. Meanwhile, whether a vertex is the overlapping node is not a clear understanding in academic.

In this paper, we propose a fast overlapping community detection algorithm with self-correcting ability through the following contributions. First, we introduce new features of modularity as a new evaluation measurement and explore the advantage of the new weighted modularity in structure through combining the cohesive and density synthetically. Second, we propose three test conditions for overlapping nodes and present a fast overlapping community detection algorithm with self-correcting ability, which consists of two processes. Under the control of overlapping properties, the complexity of our algorithm tends to be approximate linear. Third, we give a new understanding on membership vector to improve the bridgeness function which evaluates the extent of overlapping nodes. To evaluate the feasibility and effectiveness, we implement our approach on three existing networks with the well-known community structures.

The rest of the paper is organized as follows. In Section 2, we describe the details of our new weighted modularity. In Section 3, we propose our fast overlapping community

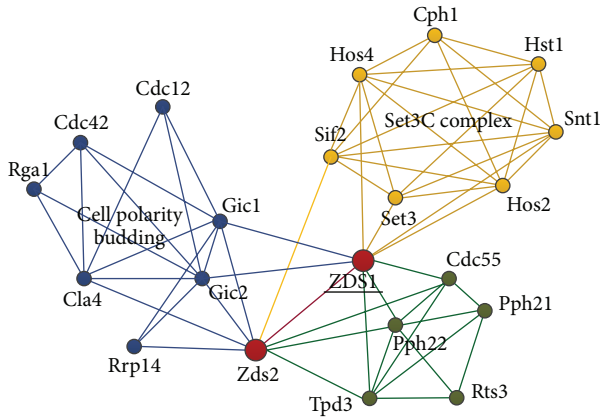


Figure 1: The community structure of Protein reaction network.

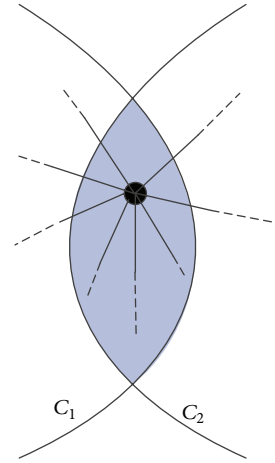


Figure 2: The common overlap.

detection algorithm. In Section 4, we explore the estimation to the effect of overlapping node. The experimental results are discussed in Section 5. Finally, Section 6 concludes our work.

2. Modularity

2.1. The Standard of Overlap. The overlapping node is the vertex that belongs to more than one community. After the analysis on the well-known networks whose structures are also known, we propose three conditions to judge the overlapping node in this paper. In particular, all those are not disrelated. The priority is from the top down, and some nodes will be in accord with several conditions. If a node meets one of the conditions, it shall be an overlapping node.

2.1.1. Addition in Modularity. This is the most common case. Referring to the view of Lázár et al. [17], the node contribution is positive to their communities. The overlapping nodes link many adjacent vertex and belong to less communities (two is general), just as vertex Zds2 shown in Figure 1. In addition, if the adjacent nodes of overlap contain overlapping nodes, they can expand to overlapping region [17] as shown in Figure 2. With the help of this region, all nodes in the sharing area make positive contribution to the belonging communities. The criterion is used in the variation caused by the nodes joining.

2.1.2. Strengthen the Internal Connection. The analysis on the known networks reveals that some core nodes in community may reduce the holistic modularity. Though they have many adjacent nodes, the gain in inner connection is less than the outer connection, which results in the decrease of modularity. However, the internal connection of community is strengthened. In the sparse network, it performs as the addition in density besides the stationarity in modularity whose threshold value is 0.015 in Karate club network as shown in Figure 3. And in the dense network, the proportion of adjacent nodes in community is more than 1/3, according to the research on the Protein reaction network.

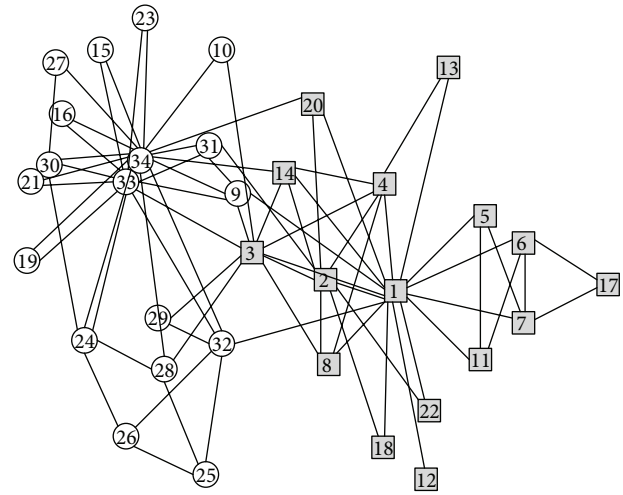


Figure 3: The community structure in Karate club network.

2.1.3. The Average Distribution of Belonging Factor. In the networks, the belonging factors of some nodes belong to their communities impartially, which means the adjacent nodes are distributed to the communities averagely. If the degree of node is high, it can be in accord with the previous two cases, just as the Zds2 in Figure 1. Here, the average distribution is not absolute. To be in accord with the membership number, this paper sets a dynamic threshold. Calculating the absolute value of deviation with average distribution, if the sum is less than the dynamic threshold, it is all right.

Definition 1. The average distribution of belonging factor should meet the condition in (1), in which a_{iC_c} is the belonging factor of node i in community C_c and n is the membership number:

$$\sum_{c=1}^{n} \left| a_{iC_c} - \frac{1}{n} \right| \leq \frac{1}{2n}. \quad (1)$$

In (1), $1/n$ is the value of average distribution. With the consideration of the distribution in real network and the

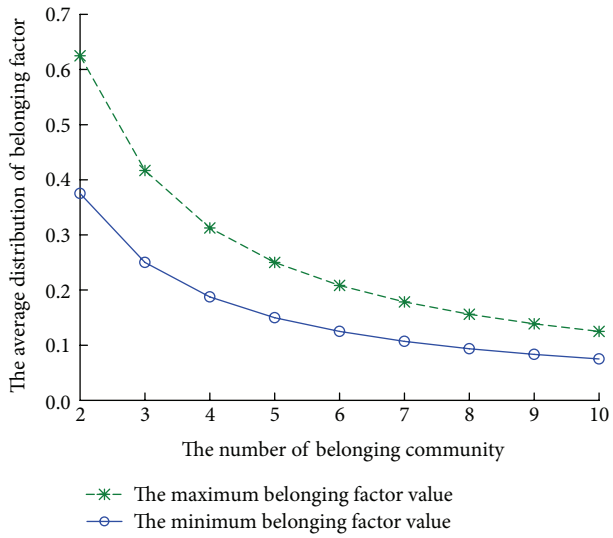


Figure 4: The graph of maximum and minimum value in average distribution.

accepted empirical value, the dynamic threshold is set as $1/2n$. With the increasement in membership number, the threshold decreases gradually, and the average feature is more and more obvious in Figure 4.

2.2. The New Weighted Modularity. Modularity is the measurement of evaluation and is the preferred object function. Considering various definitions of the past types [18], it summarizes three features after comparing the advantages and disadvantages of each type, which are the fairness, rationality, and independence.

There is the defect in the actual modularity [19], and it disobeys the fairness and rationality. As for the other cases, such as modularity based on node contribution [17] and the overlapping modularity, the weaknesses are discussed [18]. The fitness function [15] handles the inner and outer nodes of community, respectively, defined as (2). For a given community C_i, the inner connection is $k_{C_i}^{\text{in}}$, the outer is $k_{C_i}^{\text{out}}$, and α is the regulatory factor to control the size with the default value 1.0:

$$f_{C_i} = \frac{2k_{C_i}^{\text{in}}}{\left(2k_{C_i}^{\text{in}} + k_{C_i}^{\text{out}}\right)^{\alpha}}. \tag{2}$$

The fitness function is unfit for modularity, too. That is because it weakens the inner connection which leads to recognize some structures unsuccessfully. For example, the complete graph and the ring structure are shown as Figure 5. They get the same value 1.0, but the structure is quite a bit different. In some subjects like biology or macromolecule, some special structure determines the function.

From the above discussion, in view of network density, we propose a new weighted modularity, which is composed of the density and cohesion.

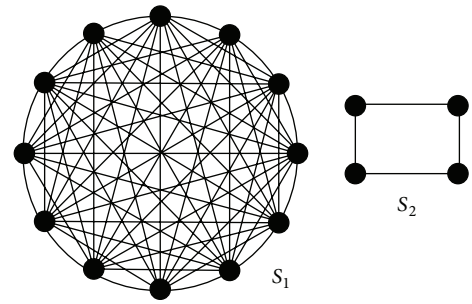


Figure 5: The complete graph S_1 and ring structure S_2.

Definition 2. The new weighted modularity is defined as (3), in which the allocation parameter β meets $0 < \beta \leq 0.5$ and, for the community C_i, n_{C_i} is the number of inner nodes:

$$Q_{C_i} = \beta \frac{2k_{C_i}^{\text{in}}}{n_{C_i}\left(n_{C_i} - 1\right)} + \left(1 - \beta\right) \frac{2k_{C_i}^{\text{in}}}{2k_{C_i}^{\text{in}} + k_{C_i}^{\text{out}}}. \tag{3}$$

The modularity of whole network is the average of each community. The research on several typical networks shows that the density of whole network is in a low level which is varying between 0.10 and 0.30. For example, Karate club network [1, 2] is 0.139, Protein reaction network [3] is 0.290, and Dolphins interaction network [2] is 0.084. And here, the rule of parameter allocation β obeys Pareto law that is twenty-eighty law. The cohesion gets the 80% weight, and the density gets the 20% weight. With the expansion on the size of networks, the links between nodes are weaker and weaker, and therefore the contribution of density is reduced in modularity.

Test on the parameter allocation reveals that when β is getting 0.20, the minimum modularity of communities approaches 0.750 in the known networks. However, there is no regular pattern on the other allocation plan and the value is very discrete. As shown in Figure 6, the communities 1–3 are the Protein reaction network, and the communities 4 and 5 are the Karate club network. The detailed communities information of each cluster is shown in Table 1.

Set the allocation parameter β as 0.20, and the modularity of communities are as follows.

Through analyzing the test in the known networks, it gets the minimum value of the community structure. In particular, the threshold is for the rough judgment which is not the only condition. In addition, after the node joining, the impacts on the original network, namely the smoothness of the modularity, need to be considered, namely the smoothness of the modularity. The modularity threshold is the last condition in judging, and the detailed introduction is given in next section.

3. Fast Overlapping Community Detection Algorithm with Self-Correcting Ability

The traditional community detection methods [3, 11–13, 20] are visiting the nodes repeatedly. But after studying on many

Table 1: The detailed communities information in networks.

Network	Community	Node	Inner link	Outer link	Modularity
Protein reaction network	Community 1	9	21	11	0.751
	Community 2	7	16	11	0.748
	Community 3	8	24	7	0.870
Karate club network	Community 4	18	35	10	0.746
	Community 5	16	33	10	0.750

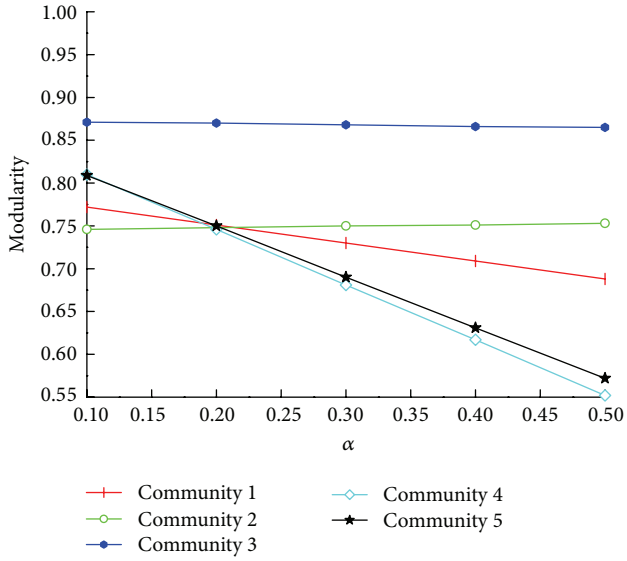


Figure 6: The allocation parameter test graph.

overlapping communities structure in some different types, we find out that the vast majority of nodes belong to only one community and the proportion of overlapping nodes is very small. For example, the proportion of overlapping nodes is 3/34 in Karate club network, 2/21 in Protein reaction network, 3/62 in Dolphins interaction network. Visiting the irrelevant nodes again and again results in the reduction of the algorithm efficiency. If we can distinguish the probable overlap nodes or the nonoverlapping nodes, the complexity can be cut down. The algorithm proposed in this paper is based on this idea. It labels the nodes by different property and removes the unrelated nodes to reduce the unnecessary visiting. The fast overlapping community detection algorithm with self-correcting ability is adopted in two stages. The first stage is the initial community discovery, and the second stage is the error detection and correction for specific nodes.

3.1. Initial Community Detection Algorithm

3.1.1. Raw Community Detection Algorithm. Referring to the process of local modularity [18, 21], our algorithm selects the root vertex firstly and visits the adjacent nodes in next layer. Then, our algorithm let the eligible nodes join the community and repeats those two steps until there are no qualified nodes. Some studies have verified that a majority of nodes belong to only one cluster in the networks. So, once the node has belonged to the community, it is unnecessary to be visited again. On the basis of this thought, we set each node with two attributes, which are isVisited and isLocated with default value false. They indicate whether a node is visited and located to the community and control the beginning root of the cluster and the range in the next access layer, respectively. Regulating the attributes of nodes, the algorithm proposed in this paper greatly reduces the amount of the adjacent nodes and cuts down the time complexity.

Raw community detection algorithm is composed of the following steps.

(1) Pick a node randomly whose isVisited attribute is false as the root of the community, and get the core of the original community.

(2) If the count of nodes in the community is greater than 3, install the community model and set the isVisited attribute to be true for all original nodes. Otherwise, set the isVisited of root vertex to be true; then return step 1.

(3) Get the adjacent nodes set whose isLocated attributes are false on the basis of parameter nodes which are going to access. And if the count is 0, go next; otherwise, turn to step 5.

(4) If the count of nodes in the community is not less than 5, check whether the isLocated attributes are true, and output the original community, and then return to step 1. Otherwise, return to step 1 directly.

(5) Access each node in the adjacent nodes set in turn, and set its isVisited attribute value is true.

(6) If a node meet the conditions, add it to the current community, and update the community model, and then put it to the next layer to access. Otherwise, return to step 5.

(7) If all the nodes are calculated, return to step 3 with the next layer nodes set.

Here, the conditions for a node joining the community are described as follows. Assume that the node joins the community, it meets one of the following conditions: (1) it brings gain in new modularity; (2) it gets addition in density and stable; (3) the rate linked with vertex in community is not less than threshold value (1/3); (4) the modularity is greater than threshold and stable. Theoretical research proves that random selection in nodes has nothing to do with the community structure of network. In other words, every node must belong to a certain community [3]. If root vertex cannot form the community, sign the isVisited attribute to be true,

and go on to seek another available root vertex until the core is found.

In this progress, it builds the corresponding community model, which records the detailed information, such as the inner nodes and edges, and the outer edges. If a new node joins the community, it needs to update the community model immediately, which avoids the repetitive computation and lessens the complexity in time.

Moreover, the amount of expansion is not unlimited. From steps 3 to 7, even if there are coincident nodes every time, due to the six-degree theory [8], the diameter of the community is less than 6 and the cycle index is limited, too.

After discovering the community, the algorithm needs to confirm the isLocated attribute. The studies show that the threshold of most nodes' belonging factor in community is 2/3. But in this paper, the threshold of belonging factor is 3/4, and the rate of overlapping edges in all the adjacent edges is less than 1/4. Our algorithm takes a strict criterion to prevent missing the possible overlapping nodes, which is convenient for the error correction algorithm in the next step, just as the number 31 node in Karate club network.

3.1.2. Redistribution Algorithm for Unallocated Nodes. Some studies on the known networks reveal that some nodes have accessed (isVisited = true) in the raw detection stage but failed to be assigned to any detected communities. The reasons are listed as follows: (1) high threshold modularity; (2) close connections among some nodes which form the structure like analogous triangle and any individual vertex cannot meet the requirements. For example, the number 25, number 26, and number 32 nodes are unable to form an independent cluster and need to join the community in union. So, it is necessary to execute the redistribution algorithm for unallocated nodes, which ensures every node belongs to cover.

The way to acquire the unallocated nodes set is calculating the subtraction between the beginning nodes and allocated nodes. For each unallocated node, the flow of the process is described as Algorithm 1.

Owing to the complexity of networks, there may be chain effect. For example, the adjacent nodes connected in sequence just form a chain. To deal with this case, the first node is allocated to the community, and the others are isolated nodes, which will not be participated in the next community detection.

3.2. Error Detection and Correction Algorithm for Specific Nodes. The error detection and correction algorithm aims to recognize and check the overlapping nodes, which ensure the accuracy of the result. Here, the specific nodes are those whose isLocated attribute is false. In the process of initial community detection, the timing of nodes joining the community is different, and some core nodes are put in the cluster first. The other will not be identified completely because of missing the information of adjacent nodes. In the following redistribution process, the unallocated nodes decrease the membership value to the community, which may lead to wrongly label to other community. However, the isLocated attribute of wrong classified nodes is false, too. It is the minimum range to execute error correction algorithm on those nodes whose isLocated attribute is false. The researches on some known network show that the more evident the community structure is, the less the specific nodes are. In Karate club network, it is 11/34, in Protein reaction network it is 5/21, and in Dolphins interaction network it is 8/62.

In our algorithm, for every node, the procedure is described as follows.

(1) Get the adjacent communities list. If they exist, go next. Otherwise, go to step 5.

(2) Select an adjacent community, and verify whether the node meets the conditions to join the community (referring to distribution algorithm), and then decide to join or continue.

(3) If all the adjacent communities are tested, then check whether it is equal distribution. If it is, join each adjacent cluster. Otherwise, go next.

(4) If the type of joining community is equal distribution, return. Otherwise, continue to go.

(5) Get the belonging community set of the node. If the count is 1, return. Otherwise, go next.

(6) Choose an unverified community; recalculate the belonging factor which is linked with the community. If it is bigger than the threshold, return.

(7) Calculate the modularity variation when removing the node from the cluster. If it is positive, remove the node, return. Otherwise, directly return.

After the initial community detection, the information of nodes membership is completed mostly. The experiments reveal that some overlapping nodes interplay, which would change the node property. That is to say, a new joining node will bring its unallocated adjacent nodes to the same community, and the allocated nodes may be overlapping nodes and expand the overlap region, such as the number 10 and number 3. Therefore, the mutual linking nodes should be extracted, and the error detection and correction algorithm should be executed once again. It could clear up the possible wrong division, which makes the partition reasonable and steady.

Moreover, since the error detection and correction algorithm aims to the specific nodes, when expanding the nodes to the whole network, it is able to detect the validity of other community detection algorithms. If there is no change in node membership, they are just steady and accurate.

4. Estimation to the Effect of Overlapping Node

4.1. Original Bridgeness. Community modularity is independent with the partition pattern, whether it is hard or crisp. But for the overlapping and nonoverlapping nodes, the roles are different in the network. It needs bridgeness [9] function to evaluate the position and importance of the overlap in the network, and the membership distribution is a major factor. The previous studies suggested that the sum of membership

```
(1)  communities = getCommunities(); /* get the communities of adjacent nodes */
(2)  if (communities. Number == 0 then
(3)     /* if the number is 1 */
(4)     Add(community); /* join the community */
(5)     Return;
(6)  else
(7)     for communities do
(8)        /* for each community */
(9)        if matchConditions then
(10)          /* judge whether node match the conditions */
(11)          Add(community); /* join community */
(12)       end if
(13)       end for
(14)       if averageDistribution() then
(15)       /* if it is equal istribution */
(16)       AddToAll(communities); /* join each community */
(17)       Return;
(18)    end if
(19)    if noAdd then
(20)       /* if it doesn't match the above */
(21)       addMaxMembership(); /* join the community getting maximum belonging factor */
(22)    end if
(23) end if
```

Algorithm 1: The redistribution algorithm for unallocated nodes.

factor is 1.0 [17, 22]. However, there are limitations with this suggestion. In the overlapping communities, the connections among overlaps form the overlapping edges and are the member of multiple communities. So it is calculated more than once and the sum of node membership can be greater than 1.0. For example, the membership vector of Zds1 in Protein reaction network is [0.3, 0.4, 0.4] and the sum of them is 1.1.

Definition 3. The sum of node membership vertex meets the condition in (4), in which we assume that the number of belonging communities is c:

$$\sum_{i=1}^{C_c} a_{iC_c} \geq 1.0. \tag{4}$$

For each node, bridgeness means the degree of sharing in different communities. Nepusz et al. define that it is 0 when the node belongs to only one cluster. And it is 1.0 when sharing equally by the belonging communities. On the basis of membership vertex $[a_{i1}, a_{i2}, \ldots, a_{ic}]$ of overlapping node i, after setting the uniform distribution $[1/c, 1/c, \ldots, 1/c]$ as reference vertex, they define the bridgeness [9] as

$$b_i = 1 - \sqrt{\frac{c}{c-1}\sum_{r=1}^{c}\left(a_{iC_r} - \frac{1}{c}\right)^2}. \tag{5}$$

4.2. Improved Bridgeness. The distribution of belonging factors determines the status in network. The more approximate to the uniform distribution it is, the greater the effect is. However, the membership number is a positive and significant element. In addition, the degree of node itself is not an ignorable element. Obviously, Nepusz et al. overlook its own factors. If two nodes all conform to uniform distribution, the bridgeness is 1.0. So it cannot indicate the importance. Moreover, the node contribution of overlap can be positive in more than one community, and $1/c$ fails to express the actual average value of the uniform distribution. So, we set the average value as all the belonging factors. Considering the membership number is lesser than the degree of node, we introduce the improved bridgeness as follows synthetically.

Definition 4. Improved bridgeness is defined as follows, in which k_i is the degree of node i, and $\overline{a}_i$ is the value of the actual uniform distribution:

$$b_i = 1 - \sqrt{\frac{c}{c-1}\sum_{r=1}^{c}\left(a_{iC_r} - \overline{a}_i^2 + \frac{1}{c^2} + \frac{1}{k_i}\right)}. \tag{6}$$

As shown in (6), the greater the membership number is, the higher the degree is, and the more similar to uniform distribution the membership vertex is. The sum of variance is less, and then the value of bridgeness is greater, which signifies the important standing in network. The detailed results of comparison and analysis are demonstrated in the next experiments.

5. Experiment Results and Analysis

In this section, we evaluate our algorithm with the community structures of three well-known networks, which, respectively, are Karate club network, Protein reaction network, and Dolphins interaction network. Karate club network [1, 15] is the community structure network initially found

by Newman, which represents the traditional social research network. Protein reaction network [3] is a network composed of protein metabolism, on behalf of the emerging biology research network, and Palla et al. find the overlapping characteristics of the community through it. Dolphins interaction network [15] is a network built according to interaction information of waters bottlenose dolphin living in New Zealand, which belongs to natural science research field, and many scholars take it as a research subject.

5.1. Karate Club Network. Karate club network is the classic interpersonal relationship network, in which 34 members constitute 78 connections. Since the disagreements between members lead to divide into two pies, the network is divided into two distinct communities. In traditional hard classification model, the node can only belong to one community. However, after utilizing our algorithm on this network, we found that three nodes meet the criteria of overlapping node. Since overlap is an important characteristic of complex networks, through analyzing the network structure, the overlap model is more in accord with the actual situation.

During our algorithm implementation process, firstly, it chooses the root node of the community, which is "1," after three extensions, no subsequent nodes can join in, the original community is found; then it selects the root node, which is the "15"; it ends after three times external extension and completes the found of another community; the obvious overlap nodes {"9", "31"} are identified. Original community discovery process is described as follows:

Com1: "1" → "11", "14", "22", "20", "13", "18", "8", "9", "2", "3", "4", "5", "6", "7" → "12", "17", "31".

Com2: "15" → "33", "34" → "16", "19", "21", "23", "24", "28", "30", "31", "9" → "27".

After forced distribution, the set of the unallocated nodes is {"32", "25", "26", "29", "10"} and none of overlapping nodes is classified by mistake. After the initial community distribution, 29 nodes determine the belonging community.

In the error detection and correction algorithm, the set of the detected nodes is {"34", "3", "32", "9", "31", "28", "29", "26", "26", "25", "20", "10"}. Since the adjacency node has not been fully allocated in the process of the initial distribution, most of the nodes lack of adjacency information reference and are unable to determine isLocated = true. In view of the connected closely node {"3", "9", "10"}, the theoretical analysis has pointed out this problem that wrong classification may result in the change of the node properties. So, the error detection and correction algorithm should be performed again for such a node, which can eliminate the unreasonable factors and make the classification tend to be stable.

Figure 7 is the final community structure found by our algorithm, which is in accordance with the standard distribution and includes the red solid nodes {"3", "9", "31"}, namely, the overlapping nodes. More information of overlapping nodes is shown in Table 2. Compared with number 9 and number 31, both of them belong to two communities. The variance of the former is less than number 31, and the degree is also greater. However, the original bridgeness of number 9 is smaller, which is irrational. And the improved bridgeness displays the difference in nodes, in accord with their own status in the network.

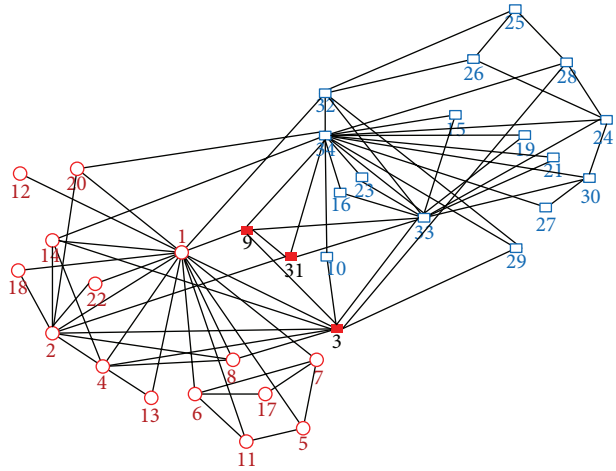


Figure 7: The community structure found by our algorithm in Protein reaction network.

As seen from the membership value, in Karate club network, the sum of overlap node membership degree is greater than 1.0, and each node can increase modularity for belonging community. In addition, the cumulative difference of the absolute value of the node 3 and node 31 membership vertex value and average distribution is less than or equal to a quarter, which are also in accord with the condition of average distribution. Therefore, it also verifies the principle and sequence of overlapping node determination conditions.

5.2. Protein Reaction Network. Protein network is built according to metabolism response relationship between the biological protein, containing 21 nodes and 61 sides. It is a typical overlapping community network, the community structure of which is obvious. Through running our algorithm for finding original community, all of the communities are identified, and the nodes are all classified correctly, including high overlapping nodes. The redistribution algorithm needs not to run; thus our algorithm quickly found the overlapping community, which validates the efficiency of our algorithm.

Let us focus on the implementation process of our algorithm. The root node of each community and its subsequent extension process are described as follows:

Com1: "cdc12" → "gic2", "cla4", "gic1" → "cdc42", "rga1", "rrp14", "zds2" → "zds1".

Com2: "cph1" → "snt1", "sif2", "hst1", "hos2", "hos4" → "set3", "zds1".

Com3: "pph21" → "pph22", "tpd3", "rts3", "cdc55" → "zds1", "zds2".

Figure 8 illustrates the three communities found by our algorithm, which is labeled by different shapes and colors. The structure is the same as the standard classification, and the solid red nodes {"zds1", "zds2"} are the overlapping nodes. The detailed information is shown in Table 3. In particular,

Table 2: The detailed information of overlapping nodes in Karate club network.

Overlapping node	Degree	Membership number	Belonging factor	Original bridgeness	Improved bridgeness
3	10	2	0.60 0.50	0.86	0.40
9	5	2	0.60 0.80	0.55	0.30
31	4	2	0.50 0.75	0.65	0.25

Table 3: The detailed information of overlapping nodes in Protein reaction network.

Overlapping node	Degree	Membership number	Belonging factor	Original bridgeness	Improved bridgeness
Zds2	9	2	0.56 0.44	0.89	0.39
Zds1	10	2	0.30 0.40 0.40	0.88	0.53

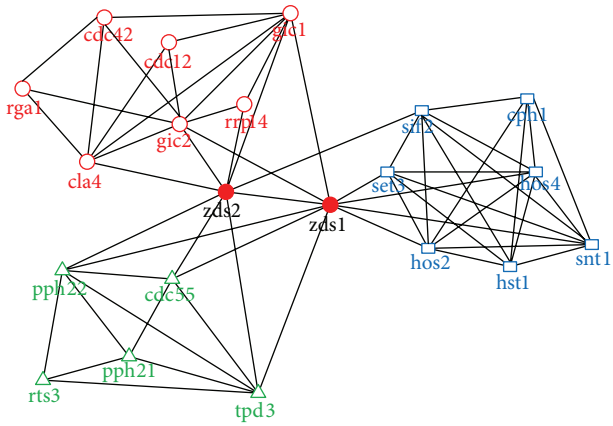


Figure 8: The community structure found by our algorithm in Protein reaction network.

they are in accord with the three conditions of overlap, which verifies the effectiveness of our proposed conditions again.

Protein reaction network also demonstrates the irrationality of the original bridgeness. Zds1 node degree and community membership number are all greater than Zds2. But the original bridgeness is still lower than Zds2, and the difference of numerical value is very small, while our improved bridgeness avoids this defect, which is more in line with the node's position in the network community.

5.3. Dolphins Interaction Network. Dolphins interaction network is established according to the dolphin interaction information. Its community structure found by our algorithm is shown as Figure 9, in accord with the actual situation. The difference in number of nodes in communities is big, which is 44 and 21, respectively. Through running the algorithm, we found that from the start of the root node "0," the network cycle needs to extend 7 times to find the large community, and it seems not to be in accord with the small world network model. Through deep analysis, the reason is that some nodes link more closely in the network. If there are no adjacent nodes to join in, the adjacent information is missing and nodes are delayed to join the community, which cause that they are repeatedly visited. This is not in conflict with the six degrees theory in network, because it does not mean extending the network to the seventh layer from the root node. The detailed extension process of raw community is described as follows:

Com1: 0 → "14", "47", "42", "15", "10", "40" → "2", "24", "30", "52", "55", "7" → "19", "28", "29" → "18", "20", "35", "45", "51", "8" → "11", "21", "23", "3", "36", "37", "4", "44", "50", "59" → "16", "33", "34", "38", "39", "43", "61" → "12", "46", "49", "53", "58".

Com2: 13 → "5", "6", "41", "32", "17", "57", "9", "54" → "1", "56", "60" → "19".

Between the starting node 0 in the community Com1 and the last joined node 58 in the community Com1, the shortest distance among them is only 3, in line with the feature of the small world network. Unallocated node set {"27", "25", "26", "22", "31", "48"} is allocated correctly, and the specific nodes {"17", "1", "27", "7", "19", "25", "26", "39"} are achieved stability after completing the error detection and correction algorithm.

The detailed information of overlapping nodes is shown in Table 4. The set of found overlap nodes is {"19", "39", "7"}, in which {"19", "7"} are in accord with multiple overlap conditions. In addition, node 39 conforms to standard uniform distribution, fairly connecting the two communities. Through the comparison of the original and improved bridgeness, since the original bridgeness only considers the membership values, it is unreasonable distinctly, which is illustrated by node 39.

5.4. Results Analysis and Discussion

5.4.1. Multiple Constraints of the Algorithm. The other community found that optimization algorithms take the maximize objective function value as the end condition. The

Table 4: The detailed information of overlapping nodes in Dolphins interaction network.

Overlapping node	Degree	Membership number	Belonging factor	Original bridgeness	Improved bridgeness
19	4	2	0.50 0.75	0.65	0.25
39	2	2	0.50 0.50	1.00	0.14
7	5	2	0.60 0.60	0.80	0.33

Table 5: The proportion of overlapping nodes and nodes of each stage in the networks.

Community structure	Overlapping proportion	Raw community detection proportion	Redistribution proportion	Error detection and correction proportion
Karate	0.09	0.85	0.15	0.32
Protein	0.10	1.00	0.00	0.24
Dolphins	0.05	0.9	0.10	0.13

initial community detection algorithm proposed in this paper is different and based on extracting the overlapping node features. Multiple conditions and multiple thresholds are constrained to find natural communities, which contains situations as follows: (1) the basic modularity increases; (2) density increases with stability; (3) the connection is greater than the community size threshold; (4) the modularity is greater than the specified threshold with stability; (5) the belonging factors are in accord with uniform distribution. These conditions form an access priority according to the judgement order, and high computational complexity would inspect in the final. Through the reasonable arrangement of the condition priority, it avoids the unnecessary calculation and reduces the complexity.

5.4.2. Operating Efficiency of the Algorithm. In the process of our algorithm running, through controlling the attribute of isLocated, it gradually shrinks the expansion space of the adjacent available nodes and cuts down the repeated access of the nodes, which improve the operation efficiency of our algorithm. For nonoverlapping nodes, theoretical visit is only one time. For the possible overlapping nodes, it only needs to visit and verify the relevant adjacent communities. The attributes are to classify nodes, and it can avoid the repeated visit and compute for the irrespective node effectively.

5.4.3. Strategy of the Algorithm. In the process of discovering natural communities, by establishing community model and updating community information in real time, it avoids the repeated compute community information when the nodes join the community. Most of the time is spent in the process of finding natural community. The subsequent error detection and correction algorithm is a supplement. Since the involved node is less, the complexity of the calculation is lower than community discovery process. So the overall complexity is approximate linear. It shows the node proportion information in each stage during our algorithm running in Table 5.

In each period of our whole algorithm, the core is the raw community detection algorithm, which detects the main areas of communities. It determines the number of network partition, and affects the efficiency of the algorithm. Compared with the data in the network, the vast majority of nodes in the community are nonoverlap. The overlap nodes and nonoverlap nodes can be identified by the attribute of isLocated, which greatly reduces the repeated visit and calculation to the irrelevant nodes, and the algorithm enables to detect communities rapidly. In Figure 10, more than 80% nodes have been confirmed in the stage of raw community detection. In addition, the more obvious the community structure in the network is, the smaller the proportion of nodes need to redistribute is. Since the number of nodes in error detection and correction stage is limited and the community structure has been clear, it only needs to verify the specific nodes, which do not access large-scale adjacent nodes. So the error detection and correction algorithm has less effect on the increase with overall complexity.

6. Conclusions

In this paper, we first put forward the new features of modularity and also show the advantage of the new weighted modularity in structure, which is based on cohesive and density synthetically. The experiments are conducted on the classical networks with well-known community structure, which explore the distribution of the parameter factor. In addition, according to the proportion of overlap in the community, we present a fast overlapping community algorithm with self-correction by setting the nodes with the attributes of isVisited and isLocated, which consists of two stages: (1) the initial community detection algorithm and (2) the error detection and correction algorithm. We also propose an improved bridgeness function to evaluate the extent of overlapping nodes. The experimental results demonstrate that our algorithm is good for the expansion of discovery algorithm when extracting the overlap features. Although our algorithm is already effective, but the later

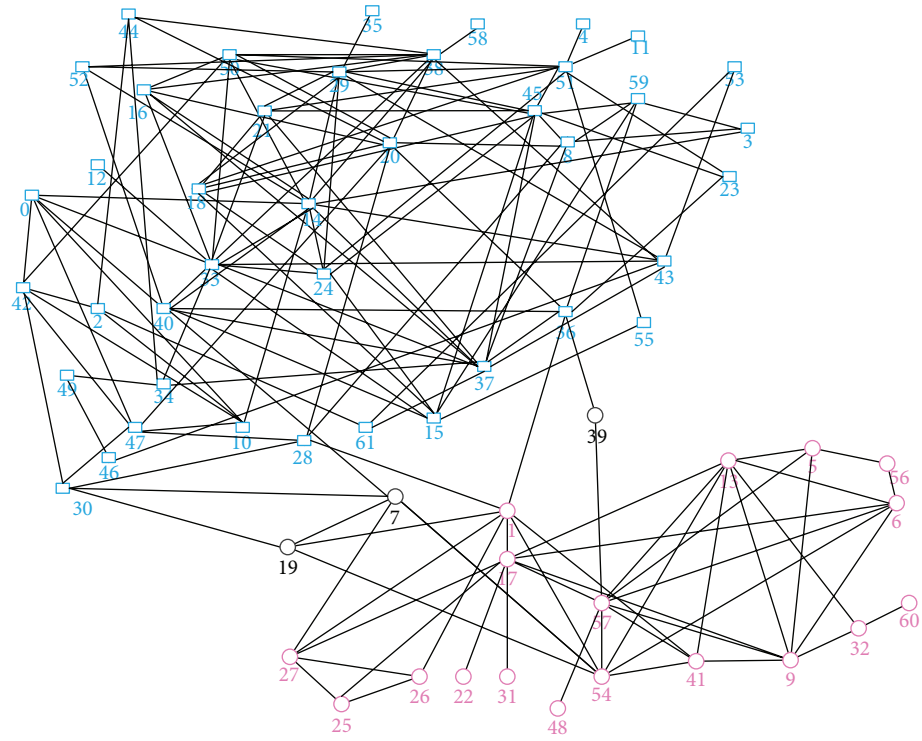


Figure 9: The community structure found by our algorithm in Dolphins interaction network.

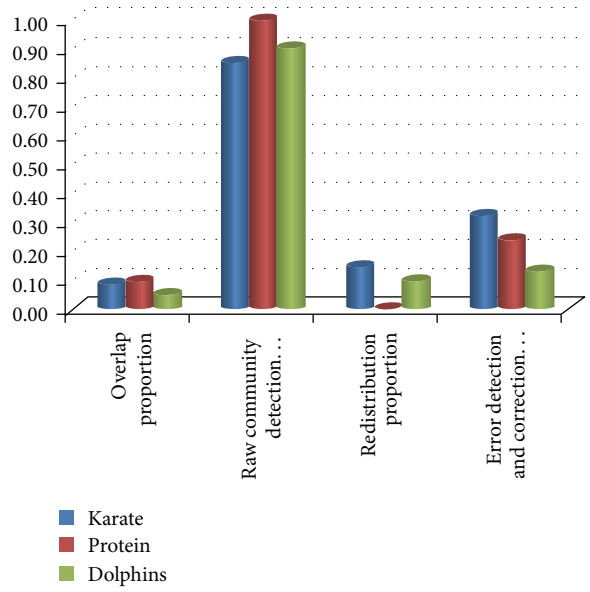


Figure 10: The proportion of the various types of nodes in networks.

work can be expanded in more different types of networks, to test out appropriate parameter and conclude parameter distribution principle. In addition, the threshold setting in the overlapping node conditions, such as in the modularity, stationarity, and the close connection, is strict in algorithm, which expands the scope of nodes in error detection and correction slightly. However, finding and extracting the new features of overlapping node are the directions in the next step. Through the experiments on the existing network, our algorithm can be applied to large-scale networks in the future.

Conflict of Interests

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

Acknowledgment

This work was supported in part by Guangdong Natural Science Foundation (Grant no. S2013040012895), the Major Fundamental Research Project in the Science and Technology Plan of Shenzhen (Grant no. JCYJ20120613104215889, Grant no. JCYJ20130329102017840, and Grant no. JCYJ20130329102032059), and Natural Science Foundation of SZU (Grant no. 00035697). The authors wish to express their appreciation to the four anonymous reviewers and the journal editor for their valuable suggestions and comments that helped to greatly improve the paper.

References

[1] 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.

[2] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, Article ID 026113, 15 pages, 2004.

[3] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.

[4] B. Yang, W. K. Cheung, and J. Liu, "Community mining from signed social networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 10, pp. 1333–1348, 2007.

[5] B. Yang, D. Y. Liu, J. Liu, D. Jin, and H. B. Ma, "Complex network clustering algorithms," *Journal of Software*, vol. 20, no. 1, pp. 54–66, 2009.

[6] M. E. J. Newman, "Detecting community structure in networks," *The European Physical Journal B*, vol. 38, no. 2, pp. 321–330, 2004.

[7] R. Guimerà and L. A. N. Amaral, "Functional cartography of complex metabolic networks," *Nature*, vol. 433, no. 7028, pp. 895–900, 2005.

[8] G. W. Flake, S. Lawrence, C. Lee Giles, and F. M. Coetzee, "Self-organization and identification of web communities," *Computer*, vol. 35, no. 3, pp. 66–71, 2002.

[9] T. Nepusz, A. Petroczi, L. Negyessy, and F. Bazso, "Fuzzy communities and the concept of bridgeness in complex networks," *Physical Review E*, vol. 77, Article ID 016107, 12 pages, 2007.

[10] S. Gregory, "Fuzzy overlapping communities in networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2011, no. 2, Article ID P02017, 2011.

[11] A. McDaid and N. Hurley, "Detecting highly overlapping communities with model-based overlapping seed expansion," in *International Conference on Advances in Social Network Analysis and Mining (ASONAM '10)*, pp. 112–119, August 2010.

[12] J. M. Kumpula, M. Kivela, K. Kaski, and J. Saramaki, "A sequential algorithm for fast clique percolation," *Physical Review E*, vol. 78, no. 2, Article ID 026109, 7 pages, 2008.

[13] H. Shen, X. Cheng, K. Cai, and M.-B. Hu, "Detect overlapping and hierarchical community structure in networks," *Physica A*, vol. 388, no. 8, pp. 1706–1712, 2009.

[14] 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. 43:1–43:35, 2013.

[15] 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.

[16] F. Havemann, M. Heinz, A. Struck, and J. Gläser, "Identification of overlapping communities and their hierarchy by locally calculating community-changing resolution levels," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2011, no. 1, Article ID P01023, 2011.

[17] A. Lázár, D. Ábel, and T. Vicsek, "Modularity measure of networks with overlapping communities," *Europhysics Letters*, vol. 90, no. 1, Article ID 18001, 2010.

[18] L. Qin, N. Lu, and Y. Jin, "Process of modularity in networks: the art of evaluation," in *Proceedings of the 5th IEEE International Conference on Advanced Computational Intelligence (ICACI '12)*, pp. 1–6, 2012.

[19] L. Wang, G.-Z. Dai, and H.-C. Zhao, "Research on modularity for evaluating community structure," *Computer Engineering*, vol. 36, no. 14, pp. 227–229, 2010.

[20] 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.

[21] A. Clauset, "Finding local community structure in networks," *Physical Review E*, vol. 72, no. 2, Article ID 026132, 2005.

[22] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri, "Extending the definition of modularity to directed graphs with overlapping communities," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2009, no. 3, Article ID P03024, 2009.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 581052, 9 pages
http://dx.doi.org/10.1155/2014/581052

Research Article

WDM Network and Multicasting Protocol Strategies

Pinar Kirci[1] and Abdul Halim Zaim[2]

[1] *Computer Engineering Department, Istanbul University, 34010 Istanbul, Turkey*
[2] *Computer Engineering Department, Istanbul Commerce University, 34840 Istanbul, Turkey*

Correspondence should be addressed to Pinar Kirci; pkirci@istanbul.edu.tr

Received 23 December 2013; Accepted 17 February 2014; Published 12 March 2014

Academic Editors: X. Yang, Z. Yu, and W. Zhang

Optical technology gains extensive attention and ever increasing improvement because of the huge amount of network traffic caused by the growing number of internet users and their rising demands. However, with wavelength division multiplexing (WDM), it is easier to take the advantage of optical networks and optical burst switching (OBS) and to construct WDM networks with low delay rates and better data transparency these technologies are the best choices. Furthermore, multicasting in WDM is an urgent solution for bandwidth-intensive applications. In the paper, a new multicasting protocol with OBS is proposed. The protocol depends on a leaf initiated structure. The network is composed of source, ingress switches, intermediate switches, edge switches, and client nodes. The performance of the protocol is examined with Just Enough Time (JET) and Just In Time (JIT) reservation protocols. Also, the paper involves most of the recent advances about WDM multicasting in optical networks. WDM multicasting in optical networks is given as three common subtitles: Broadcast and-select networks, wavelength-routed networks, and OBS networks. Also, in the paper, multicast routing protocols are briefly summarized and optical burst switched WDM networks are investigated with the proposed multicast schemes.

1. Introduction

In our century, internet is an indispensable technology and the main reason of the increasing internet traffic is the ever growing demands of the users. Thus, new technologies are needed for high-bandwidth intensive applications to provide enough bandwidth. WDM optical network is proposed to deal with this problem. Optical network occupies local to wide area, connects millions of users, and offers high data rates and capacities exceeding those of familiar networks. WDM technology well utilizes the high-bandwidth characteristic of the fiber-optic links. In the concept of WDM over fiber-optic link, the laser beams travel over a single fiber where each laser beam travels over a different optical wavelength [1]. Hence, on these days, WDM networks become the most preferred architecture for backbone networks with the many advantages they own. The first one is the ability of building robust multicast trees with the optical layer topology instead of electronic layer. The other one is the ability of light-splitting which is more efficient than copying IP packets. The last one is the bit-rate transparency in optical multicast. Furthermore, multicast services' needed applications are also gaining great concern. Multicasting is a way of sending information from a single or multiple sources to many destinations. Recently, multicasting in optical domain gains much more attention than multicasting in electronic domain because of the light-splitting capability of optical switches and because these optical switches present excellent solutions instead of duplicating data in electronic domain [2–4].

The paper is organised as follows. In Section 2, WDM networks' general structure is presented. Optical WDM networks and main topics about multicasting are briefly summarized in Section 3. In the last section, a new multicasting protocol for OBS is proposed.

2. General Properties of WDM Networks

As the bandwidth intensive application usage increases, the bandwidth demand increases too. Therefore, WDM is presented to meet this increasing need for very-high-bandwidth transport networks. With the WDM technology, it

is easy to build very large wide-area networks. Thus, internet, video-on-demand, distributed interactive simulation, graphics and visualization, worldwide web browsing, e-commerce, medical image access and distribution, shared whiteboards, teleconferencing, and many more bandwidth intensive applications demand multicast services. On the other hand, these multicast applications belong to the category of multipoint applications.

The first service is *one-to-many* which includes most of the applications such as on-demand video distribution, network news distribution, file distribution, and document distribution. The second service, *many to one,* includes resource discovery, data collection at a central location, auctions, group polling, and accounting. The last service *many to many* applications are multimedia conferencing, distance learning, and distributed simulations. The transmission rates of these applications are on the order of subwavelength rates. This means that some of the applications do not need the whole transmission rates of a lightpath. Therefore, for this kind of applications, traffic grooming is required. In the traffic grooming technique, different traffic streams are switched into higher-speed streams. There are many traffic grooming models in the literature but most of them are designed for unicast traffic. It is obvious that the network design and traffic structures of unicast traffic and a multicast traffic are different from each other. For example, in order to support a multicast traffic, the packets need to be duplicated but in a unicast traffic there is no need for packet duplication. In unicast connections, a path should be routed from source to destination. Similarly, in multicast connections, the optical signal should be routed from the source of a tree and span all destinations. Also, wavelength assignment should be done for these connections. The problem of assigning wavelengths and routes for a group of connections is known as routing and wavelength assignment (RWA) [1, 3, 4]. Multicast routing and wavelength assignment (MC-RWA) problem is studied according to many aspects. Some of these aspects are proposed in [5] with the RWA problem investigation.

The main structure of an optical WDM is composed of many routing nodes that support a definite number of wavelengths. Also, these nodes are connected to each other by point-to-point fiber links. In a WDM network, a connection is named as light path. In general, there is a packet transmission from one source to multiple destinations in a multicast network. For this type of transmissions there should be a connection established from one source node to many destination nodes.

For an efficient multicasting, it is better for the nodes to have the capability of splitting and/or broadcasting from input to output. In some situations, an incoming channel is needed to be connected to a group of outgoing channels each of which is on a different fiber. There are three multicast models. In *multicast with same wavelength* (MSW) model, the same wavelength is assigned to the source and the whole destinations in the multicast connection; that is, there is no wavelength conversion taking place in the transmission. In the *multicast with same destination wavelength* (MSDW) model, the same wavelength is assigned to the whole output signals but this wavelength may be different from the input wavelength. The last model is *multicast with any wavelength* (MAW). In this model, the source and destinations use different wavelengths [6].

These types of network services require optical cross-connects (OXC) to be able to switch data carrying connections. In general, opaque (optical-electronic-optical) and transparent (all-optical) switching architectures are studied with the fractional capacity sessions and sparse splitting constraints [7].

Clearly, optical switches are essential for WDM networks. The switches may have much different architecture and two of them are shown in Figure 1. They have two important facilities; these are routing of optical paths and termination or addition of optical paths. Light-splitting capability is so important for optical switches in bandwidth intensive applications. In all-optical networks, power is such an important measurement. For this reason, in optical multicasting, power splitters need to be used. These devices are distributing the input signal to all outputs so there is no need for buffering in multicast connections [8].

The basic architectures of WDM networks in literature are summarized with three sub titles. The general structure of a broadcast-and-select WDM network consists of N nodes, a passive star coupler (PSC), and optical fibers. The nodes and passive star couplers are connected to each other with a pair of optical fibers. Also, the nodes have a number of transmitters and receivers. The transceivers may be fixed to a wavelength or not. If they are not fixed, then they are tunable over a number of wavelengths. In literature, there are four types of node transceivers explained. These are fixed transmitter and fixed receiver (FT-FR), fixed transmitter and tunable receiver (FT-TR), tunable transmitter and fixed receiver (TT-FR), and tunable transmitter and tunable receiver (TT-TR).

Broadcast-and-select in WDM networks can be examined in two categories. These are single hop and multihop. In a single hop network, in order to support a packet transmission, the source node's transmitter and the destination node's receiver should be tuned to the same wavelength. When a source node sends a data, it passes through the PSC, but not an intermediate node, and reaches the destination node. In general, broadcast-and-select networks utilize PSCs. Here, every node is connected to a passive star with two-way fibers. In this system, the data is transmitted from N different nodes to passive star via N wavelengths. Furthermore, the nodes have only one receiver which is tuned to only one of the N channels and also these nodes may listen to only one of the N information streams. In other words, a PSC is used for dividing the incoming light from any port equally to all other ports. Hence, the arriving packet is broadcasted to all of the other ports by a PSC. When a node receives a packet, it checks if this packet belongs to itself or not. Thus, single hop WDM network is a kind of shared-channel network. Here, the system utilizes wavelength-agile transmitters or receivers, which can tune quickly to different wavelengths but, as mentioned before, this function costs higher than multihop communication. Compared with the multihop networks, single hop networks need expensive wavelength tunable transceivers. The observations imply that these kinds of networks may have contentions during the

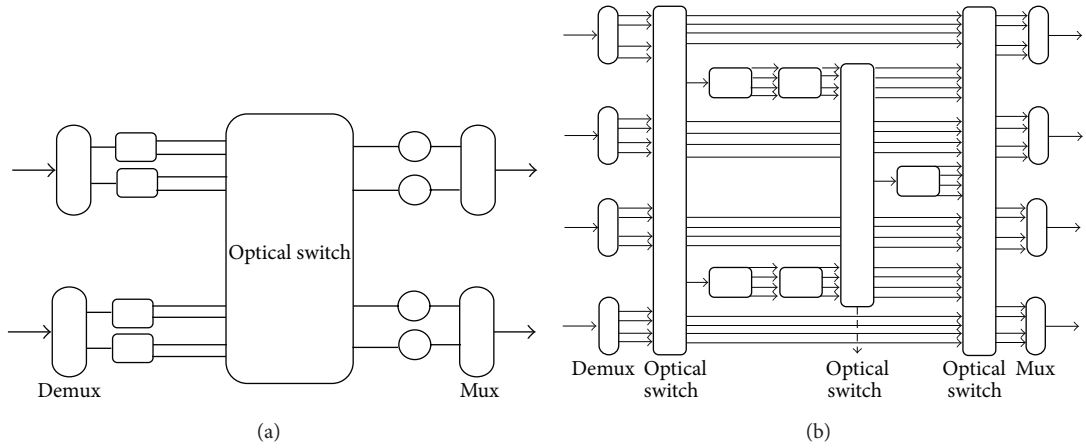


Figure 1: (a) 2×2 switch architecture and (b) 4×4 switch architecture.

packet transmissions. Collision is a type of contention, it occurs when two or more transmitters try to transmit to the same destination node with tunable receivers over different channels simultaneously. To avoid this type of contentions, the most important fact is the coordination of transmissions which is adjusted by a media access protocol or a multicast scheduling algorithm (MSA).

There are three types of well-known multicast scheduling algorithms. The first one is reservation-based MSA. Here, a shared control channel is utilized for sending nodes' transmission requests. The arriving packets are placed in an arrival queue and a control packet is sent over a control channel to all of the nodes. The control channel has a slotted structure and the access of the nodes are arranged by round-robin time division multiple access (TDMA). In fact, the control channel is divided into control frames and each of these frames is also subdivided into N control slots. If node i wants to transmit its control packet, it can only transmit the packet in the ith slot of each control frame. When the transmission issue is over, then the data packet is sent to the nodes' waiting queue, till it is transmitted to all of its own destination nodes. By the way, the control packet will arrive to the nodes in the network after a propagation delay. These nodes will run the same MSA for reserving data channels and time slots and scheduling the transmission of the corresponding packet. Then each node updates its record after the reservation and gets ready for a new scheduling. The second one is random-access-based MSA; Here in order to support the system, a centralized scheduler is used, and the nodes send their requests to the control channel with an unslotted random access protocol but the central scheduling works with a slotted manner. The transmission requests arrive to the centralized scheduler via a control wavelength. After receiving the requests, the centralized scheduler runs the MSA and informs the nodes via a separated control wavelength. In this system, there is a request query for each node which is checked consistently and scheduled to the best slot. This type of MSA is presented to reduce the complexity of reservation-based MSA because the nodes try to access the data channels in a random manner. The third one is preallocation-based MSA which has a static structure. In this system, the transmissions are arranged by predetermined schedules, such as, for each slot, the active transmitters, the corresponding wavelengths, the active receivers are predetermined. Thus, the overhead of the control messages can be decreased.

Besides, multihop WDM network works with cheaper fixed-wavelength transceivers. In other words, it functions with fixed-tuned transmitters and receivers. Furthermore, the network owns a fixed logical topology because of the fixed connectivity of nodes. The main reason of this fact is the FT-FR node structure of the network. The multihop networks' logical topology may be irregular or regular. The irregular topology design has a high routing complexity. Consequently, it may not have gained too much attention. But many of the regular topologies are well studied such as shuffle-net, Manhattan Street (MSN), and Hypercube. Furthermore, the main problem of MC-RWA is finding a multicast routing tree and then allocating necessary wavelengths on every link of the tree. To deal with this problem, it is examined as two subtitles: routing and wavelength assignment. In other words, when a multicast request is received, the first thing is finding a tree-structured route which is rooted at the source and spans all of the destinations. To find such a multicast tree, there are many schemes presented in the literature but the simplest way is utilizing the IP multicast routing protocols at the IP layer. After constructing a tree, the next step is allocating available wavelengths for each link of the tree. Routing and wavelength assignment problems are the main problems of wavelength-routed networks. In the wavelength-routed system, the access nodes have transmitters and receivers. In addition to this, the incoming light is routed to the intended outgoing link by the wavelength routing switches. These switches should be multicast capable of providing multicasting in wavelength-routed networks. In other words, these switches should be able to replicate and forward each incoming optical signal to all outgoing links all optically. In order to support multicasting, it is important for a wavelength-routed switch to have a light-splitting capability. In general, this is achieved by an optical splitter. If a wavelength-routed switch does not have a light-splitting capability, then they are called sparse

splitting capable switches. Multicast routing and wavelength assignment (MC-RWA) are basically divided into two categories: *single multicast request* case and *multiple multicast request case*. The first case is also categorized as *multicast tree optimization* and *light-forest construction in the case of sparse splitting*. In fact, the main aim of *single multicast request case* is to find routes from source to all other destinations. In *multicast tree optimization*, all of the nodes in the network are multicast capable; so there are many multicast trees to choose from. In this situation, the best one is chosen but, in *light-forest construction in the case of sparse splitting*, the network has sparse splitting nodes so alternative routes must be found to support multicasting. In order to support MC-RWA, new algorithms are presented for static and dynamic structures. In static structure, the multicast trees are predetermined and cannot be changed till the end of the wavelength assignment, in other words a multicast group's membership does not change during its life time. Also, in the dynamic structure, new multicast trees can be constructed for unserved groups; this means that the nodes may join/leave whenever they want. *Multiple multicast request case* structure is composed of dynamic traffic and static traffic. In dynamic traffic MC-RWA, many multicast requests arrive and leave nodes dynamically. For each arriving request, a route and wavelength should be allocated to decrease the blocking rates. In static traffic MC-RWA, multiple multicast requests arrive statistically and they may be supported simultaneously at a meantime or a long term multicast traffic matrix [7–14].

IP multicast routing protocols are examined in two subtitles which are dense/sparse mode and source-based/core-based modes. The first one depends on the expected distribution of multicast group members throughout the network. The second one depends on the construction method of multicast tree and the root of the multicast tree constructed. The Distance Vector Multicast Routing (DVMRP), the Protocol-Independent Multicasting Dense Mode (PIM-DM), and Multicast Extensions to OSPF (MOSPF) multicast routing protocols are dense mode and also use source-based tree but the well-known multicast routing protocols are the Protocol-Independent Multicasting Sparse Mode (PIM-SM) and the core-based tree (CBT) which are both sparse mode and use shared trees rooted at a core router or rendezvous point (RV) which is a meeting point between sources and destinations of a definite group. Furthermore, PIM-SM may also use source-based tree. DVMRP is the first and most famous routing protocol which is used to support IP multicast on MBone. DVMRP and PIM-DM protocols benefit from a flood and prune algorithm, to form a shorter path tree between the sources and receivers and to discover the exact positions of the receivers.

In DVMRP, a neighbour probe message is sent to the whole network on definite time intervals. The messages include neighbour routers' list that sent their own neighbour probe messages before. So the routers discover their neighbours in the network. DVMRP utilizes a reverse shortest path tree and owns its own integrated unicast counting protocol. Also, tunneling is used to span the nodes which are not multicast capable. In other words, packets are broadcasted to the destinations by Reverse Path Forwarding (RPF). If there is not a downstream member then branches are pruned. Also, if new members on pruned branches want to join the multicast group, then they should send explicit GRAFT messages upstream.

In MOSPF, routers change link state information with each other so they gain the exact and last information about actual state of the network. Hence, periodically multicast group membership is flooded to the routers. As soon as a packet arrives at a router, a shortest path tree is computed and the results are stored for the forwarding of next multicast traffic. MOSPF protocol forwarding model and routing mechanisms are summarized in [15, 16].

The next routing protocol is Protocol-Independent Multicasting (PIM). PIM defines the networks as PIM domains and non-PIM enabled domains. Inside the PIM domains, the routers utilize PIM for multicasting and there are bootstrap routers that distribute the information about rendezvous points. Moreover, on the boundaries of these PIM domains the multicast boundary routers are placed to provide the interaction with the non-PIM enabled domains. PIM includes two different protocols; these are PIM-sparse and PIM-dense.

In PIM-sparse mode, the places of the members are supposed to be a long way off each other and the convenient bandwidth tends to be small. Also, the convenient members should belong to a subnetwork. This routing protocol includes tree structure with rendezvous points. Therefore, it is more suitable for groups which are geographically distributed in the network. Here, to join a group for group membership, explicit join operations are needed. By the way, the data is transmitted to only rendezvous point not to the entire network. PIM-sparse mode structure is suitable for construction of multicast trees with rendezvous points. Rendezvous points are utilized for group awareness and explicit join functions. Additionally, the concept of PIM-dense mode is slightly different from PIM-sparse mode. First of all, the members in the PIM-dense mode are closer to each other and the convenient members might be in every subnetwork. Also, the density of group members is high. Furthermore, it has minimal complexity which is a great advantage for a protocol. According to the PIM-dense mode, the group members of all subnetworks want to receive the data at startup. Thus, flooding and pruning are used. Immediate integrations of new members to the multicast tree are performed by sending graft data units. There are specific differences between PIM-sparse and PIM-dense modes. In PIM-dense mode, periodically transmitted join data units and rendezvous points are not used. Also, PIM-dense mode is preferred to be used in domains but PIM-sparse mode is better for larger networks.

The core-based tree (CBT) includes shared bidirectional multicast trees with rendezvous points. When these trees are constructed, they include current group membership. The concept of CBT structure is to reduce the status information amount and the overhead of control data units. To deal with the status information amount, shared multicast trees are used [17, 18]. Here, the decision about the location of the rendezvous point in the network is important. In literature, many methods are presented for deciding the optimum

location. The method in [19] depends on distance vector information to obtain the optimum RP location with the lowest total cost.

3. Optical WDM Networks

The main function of WDM structure may be explained as follows. It offers the needed capacity and the connections of the optical devices which are important for performing the transmission in the optical level. At the beginning, WDM technology is presented for increasing the capacity of point-to-point fiber links in opaque networks. For multiple accesses, before the WDM technology, time division multiplexing (TDM) and some hybrid architectures [20] were used. But the great increase about the employment of OXCs in networks inspired the transparent optical network designs. In transparent optical networks, there is no need for signal regeneration or OEO conversions. The main role of an OXC in an optical network is to switch an input fiber's arriving signal to the same wavelength on an output fiber. In literature, the IP traffic in optical WDM networks is transmitted by three main switching technologies. These are optical circuit switching (OCS), optical packet switching (OPS), and optical burst switching (OBS).

In OCS, before starting a transmission, a station-to-station lightpath establishment should be arranged as in telephony technology. After the establishment, the data is transmitted over the lightpath and then the connection is closed so the OCS structure may be defined as three phases: establishment, data transfer, and termination. There is no data loss in OCS but if the traffic load increases, then the packets should wait at the incoming or outgoing queues. OCS switching is also used in high performance computing systems to perform cheaper and power-efficient interconnections but, in OCS, reserving a channel for a connection causes great inefficiency [21–23].

OPS is proposed as a new option for wavelength routing. An OPS node includes an optical switch fabric. The switch fabric is used for the reconfiguration according to the packet headers. The packet headers are either transmitted in-band with the packet or out of band on a separate control channel and processed electronically. It is obvious that processing a header and reconfiguration of a switch takes some time; by the way, the packet needs to be delayed by optical delay lines. Thus, fast switching times are important for optical packet switching. Most of the studies about OPS are about throughput improvement. A study about the throughput analysis with big optical packet switches is presented in [24]. Most of the studies in literature are based on flow control algorithms or the packet size and window size explorations for the improvement of the network throughput. The most important concepts in OPS are routing, traffic control, and error control. In OPS, the packets should be routed from one node to the other one because the source and the destination are not connected directly; also for gaining efficient network performance, the whole network traffic is regulated. Besides, in OPS, the error control is an important feature for preventing the packet loss at the stations.

OBS includes some of the main characteristics of wavelength routing and OPS. In both wavelength routing and OBS, the data is not processed or buffered at the intermediate nodes. Also, in both of the OPS and OBS, the data and the header are processed separately. In OBS, the burst is composed of two parts which are control data and user data. The information carried by the control data are routing information, address information, wavelength channel information and the offset time information. An OBS network is composed of edge routers, core routers, and fiber links. The nodes are connected to each other by fiber links. Moreover, by the help of the WDM technology, these fiber links carry multiple wavelength channels. In ingress edge nodes, first of all, the arriving IP packets are assembled into a burst (called burstification) according to the destination multicast group address and QoS requirements and then scheduled for transmission on outgoing wavelengths. In the core nodes, the bursts are switched from input ports to output ports according to the information in the headers. Furthermore, these core nodes handle burst contentions [25–27]. In an OBS network, multiple IP networks construct bursts and these bursts are switched all-optically. To configure the switches along the bursts' route, a control packet is sent ahead of the burst as presented in Figure 2. During a definite offset time, the control packet is processed. According to the information in the control packet, the switches of intermediate nodes are configured for the coming bursts. In addition to this, the assembled bursts are transmitted over OBS core routers all-optically. First, the bursts arrive at the egress edge node; then the egress edge node disassembles the bursts into packets and then forwards them to the destinations [28–35].

4. The Proposed Structure for Optical Burst Switched WDM Networks

In the paper, an optical WDM network is constituted. The network model is composed of multicast capable source, ingress switch, intermediate switches, edge switches, and clients. The nodes of the network are connected to each other by fiber links that carry different numbers of wavelengths.

The beginning of a session depends on the determined video programs. If a source has a video program to send, it will broadcast the video context to the clients. If the clients want to view the program, they will request the program with sending a join-request message. Consequently, the source multicasts the video bursts to the considered clients. The optimum paths between clients and source are calculated by intermediate switches. Also, the join-request traffic monitoring and controlling is managed by intermediate switches. The request-refused message is sent by the intermediate switch to the related edge switch after receiving a join-request message from the edge switch, to inform the edge switch that it will not be convenient for the video transmission. After receiving the request-refused message, edge switch sends a rejoin-request message to another close intermediate switch for the same video transmission. Also, keepalive messages are sent by clients during the long data transmission periods to inform that the multicasting session connection is still

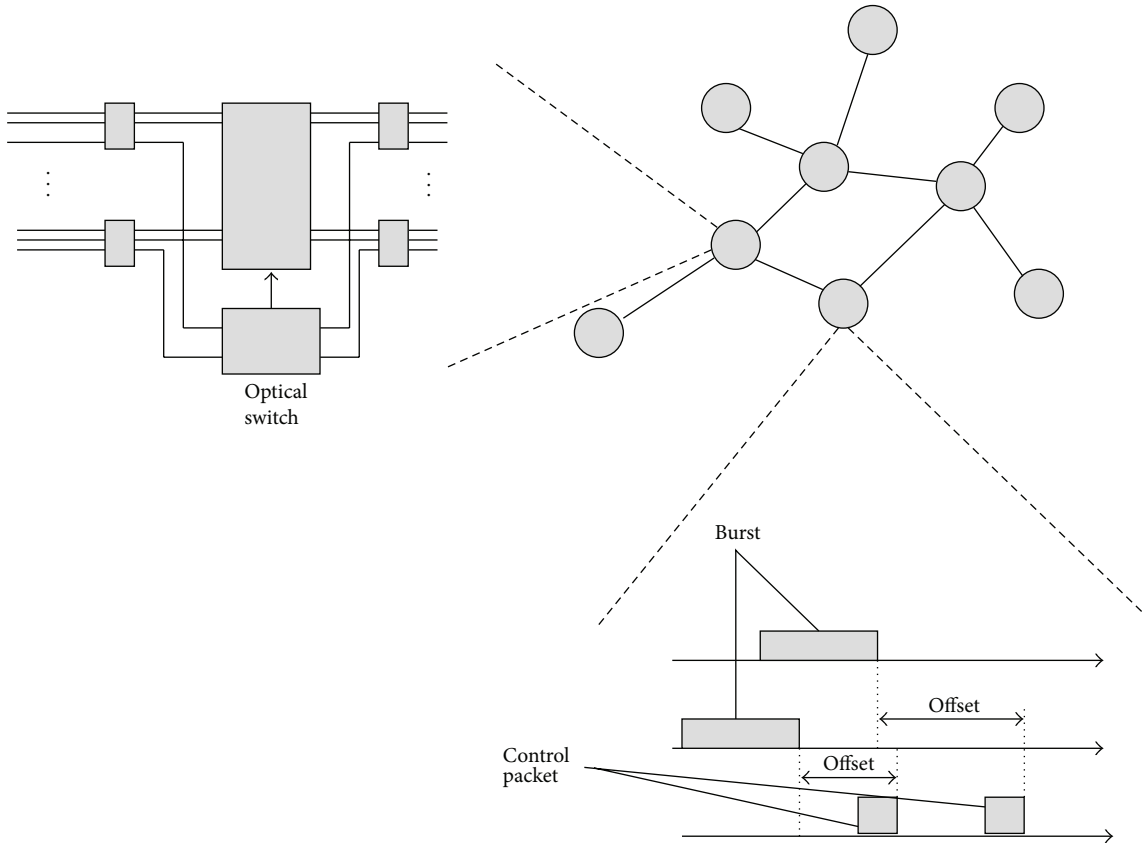


Figure 2: An optical burst switched network.

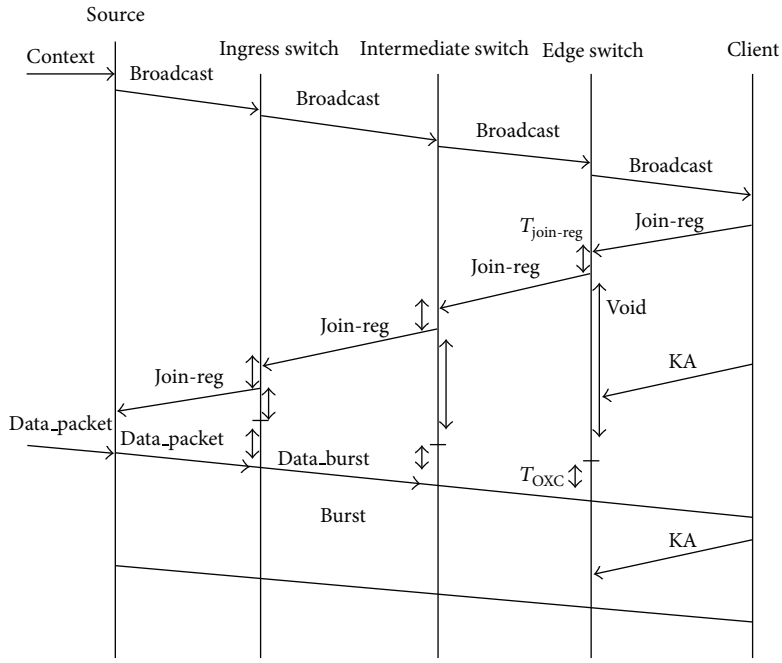


Figure 3: The message flows in the protocol.

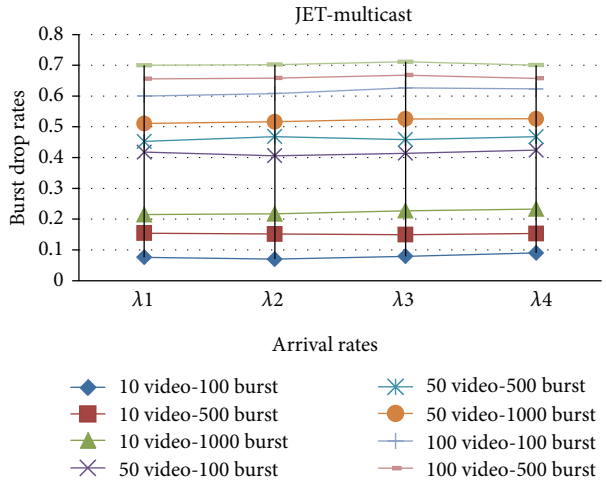


Figure 4: The effect of λ values on the burst drop rates with JET.

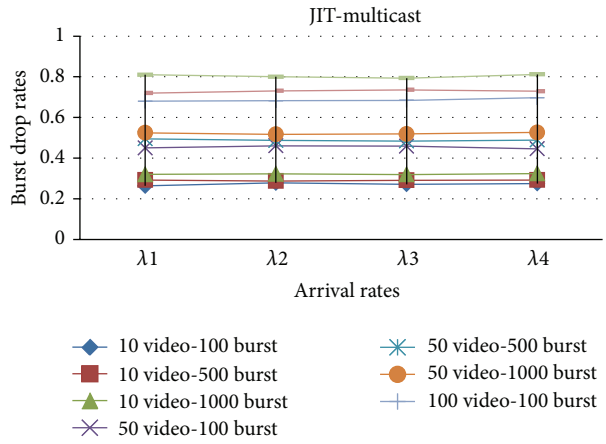


Figure 5: The effect of λ values on the burst drop rates with JIT.

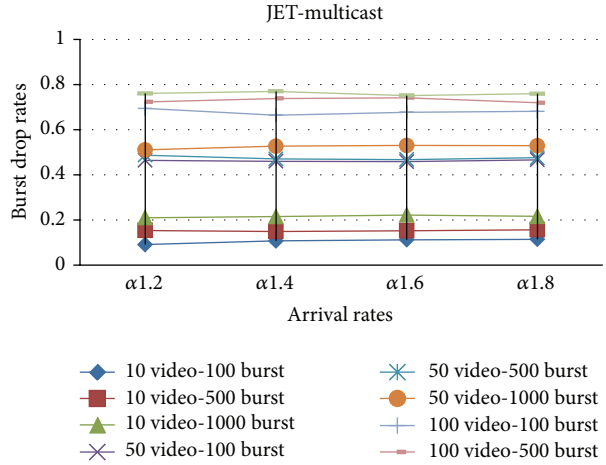


Figure 6: The effect of α values on the burst drop rates with JET.

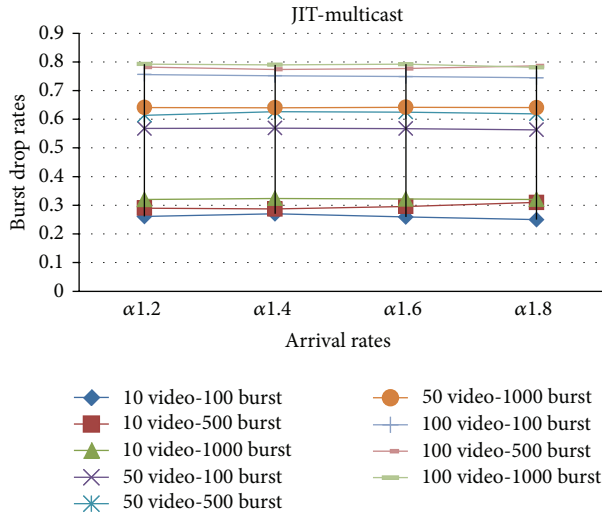


Figure 7: The effect of α values on the burst drop rates with JIT.

continuing. Figure 3 demonstrates these basic message flows of the protocol. On the proposed NSFNET topology, all of the switches assumed to have full-splitting capability.

In Figures 4 and 5, the simulation is constructed over a network that transmits both multicast and unicast traffic with (Just Enough Time) JET and (Just In Time) JIT reservation protocols on Poisson arrival rates.

In Figures 6 and 7, both multicast and unicast traffics are sent over the network with (Just Enough Time) JET and (Just In Time) JIT reservation protocols on self-similar arrival rates.

According to the simulation results, the proposed protocol structure with JET reservation protocol produces less burst drop rates and better performance. By the way, the effect of different arrival rates is also examined in the simulations.

5. Conclusion

Today, the network technologies are in a revolution period. The development of these technologies shapes according to the users' demands and QoS parameters. This paper includes the main and mostly studied topics about WDM networks in literature. In this paper, multicast communications are investigated and core topics are summarized under subtitles. General structure of WDM multicasting is presented with the basic properties of optical switches at the beginning. Broadcast-and-select and wavelength-routed WDM networks are also examined. By the way, IP multicast routing protocols are presented because of their convenient and suitable structures for WDM multicasting. In the following, OBS WDM network structure is given with the other related switching technologies such as OCS and OPS. As we mentioned before, fast optical switches are needed in OPS but, in OBS, the data is transmitted in large bursts; so there is no need for fast optical switches. Also, during the burstification process an OBS core router's switching granularity will be coarser than OPS. Thus, the control and processing overhead decrease. Similarly, OBS provides coarse switching granularity and statistical multiplexing. In other words, OBS combines the advantages of OCS and OPS. As we mentioned

before, WDM is a developing technology. Hence, it includes many open research problems. In this paper, we present a general description of WDM networks and multicasting, because we work on a new multicasting protocol on WDM OBS networks and analyze the new structure with different traffic scenarios. Here, a new protocol structure is proposed over OBS network together with performing multicast and considering QoS for real time applications and the structure is supported by simulations. A new multicasting protocol structure is introduced for OBS networks together with the state diagrams and state transmissions of the protocol from source to clients. Over the basic topology, a switch is designed as the traffic source and the clients are the traffic receivers. The burst drop rates are investigated according to their arrivals to the network. Our protocol performance is examined on JET and JIT with multicasting. Our performance results show that the presented protocol structure with JET provides better drop rates than JIT.

Conflict of Interests

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

References

[1] Y. Wang and Y. Yang, "Multicasting in a class of multicast-capable WDM networks," *Journal of Lightwave Technology*, vol. 20, no. 3, pp. 350–359, 2002.

[2] L. Gargano, A. A. Rescigno, and U. Vaccaro, "Multicasting to groups in optical networks and related combinatorial optimization problems," in *Proceedings of the Parallel and Distributed Processing Symposium*, 2003.

[3] A. E. Kamal, "Algorithms for multicast traffic grooming in WDM mesh networks," *IEEE Communications Magazine*, vol. 44, no. 11, pp. 96–105, 2006.

[4] X. Zhang, J. Wei, and C. Qiao, "fundamental issues in ip over wdm multicast," in *Proceedings of the 8th International Conference on Computer Communications and Networks*, pp. 84–90, 1999.

[5] N. K. Singhal, L. H. Sahasrabuddhe, and B. Mukherjee, "Optimal multicasting of multiple light-trees of different bandwidth granularities in a WDM mesh network with sparse splitting capabilities," *IEEE/ACM Transactions on Networking*, vol. 14, no. 5, pp. 1104–1117, 2006.

[6] S. Sankaranarayanan and S. Subramaniam, "Comprehensive performance modeling and analysis of multicasting in optical networks," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 9, pp. 1399–1413, 2003.

[7] G.-S. Poo and Y. Zhou, "A new multicast wavelength assignment algorithm in wavelength-routed WDM networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 4, pp. 2–12, 2006.

[8] M. Ali and J. S. Deogun, "Cost-effective implementation of multicasting in wavelength-routed networks," *Journal of Lightwave Technology*, vol. 18, no. 12, pp. 1628–1638, 2000.

[9] J. He, S.-H. Gary Chan, and D. H. K. Tsang, "Routing and wavelength assignment for WDM multicast networks," in *Proceedings of the IEEE Global Telecommunicatins Conference (GLOBECOM '01)*, vol. 3, pp. 1536–1540, November 2001.

[10] H.-T. Wu, K.-W. Ke, and S. Huang, "A novel multicast mechanism for optical local area networks," *Computers and Electrical Engineering*, vol. 33, no. 2, pp. 94–108, 2007.

[11] J. He, S. H. G. Chan, and D. H. K. Tsang, "Multicasting in wdm networks," *IEEE Communications Surveys & Tutorials*, vol. 4, pp. 2–20, 2002.

[12] S. B. Tridandapani and B. Mukherjee, "Channel sharing in multi-hop WDM lightwave networks: realization and performance of multicast traffic," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 3, pp. 488–500, 1997.

[13] M. A. Marsan, A. Bianco, E. Leonardi, F. Neri, and A. Nucci, "Multi-hop packet scheduling in WDM/TDM broadcast-and-select optical networks with arbitrary transceivers tuning latencies," in *Proceedings of the Global Telecommunications Conference (GLOBECOM '98)*, vol. 2, pp. 1105–1111, November 1998.

[14] N. Rammohan and C. Siva Ram Murthy, "On-line multicast routing with QoS constraints in WDM networks with no wavelength converters," *Computer Networks*, vol. 50, no. 18, pp. 3666–3685, 2006.

[15] P. G. Ranjitkar, I. M. Suliman, P. Geil, M. M. Kuipers, and R. Prasad, "IP multicast implementation based on the Multicast Extensions to OSPF protocol," in *Proceedings of the IEEE International Conference on Personal Wireless Communications*, pp. 484–489, December 2000.

[16] F. Font and D. Mlynek, "Applying clustering algorithms as core selection methods for multiple core trees," in *Proceedings of the IEEE International Conference on Communications*, pp. 2030–2035, June 2004.

[17] S. Ueno, T. Kato, and K. Suzuki, "Analysis of Internet multicast traffic performance considering multicast routing protocol," in *Proceedings of the International Conference on Network Protocols*, pp. 95–104, November 2000.

[18] R. Wittman and M. Zitterbart, *Multicast Communication Protocols and Applications*, Morgan Kaufmann, San Francisco, Calif, USA, 2001.

[19] H.-C. Lin and S.-C. Lai, "Simple and effective core placement method for the core based tree multicast routing architecture," *Proceedings of the IEEE International Performance, Computing and Communications Conference*, pp. 215–219, 2000.

[20] T. Cevik, "A hybrid OFDM-TDM architecture with decentralized dynamic bandwidth allocation for PONs," *The Scientific World Journal*, vol. 2013, Article ID 561984, 9 pages, 2013.

[21] M. Jeong, H. C. Cankaya, and C. Qiao, "On a new multicasting approach in optical burst switched networks," *IEEE Communications Magazine*, vol. 40, no. 11, pp. 96–103, 2002.

[22] J. A. Hernández, J. Aracil, V. López, J. F. Palacios, and Ó. G. De Dios, "A resilience-based comparative study between optical burst switching and optical circuit switching technologies," in *Proceedings of the International Conference on Transparent Optical Networks (ICTON '06)*, vol. 3, pp. 231–234, June 2006.

[23] K. J. Baker, A. Benner, R. Hoare et al., "On the feasibility of optical circuit switching for high performance computing systems," in *Proceedings of the ACM/IEEE Supercomputing Conference*, 2005.

[24] P. Zhou and O. Yang, "How practical is optical packet switching in core networks?" in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '03)*, pp. 2709–2713, December 2003.

[25] B. Kantarci and P. Sarisaray, "A survey of time-domain optical burst switched wdm/tdm networks," in *Proceedings of the 3rd Asia Pacific Interational Sysmposium on Information Technology*, pp. 149–156.

[26] J. P. Jue and V. M. Vokkarane, *Optical Burst Switched Networks*, Springer, 2005.

[27] Z. Miao, H. Yamamoto, and T. Takahashi, "Throughput analysis of adaptive flow control protocol for optical packet switching networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '05)*, vol. 4, pp. 1984–1988, December 2005.

[28] M. Jeong, C. Qiao, Y. Xiong, and M. Vandenhoute, "Bandwidth-efficient dynamic tree-shared multicast in optical burst-switched networks," in *Proceedings of the International Conference on Communications (ICC '01)*, vol. 2, pp. 630–636, June 2001.

[29] M. Jeong, C. Qiao, Y. Xiong, H. C. Cankaya, and M. Vandenhoute, "Tree-shared multicast in optical burst-switched WDM networks," *IEEE Journal of Lightwave Technology*, vol. 21, no. 1, pp. 13–24, 2003.

[30] M. Jeong, Y. Xiong, H. C. Cankaya, M. Vandenhoute, and C. Qiao, "Efficient multicast schemes for optical burst-switched WDM networks," in *Proceedings of the IEEE International Conference on Communications*, vol. 3, pp. 1289–1294, June 2000.

[31] M.-T. Chen, S.-S. Tseng, and B. M. T. Lin, "Dynamic multicast routing under delay constraints in WDM networks with heterogeneous light splitting capabilities," *Computer Communications*, vol. 29, no. 9, pp. 1492–1503, 2006.

[32] G. N. Rouskas, "Optical layer multicast: rationale, building blocks, and challenges," *IEEE Network*, vol. 17, no. 1, pp. 60–65, 2003.

[33] W. Stallings, *Data and Computer Communications,*, Macmillan, NewYork, NY, USA, 1991.

[34] P. Kırcı and A. H. Zaim, "The JIT, JET and horizon signalling protocols on optical burst switches," *Optica Applicata*, vol. 36, no. 1, 2006.

[35] P. Kırcı and A. H. Zaim, "Performance analysis of signaling protocols on OBS switches," in *Performance, Quality of Service, and Control of Next-Generation Communication and Sensor Networks III*, vol. 6011 of *Proceedings of the SPIE*, pp. 129–137, Boston, Mass, USA, October 2005.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 341038, 7 pages
http://dx.doi.org/10.1155/2014/341038

Research Article

Motion Adaptive Vertical Handoff in Cellular/WLAN Heterogeneous Wireless Network

Limin Li, Lin Ma, Yubin Xu, and Yunhai Fu

CRC, School of Electronics and Information Engineering, Harbin Institute of Technology, Nan Gang District, Harbin 150001, China

Correspondence should be addressed to Limin Li; lilimin@hit.edu.cn

Received 27 December 2013; Accepted 3 February 2014; Published 11 March 2014

Academic Editors: H. R. Karimi, X. Yang, Z. Yu, and W. Zhang

In heterogeneous wireless network, vertical handoff plays an important role for guaranteeing quality of service and overall performance of network. Conventional vertical handoff trigger schemes are mostly developed from horizontal handoff in homogeneous cellular network. Basically, they can be summarized as hysteresis-based and dwelling-timer-based algorithms, which are reliable on avoiding unnecessary handoff caused by the terminals dwelling at the edge of WLAN coverage. However, the coverage of WLAN is much smaller compared with cellular network, while the motion types of terminals can be various in a typical outdoor scenario. As a result, traditional algorithms are less effective in avoiding unnecessary handoff triggered by vehicle-borne terminals with various speeds. Besides that, hysteresis and dwelling-timer thresholds usually need to be modified to satisfy different channel environments. For solving this problem, a vertical handoff algorithm based on Q-learning is proposed in this paper. Q-learning can provide the decider with self-adaptive ability for handling the terminals' handoff requests with different motion types and channel conditions. Meanwhile, Neural Fuzzy Inference System (NFIS) is embedded to retain a continuous perception of the state space. Simulation results verify that the proposed algorithm can achieve lower unnecessary handoff probability compared with the other two conventional algorithms.

1. Introduction

The fourth generation wireless communication system is expected to integrate several types of different wireless technologies. Future wireless network may consist of multiple layers such as cellular, WLAN, WiMAX, and satellite. It means that subscribers may always have more than one suitable network to select according to the preference and different characteristics of each type of wireless technologies. Common trends show that WLAN is an important supplement to cellular network, because the coexisting of cellular and WLAN can bring better quality of service to subscribers. In general, cellular system performs better in the services of high mobility and low latency, while it provides lower data rate than WLAN. Therefore the interworking and cooperation of these two different types of wireless network will become an important issue in the next generation cellular system.

Common trends show that vertical handoff can help the network satisfy the diverse Quality of Service (QoS) demands from different users. It is evident that the interworking of different types of wireless network is an important issue in the heterogeneous wireless network. Unnecessary handoff, which can degenerate the overall performance of network, is one of the most detrimental issues that need to be addressed with respect to handoff mechanism. It usually occurs when mobile dwells between two base stations, and then the base stations bounce the link with the mobile back and forth. In order to solve this problem, hysteresis-based [1, 2] and dwelling-timer-based [3] handoff trigger algorithms are generally used in sole cellular network.

In recent years, many kinds of vertical handoff algorithms are proposed for reducing the effect of unnecessary handoff in heterogeneous wireless network. In 2001, Ylianttila et al. analyzed the handoff delay in GPRS/WLAN network and proposed a dwell timer based handoff algorithm to optimize the handoff decisions [4]. Afterward, McNair and Zhu proposed a handoff algorithm by comprehensively considering service type, monetary cost, network conditions, et al. [5]. In 2007, W. Lee et al. presented a movement aware vertical handoff trigger scheme for WLAN/WiMAX heterogeneous

network. In this paper, an adaptive dwell timer was used to allow MS a better connection [6]. In the same year, A SINR based vertical handoff algorithm is provided by K. Yang. This algorithm converts the SINR value from one network to an equivalent value for the target network and then provides the handoff decision [7]. In addition to the above achievements, Chang proposed an adaptive vertical handoff algorithm with predictive RSS in 2008. Simulation results that validated this algorithm can reduce unnecessary handoff and improve dropping rate [8]. More recent studies were given by Haider et al. in 2011. They used intelligent fusion of adaptive threshold, signal trend, and dwell timer as the inputs of vertical handoff trigger algorithm and obtained a better result than many of traditional algorithms [9]. In [10], the authors proposed a distributed vertical handoff strategy for vehicle to vehicle and vehicle to infrastructure communication. The communication cost and transmitting time is discussed.

However these previous researches are mainly developed from the hysteresis-based and dwelling-timer-based algorithms, which are widely used in horizontal handoff of cellular network. In vertical handoff, the velocity factor has more imperative effect in handoff decision than in horizontal handoffs. The coverage of WLAN is much smaller than cellular network. Therefore, switching to WLAN when traveling at high speeds is likely to face the problem that a handoff back to the original network would occur very shortly afterwards. Since the procedure of a handoff behavior involves a set of signaling functions and consequently imposes both processing loads and signaling overhead to the network, unnecessary handoffs should be discouraged [10, 11]. Definitely, traditional algorithms are reliable on avoiding ping-pong handoff caused by the terminals dwelling at the edge of WLAN coverage, but they cannot essentially solve the unnecessary handoff triggered by vehicle-borne terminals. Moreover, hysteresis and dwelling-timer thresholds usually need to be modified to satisfy different channel environments.

Q-learning [12] is an online learning algorithm with outstanding adaptive ability. It has been widely explored in the field of automatic control. To overcome the limitations of classical Q-learning algorithm, for example, discrete state perception and discrete actions, Jouffe proposed the method that using NFIS to retain continuous perception of the state space [13]. It infers the global policy, which is relative to state, from local policies associated with each rule of the learner. NFIS is embedded to introduce generalization in the state space and generate continuous actions. On the other hand, Q-learning is used to tune a fuzzy controller. Note that, there were also some other approaches of extending fuzzy logic into Q-learning before [13] was proposed, such as those shown in [14]. In order to avoid confusion, we term the algorithm Q-learning Neural Fuzzy Inference System (Q-NFIS) in this paper instead of Fuzzy Q-learning (FQL) in [13], although it is developed from that.

The key contribution of this paper is stated as follows. Firstly, we present a Q-NFIS based vertical handoff trigger algorithm. Multiple RSS values from different AP and their rates of change were used as the input of handoff decider. It is certain that RSS values are related to the position of terminals, whereas their rates of change reflect the motion states. Consequently, the position and motion information are used as hidden function to trigger handoff decision. Then we propose an outdoor AP deployment scheme. Afterwards we analyze the unnecessary handoff triggered by vehicle-borne terminals with various speeds. Aiming at this issue, we provide the mathematical analysis for our results. Simulation results verify that the proposed algorithm can provide the decider with self-adaptive ability for handling the terminals' handoff requests with different motion types and channel conditions and achieve lower unnecessary handoff probability compared with the other two conventional algorithms. Note that many researches have proved that unnecessary handoff results in degradation of network performance [4–6, 8] with respect to throughput, delay et al. Therefore we mainly focus on how to reduce unnecessary handoff in this paper.

The scheme of this paper is organized as follows: we begin in Section 2 by discussing the mathematical model of Q-NFIS. Then we propose an AP deployment scheme for outdoor scenario in Section 3. Additionally, we derive the mathematical probability of unnecessary handoff triggered by vehicle-borne terminals with low dwelling time. In Section 4, we present the related numerical results. Meanwhile, two basic types of hysteresis-based and dwelling-timer-based handoff trigger methods are provided for comparison. Finally, Section 5 provides the conclusions of this paper.

2. Mathematical Model of Q-Learning Neural Fuzzy Inference System

The topology of Q-NFIS is shown in Figure 1. Essentially, Q-NFIS is a feed forward network with multiple layers. The neurons in different layers achieve different functions and are shown as follows in detail. The only information available for learning is the system feedback, which is the reinforcement signal according to the last action it has performed in the previous state. We use u_i^k and O_i^k to represent the input and output of the ith node in kth layer, respectively.

Layer 1. This layer consists of N neurons, which transmit the input value directly by

$$O_i^1 = u_i^1, \quad \forall i \in \{1, 2, \ldots, N\}. \tag{1}$$

In this work, we take a 4-dimensional vector as input variables, including two RSS values from different AP and their rates of change (RoC). Here we use the mean value of the RSS between the interval of current handoff request and the last one. This can reduce the deviation caused by shadow fading according to maximum likelihood estimation principle. Therefore the input can be represented by

$$\begin{aligned}\mathbf{U}^1(t) &= \left[u_1^1(t), u_2^1(t), u_3^1(t), u_4^1(t)\right] \\ &= \left[\overline{\mathrm{RSS}_1}(t), \overline{\mathrm{RSS}_2}(t), \mathrm{RoC}_1(t), \mathrm{RoC}_2(t)\right].\end{aligned} \tag{2}$$

Layer 2. The function of the T neurons in this layer is fuzzification of input value. As shown in (3) and (4), $\mathscr{M}(\cdot)$ is the linguistic variable related to input value; namely, O_i^2 is

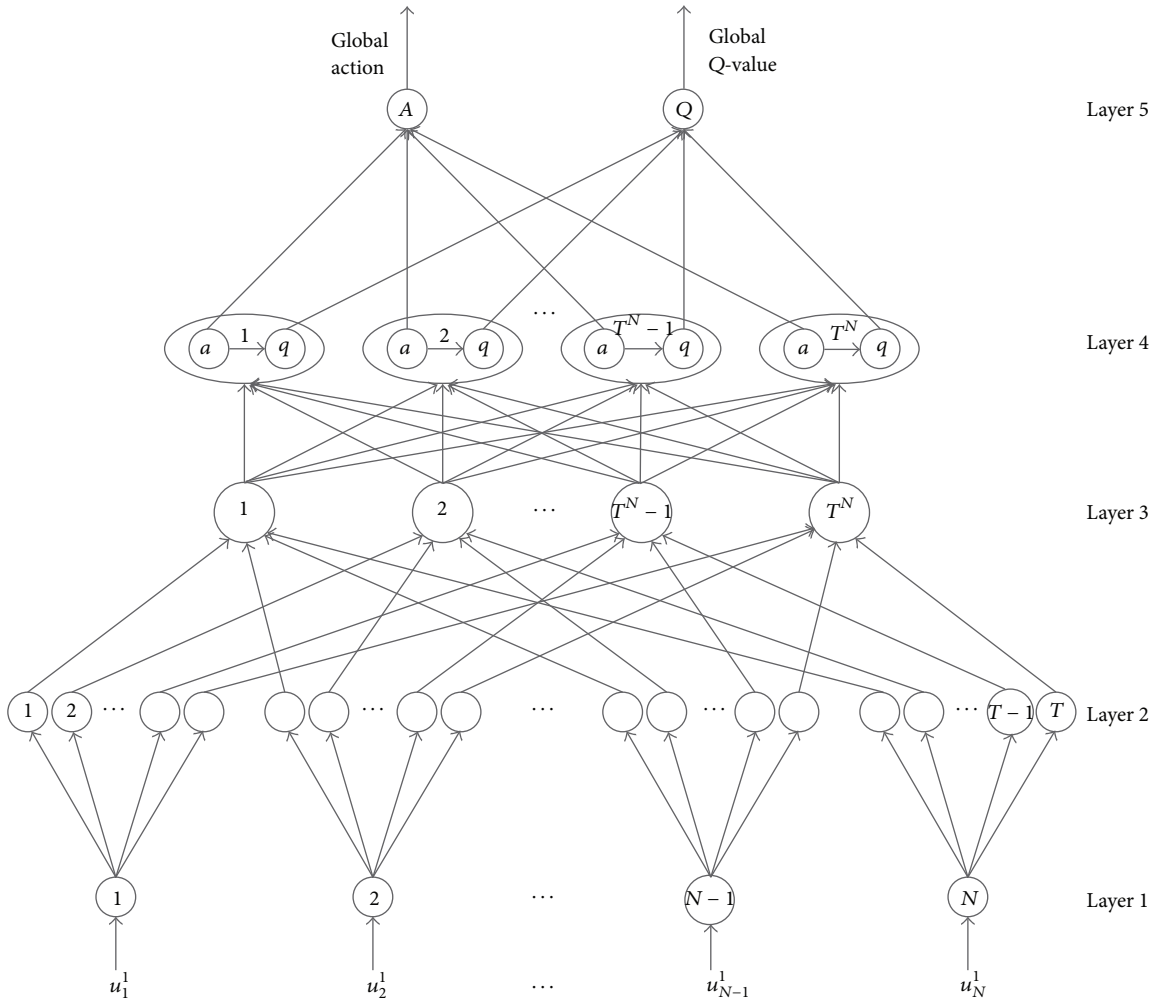


FIGURE 1: Topology of Q-learning neural fuzzy inference system.

the fuzzy membership value with respect to input. It reflects the degree that input value corresponds with $\mathscr{M}(\cdot)$. Gaussian Function is used as the parameterized membership function and the relationship between input and output is shown in (4). Each row in matrix $\mathbf{M}^{N\times T}$ is a linguistic variable set related to one dimension of the input state vector. The linguistic variable set should cover the entire area with respect to the possible distribution of the input and then each possible input case can be described precisely.

Consider the following:

$$O_i^2 = \mathbf{M}^{N\times T} = \begin{bmatrix} \mathscr{M}_{1,1}\left(u_1^2\right) & \cdots & \mathscr{M}_{1,T}\left(u_1^2\right) \\ \vdots & \ddots & \vdots \\ \mathscr{M}_{N,1}\left(u_N^2\right) & \cdots & \mathscr{M}_{N,T}\left(u_N^2\right) \end{bmatrix}, \quad (3)$$

$$\mathscr{M}_{i,j}\left(u_i^2\right) = \exp\left(-\frac{1}{2}\left(\frac{u_i^2 - m_{i,j}}{\sigma_{i,j}}\right)^2\right), \quad (4)$$

$$\forall i \in \{1,2,\ldots,N\};\ \forall j \in \{1,2\ldots,T\}.$$

Layer 3. Achieve the fusion of fuzzy rules, which equal to fuzzy multiplication operation by each input value. Define $u_{i,j}^3$ as the input from jth membership of O_i^1, and then we can obtain the strength of each rule by

$$O_k^3 = \prod_i u_{i,j}^3 = \prod_i \left(\left\{\mathscr{M}_{i,j_i}\left(u_i^2\right) \mid \forall \mathscr{M}_{i,j_i}\left(u_i^2\right) \in \mathbf{M}_i\right\}\right)$$

$$\forall i \in \{1,2,\ldots,N\},\ \forall j \in \{1,2,\ldots,T\},\ \forall k \in \left\{1,2,\ldots,T^N\right\}. \quad (5)$$

Layer 4. Every neuron in this layer includes a local action-reward pair which is represented as (a_i, q_i). Global action and global $\mathcal{Q}$ value are obtained by the fusion of local action and local q value, respectively. Local action is a finite set of output space predefined by system. With regard to a vertical handoff decider, the local action set can be defined as $\mathbb{A} = \{\text{Reject}(R), \text{Access}(A)\}$.

The local action $a_i \in \mathbb{A}$ is guided by the related local reward $q(\varphi_i, a_i)$. Assume that the optimal local action is a_i^*, which satisfies

$$a_i^* = \underset{a_i}{\operatorname{argmax}}\left\{q\left(\varphi_i, a_i\right)\right\}, \quad \forall a_i \in \mathbb{A}, \forall i \in \left\{1,2,\ldots,T^N\right\}. \quad (6)$$

As shown in (7), the output of Layer 4 is the normalized local action and q value, respectively. $\alpha_i(x)$ is the rule set and $q(\varphi_i, a_i)$ is the q value of state-action pair (φ_i, a_i).

Consider the following:

$$O_i^4 = \frac{\alpha_i(x) \cdot a_i^*}{\sum_{i=1}^{T^N} \alpha_i(x)}, \qquad \widehat{O}_i^4 = \frac{\alpha_i(x) \cdot q(\varphi_i, a_i^*)}{\sum_{i=1}^{T^N} \alpha_i(x)}. \tag{7}$$

Layer 5. Achieve the function of defuzzy by linear summation of local action and local q value, respectively.

Consider the following:

$$O_i^5 = \mathscr{A}^*\left(\mathbf{U}^1(t)\right) = \sum_{i=1}^{T^N} u_i^5, \qquad \widehat{O}_i^5 = \mathscr{Q}\left(\mathbf{U}^1(t), \mathscr{A}^*\left(\mathbf{U}^1(t)\right)\right) = \sum_{i=1}^{T^N} \widehat{u}_i^5. \tag{8}$$

Consider that the state transition is $(\mathbf{U}^1(t), \mathscr{A}^*(\mathbf{U}^1(t))) \xrightarrow{\text{handoff}} \mathbf{U}^1(t+\tau_l)$. According to the update principle from classical Q-learning [12], the global $\mathscr{Q}$ value is updated by (9), where α is learning rate, β is discount factor, $\mathscr{r}$ is reward value, and τ_l is the handoff latency.

Consider the following:

$$\begin{aligned}\mathscr{Q}'\left(\mathbf{U}^1(t), \mathscr{A}^*\left(\mathbf{U}^1(t)\right)\right) &= (1-\alpha)\,\mathscr{Q}\left(\mathbf{U}^1(t), \mathscr{A}^*\left(\mathbf{U}^1(t)\right)\right) \\ &\quad + \alpha\left(\mathscr{r} + \beta\mathscr{Q}\left(\mathbf{U}^1(t+\tau_l), \mathscr{A}^*\left(\mathbf{U}^1(t+\tau_l)\right)\right)\right).\end{aligned} \tag{9}$$

Then we can get the difference of $\mathscr{Q}$ value by

$$\begin{aligned}\Delta\mathscr{Q} &= \mathscr{Q}'\left(\mathbf{U}^1(t), \mathscr{A}^*\left(\mathbf{U}^1(t)\right)\right) - \mathscr{Q}\left(\mathbf{U}^1(t), \mathscr{A}^*\left(\mathbf{U}^1(t)\right)\right) \\ &= \alpha\left(r - \mathscr{Q}\left(\mathbf{U}^1(t), \mathscr{A}^*\left(\mathbf{U}^1(t)\right)\right)\right. \\ &\quad \left. + \beta\mathscr{Q}\left(\mathbf{U}^1(t+\tau_l), \mathscr{A}^*\left(\mathbf{U}^1(t+\tau_l)\right)\right)\right).\end{aligned} \tag{10}$$

Afterward, local q value can be updated according to the global $\mathscr{Q}$ value by

$$q'(\varphi_i, a_i^*) = q(\varphi_i, a_i^*) + \Delta q = q(\varphi_i, a_i^*) + \frac{\Delta\mathscr{Q} \cdot \alpha_i(x)}{\sum_{i=1}^{T^N} \alpha_i(x)}. \tag{11}$$

After terminal logging out, reward value in the form of positive or negative is evaluated according to the validity of the handoff decision and applied to tune local q value. Considering reward value reflects the evaluation to the handoff decision, it should be set as the normalized values which reflect the system performance as expected. For example, if low unnecessary handoff is required, the absolute value of reward in case 1 should be set bigger than that of the other three cases referring to Table 1. As a result, access policy becomes stricter and wrong reject rate in case 3 will grow as tradeoff.

Table 1: Reward scheme of Q-NFIS.

Case	Whether permit handoff request	Whether trigger unnecessary handoff	Reward value: positive (P), negative (N)
1	Y	Y	N
2	Y	N	P
3	N	Y	P
4	N	N	N

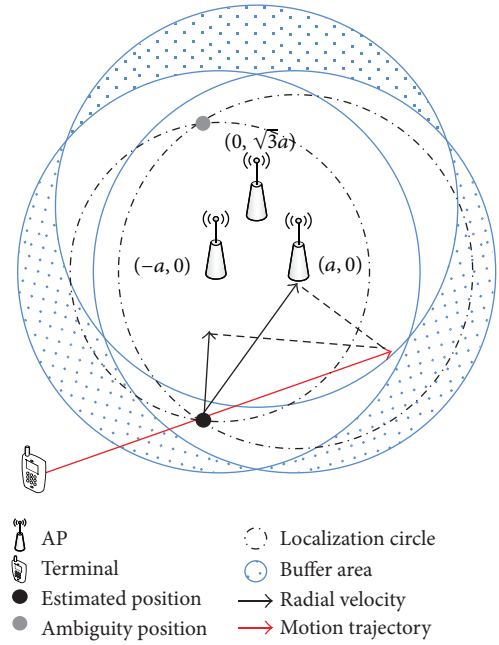


Figure 2: An outdoor AP deployment scheme.

3. Mathematical Analysis of Motion Model and Simulation Scenarion

3.1. Simulation Scenario. Inspired by the structure of cellular network, we propose an outdoor AP deployment scheme in this work. As shown in Figure 2, each three APs constitute a cluster. The advantage is that terminals are covered by more than one AP at most of the area. Therefore, they can receive more than one dimension of beacon signal, which can be used as the reference of handoff decision. Moreover, this structure can enhance the communication capacity for hotspot area. Referring to Figure 2, shadow region covered by single AP is regarded as buffer area, which is similar to hysteresis-based methods. It can reduce the unnecessary handoff triggered by terminals dwelling at the edge of WLAN coverage. We take a 4-dimensional vector as input variables, including two RSS values from different AP and their rates of change. For getting the changing rate of RSS, the handoff controller starts to work after collecting at least 2 sets of RSS at different time points. A typical log-normal propagation model of WLAN signal as (12) is adopted in this simulation.

Consider the following:

$$P(d) = P_t - P_0 - 10\gamma\log_{10}(d) + \varepsilon(\mu, \sigma). \tag{12}$$

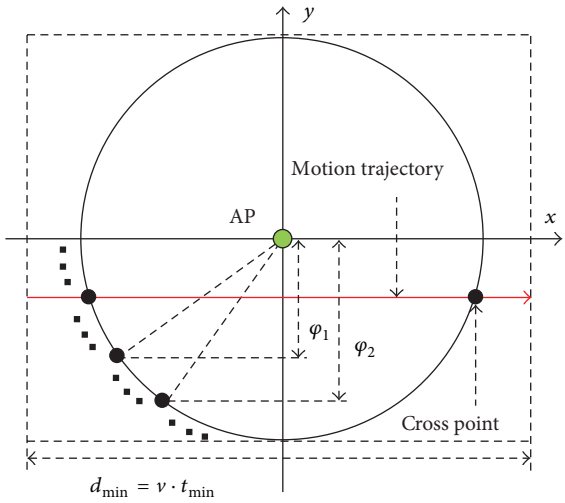


Figure 3: Schematic diagram of terminal moving through WLAN coverage.

3.2. Motion Model. The substantial idea of this paper is using Q-NFIS to estimate the terminal's motion information and then giving the handoff decision by predicting whether unnecessary handoff will be triggered. However, prediction is not always reasonable in real condition because terminal will not keep a same motion state all the time. The confidence of the handoff decision given by Q-NFIS relates to not only the performance of the algorithm itself but also the probability that terminals keep the stable motion state during the duration dwelling in WLAN coverage. Therefore this issue is analyzed based on a conditional random walk model for supporting the reasonability of our algorithm in this section. In addition, we derive an approximate estimation about the proportion of unnecessary handoff that can be predicted theoretically. Assuming the terminal's motion in accordance with random walk model, the probability of different angles that terminal moves into the coverage is the same. Therefore we only need to consider the case that terminal moves into the coverage of WLAN as horizontal direction, as shown in Figure 3.

Since the terminal cannot change velocity frequently in a real case, we may as well assume that terminals will keep a fixed velocity for $t_{\min}$ seconds at least. Based on the prerequisite, the trajectory will be $d_{\min} = vt_{\min}$ with fixed velocity. Assuming that r is the radius of WLAN coverage, p_{fx} and p_{ufx} are the probability that terminals pass through the coverage of WLAN with fixed and unfixed velocity, respectively. According to the different conditions of $d_{\min} \geq 2r$ and $d_{\min} \leq 2r$, p_{fx} and p_{ufx} can be obtained as follows.

(i) Case 1, $d_{\min} \geq 2r$:

$$\frac{p_{ufx}}{p_{fx}} = \frac{\int_0^r 2\left(r^2 - \varphi^2\right)^{1/2} d\varphi}{\int_0^r \left(d_{\min}/2 - \left(r^2 - \varphi^2\right)^{1/2}\right) d\varphi} = \frac{2\int_0^r \left(r^2 - \varphi^2\right)^{1/2} d\varphi}{rd/2 - \int_0^r \left(r^2 - \varphi^2\right)^{1/2} d\varphi}. \tag{13}$$

Table 2: Simulation parameters.

Parameters	Value	Description
P_t	20 mW	Transmit power of AP
P_0	37.3 dB	Path loss in the first meter
P_{th}	−85 dBm	Sensitivity of terminal
γ	3.3	Path loss exponent
$\varepsilon(\mu, \sigma)$	$\mu = 0,\ \sigma = 5$	Gaussian random noise
a	20 m	Parameter of AP location

The integration term $\int_0^r \left(r^2 - \varphi^2\right)^{1/2} d\varphi$ is given by

$$\int_0^r \left(r^2 - x^2\right)^{1/2} dx = \left(\frac{x}{2}\left(r^2 - x^2\right)^{1/2} + \frac{r^2}{2}\sin^{-1}\frac{x}{r}\right)\Bigg|_0^r = \frac{1}{4}\pi r^2. \tag{14}$$

Substitution of (14) into (13) leads to

$$\frac{p_{ufx}}{p_{fx}} = \frac{2\pi r}{2d_{\min} - \pi r}. \tag{15}$$

Considering that $p_{fx} + p_{ufx} = 1$, we can obtain

$$p_{fx} = \frac{2d_{\min} - \pi r}{2d_{\min} + \pi r}. \tag{16}$$

(ii) Case 2, $d_{\min} < 2r$.

Similar to Case 1, we can derive the result in (17), where $\Delta = \sqrt{r^2 - (d_{\min}/2)^2}$.

Consider the following:

$$\frac{p_{ufx}}{p_{fx}} = \frac{3\Delta d_{\min} + 2r^2 \tan^{-1}\left(d_{\min}/2\Delta\right)}{2rd_{\min} - \Delta d_{\min} - 2r^2 \tan^{-1}\left(d_{\min}/2\Delta\right)},$$
$$p_{fx} = \frac{2rd_{\min} - \Delta d_{\min} - 2r^2 \tan^{-1}\left(d_{\min}/2\Delta\right)}{2d_{\min}(r + \Delta)}. \tag{17}$$

According to the equations above, we can obtain p_{fx}, which is the probability of that terminals passing through the coverage of WLAN with fixed velocity. In other words, the motion information is reliable for handoff decision by probability of p_{fx}. Monte Carlo simulations were done to confirm the validity of the results.

4. Simulation Results and Analysis

Simulation scenario is shown in Figure 2. Relative parameters can be found in Table 2. As shown in Figure 2, two localization solutions will be obtained according to the propagation model when terminal is covered by two APs. Note that the ambiguity solution is always covered by three APs from the structure of WLAN coverage. Therefore, from the theoretical point of view, 2-dimensional RSS can reflect the location information to a certain degree when terminal implements a continuous motion state.

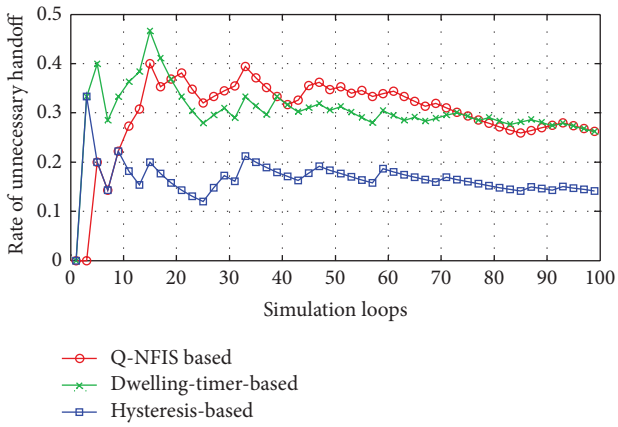


Figure 4: Comparison of unnecessary handoff rate for 3 algorithms in first 100 simulation loops.

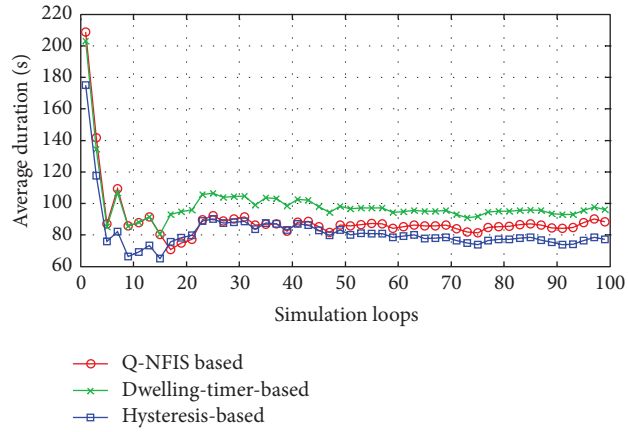


Figure 5: Comparison of average duration for 3 algorithms in first 100 simulation loops.

The motion of terminals is defined in accordance with conditional random walk model which is analyzed in Section 2, and $t_{\min}$ is set for 1 minute. In addition, we consider three typical motion types, which are pedestrian-borne terminal, bicycle-borne terminal, and vehicle-borne terminal with the velocity of 1 m/s, 3 m/s, and 15 m/s, respectively. They are all generated outside the coverage of WLAN at a random initial position. The trigger threshold of unnecessary handoff is set for 10 seconds. For referring to the performance of Q-NFIS based handoff trigger algorithm, two basic types of hysteresis-based and dwelling-timer-based handoff trigger methods are provided for comparison in the same scenario. In the hysteresis-based handoff scheme, handoff is triggered when any dimension of RSS is greater than $(P_{th} + 5)$ dBm. In the dwelling-timer-based handoff scheme, handoff is triggered after receiving 5 consecutive signals greater than P_{th} dBm.

Hysteresis and dwelling-timer based algorithms are both achieved at the cost of coverage essentially, which will reduce the average service time of users acquired. Similarly, the proposed algorithm also needs a buffer period to estimate the motion state. Based on the consideration above, besides the trigger rate of unnecessary handoff, we take users' average duration of accessing WLAN into account as well. In order to present the detailed tuning trends, the simulation results of first 100 loops are shown in Figures 4 and 5, which are separated from the overall performance in Figures 6 and 7, respectively.

None priori knowledge is added into Q-NFIS; therefore the trigger rate of unnecessary handoff is high at the beginning of simulation referring to Figure 4, while terminals' average duration is low referring to Figure 5. With the growing of simulation loops, Q-learning system is tuned by collecting more and more knowledge of state space state/action pair, and the improving of the performance from Figure 7 validates the online learning ability of the proposed algorithm. As we know, unnecessary handoff is mainly triggered by vehicle-borne terminals with low dwelling time in WLAN. According to the discussion in the previous section, we can obtain that the trigger rate of unnecessary handoff by

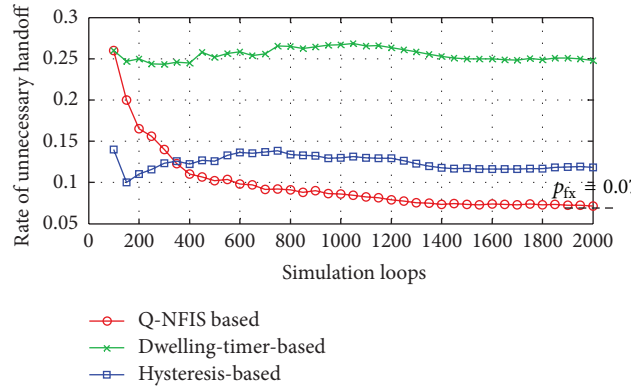


Figure 6: Comparison of unnecessary handoff rate for 3 algorithms in 2000 simulation loops.

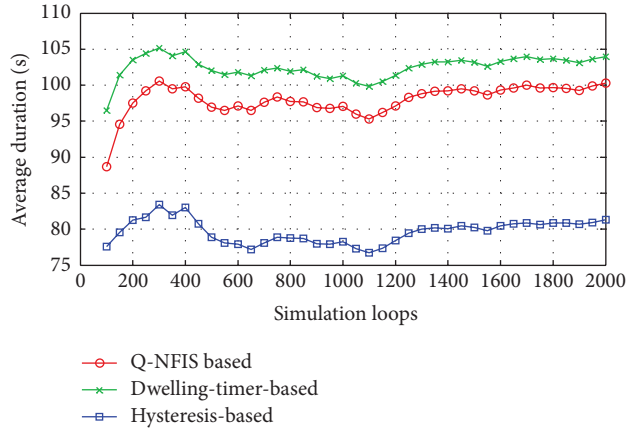


Figure 7: Comparison of average duration for 3 algorithms in 2000 simulation loops.

optimal handoff control is p_{ufx} approximately. From Figure 7, we can find that the simulation result by Q-NFIS is close to the optimal solution theoretically. From the simulation results shown in Figures 6 and 7, an acceptable rate of unnecessary handoff rate can be achieved by hysteresis based algorithm; however the average duration is much lower compared with

the other two algorithms. It is because hysteresis costs much WLAN coverage for the characteristic of its transmission model. By dwelling-timer based algorithms we can achieve a high average duration, because it will permit handoff requests as soon as the dwelling-timer is reached. This scheme is especially good for pedestrian type terminals; however, it will be less effective for vehicle-borne terminals. This is the reason why its rate of unnecessary handoff is much higher than the other two algorithms, and meanwhile it validates the discussion we present in Section 1.

As a result, we can find that unnecessary handoff rate of Q-NFIS reduced evidently with the growing of simulation loops and nearly converged at about 0.07 after 1500 simulation loops from Figure 6. Figure 7 shows that Q-NFIS achieves terminals' average duration higher than hysteresis based scheme while a little lower than dwelling-timer based scheme. This indicates that the proposed algorithm can provide reasonable motion predictions for handoff decision by sacrificing a little degree of duration.

5. Conclusions

In this paper, in order to solve the problem of unnecessary handoff caused by vehicle terminals with low dwelling time, a motion adaptive vertical handoff algorithm based on Q-NFIS is proposed. For supporting the reasonability of our algorithm, we provide the mathematical analysis about the unnecessary handoff that can be predicted theoretically. Simulation results validate the proposed that algorithm can reduce unnecessary handoff effectively by providing reasonable motion predictions for handoff decision. In addition, its performance outperforms two typical traditional handoff trigger algorithms in the same simulation scenario. Therefore we can draw the conclusion that it is a reasonable vertical handoff algorithm for cellular/WLAN heterogeneous wireless network.

Conflict of Interests

The authors declare no conflict of interests.

Acknowledgments

This paper was supported by the National Nature Science Foundation of China (Grant no. 61071105). The authors would like to thank all the members that participated in this work and also anonymous reviewers for their valuable comments.

References

[1] S. Su, J. Chen, and J. Huang, "Performance analysis of soft handoff in CDMA cellular networks," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 9, pp. 1762–1769, 1996.

[2] P. Marichamy and S. Chakrabarti, "On threshold setting and hysteresis issues of handoff algorithms," in *Proceedings of the IEEE International Conference on Personal Wireless Communications*, pp. 436–440, February 1999.

[3] A. E. Leu and B. L. Mark, "An efficient timer-based hard handoff algorithm for cellular networks," in *Proceedings of the IEEE Wireless Communications and Networking (WCNC '03)*, vol. 2, pp. 1207–1212, New Orleans, La, USA, March 2003.

[4] M. Ylianttila, M. Pande, J. Mäkelä, and P. Mähönen, "Optimization scheme for mobile users performing vertical handoffs between IEEE 802.11 and GPRS/EDGE networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '01)*, vol. 6, pp. 3439–3443, San Antonio, Tex, USA, November 2001.

[5] J. McNair and F. Zhu, "Vertical handoffs in fourth-generation multinetwork environments," *IEEE Wireless Communications*, vol. 11, no. 3, pp. 8–15, 2004.

[6] W. Lee, E. Kim, J. Kim, I. Lee, and C. Lee, "Movement-aware vertical handoff of WLAN and mobile WiMAX for seamless ubiquitous access," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 4, pp. 1268–1275, 2007.

[7] Y. Kemeng, I. Gondal, Q. Bin, and L. S. Dooley, "Combined SINR based vertical handoff algorithm for next generation heterogeneous wireless networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '07)*, pp. 4483–4487, Washington, DC, USA, November 2007.

[8] B. Chang and J. Chen, "Cross-layer-based adaptive vertical handoff with predictive RSS in heterogeneous wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3679–3692, 2008.

[9] A. Haider, I. Gondal, and J. Kamruzzaman, "Dynamic dwell timer for hybrid vertical handover in 4G coupled networks," in *Proceedings of the IEEE 73rd Vehicular Technology Conference (VTC Spring '11)*, pp. 1–5, Yokohama, Japan, May 2011.

[10] K. Shafiee, A. Attar, and V. C. M. Leung, "Optimal distributed vertical handoff strategies in vehicular heterogeneous networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 3, pp. 534–544, 2011.

[11] A. Hasswa, N. Nasser, and H. Hassanein, "Generic vertical handoff decision function for heterogeneous wireless networks," in *Proceedings of the 2nd IFIP International Conference on Wireless and Optical Communications Networks (WOCN '05)*, pp. 239–243, Dubai, UAE, March 2005.

[12] K. Shafiee, A. Attar, and V. C. M. Leung, "Optimal distributed vertical handoff strategies in vehicular heterogeneous networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 3, pp. 534–544, 2011.

[13] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[14] L. Jouffe, "Fuzzy inference system learning by reinforcement methods," *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 28, no. 3, pp. 338–355, 1998.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 801854, 7 pages
http://dx.doi.org/10.1155/2014/801854

Research Article

A Novel Complex Networks Clustering Algorithm Based on the Core Influence of Nodes

Chao Tong,[1,2] Jianwei Niu,[1] Bin Dai,[1] and Zhongyu Xie[1]

[1] *School of Computer Science and Engineering, Beihang University, Beijing 100191, China*
[2] *School of Computer Science, McGill University, Montreal, QC, Canada H3A 0E9*

Correspondence should be addressed to Jianwei Niu; niujianwei@buaa.edu.cn

Received 27 December 2013; Accepted 4 February 2014; Published 10 March 2014

Academic Editors: H. R. Karimi, X. Yang, Z. Yu, and W. Zhang

In complex networks, cluster structure, identified by the heterogeneity of nodes, has become a common and important topological property. Network clustering methods are thus significant for the study of complex networks. Currently, many typical clustering algorithms have some weakness like inaccuracy and slow convergence. In this paper, we propose a clustering algorithm by calculating the core influence of nodes. The clustering process is a simulation of the process of cluster formation in sociology. The algorithm detects the nodes with core influence through their betweenness centrality, and builds the cluster's core structure by discriminant functions. Next, the algorithm gets the final cluster structure after clustering the rest of the nodes in the network by optimizing method. Experiments on different datasets show that the clustering accuracy of this algorithm is superior to the classical clustering algorithm (Fast-Newman algorithm). It clusters faster and plays a positive role in revealing the real cluster structure of complex networks precisely.

1. Introduction

With the population of information networks and the discovery of the small world effect and the scale-free characteristic, research on complex networks has become a trend. Complex network study involves graph theory, statistical physics, computers, ecology, sociology, and economics [1]. Complex networks cover a variety of biology networks, the Internet/WWW networks, technology networks, social networks (such as the disease spreading networks and human relationships), and so on.

One of the most important features in complex networks is the cluster structure. Many studies have shown that some networks have cluster structures other than a large number of nodes only randomly linked. Heterogeneity has been found in many real-world networks. The heterogeneity of complex networks is embodied in more connections in similar types of nodes, while different types of nodes have fewer connections. These subgraphs with similar types of nodes and their connections are called "clusters."

Clustering algorithm plays a basic role in studying the cluster structure of complex networks. It has not only important theoretical significance in researching complex network topology, understanding the network function, revealing hidden laws, and predicting the network behavior but also broad application prospects. Clustering algorithm has been applied to the social network analysis, biological network analysis, search engine, spatial data clustering and image segmentation, and many other areas [2].

According to the analysis strategy, complex network clustering methods are divided into optimization methods and heuristic methods. The earlier clustering algorithms like spectral method [3–6] and the Kernighan-Lin algorithm (KL algorithm) [7] are optimization methods. Spectral method, derived from the early resolution of the graph partition problem, uses the quadratic optimization techniques to minimize a predefined "cut" function. A partition with minimum "cut" is considered to be an optimal network partition. With rigorous mathematical theories, the spectral method is widely used in graph partitioning and spatial points clustering. However, high reliance on the prior knowledge and adoption of bipartite recursion strategy makes it inadequate in complex multiple clusters networks.

KL algorithm is also based on the idea of graph partition, which aims at minimizing the difference between the number of intercluster connections and internal connections. By continuously adjusting clusters, the algorithm chooses and accepts the candidate solutions that can get the minimization of the objective function. KL algorithm, very sensitive to the initial solution and highly dependent on the prior knowledge, often gets local optimal results.

Girvan and Newman proposed GN algorithm [8], which uses heuristic strategy by repeatedly identifying and removing the connections between clusters. GN is a big time-consuming and space-consuming algorithm, resulting from the complexity of the edge betweenness calculation ($O(m \times n)$). Thus, it is difficult to perform well in a large network.

Based on the Maximum Flow-Minimum Cut Theorem, Flake et al. proposed a heuristic clustering algorithm, the Maximum Flow Community [9] (MFC algorithm). By calculating the minimum cut sets, MFC algorithm identifies intercluster connections that give rise to the network "bottleneck" and gradually split the network into cluster units by removing the connections between clusters. However, the MFC algorithm performs clustering based on connections and cannot be applied to the network with heterogeneous nodes.

Newman proposed a fast clustering algorithm based on local search [10], the Fast-Newman. The algorithm is an optimizing algorithm aiming at maximizing the network modular evaluation function (Q function) that Newman put forward in the same year. The Q function, denoted by the difference between the number of connections within a cluster and the expected number of connections in a random state, is used to show the pros and cons of the cluster structure. Larger Q value means a better clustering structure.

Based on the FN algorithm, Guimera and Amaral similarly adopted the Q function as the optimization objective function and proposed a complex networks clustering algorithm based on simulated annealing (SA), the GA algorithm [11]. The algorithm evaluates candidate solutions by calculating the corresponding Q function value and calculates the probability of accepting a candidate solution through the SA model. GA algorithm has the ability to find the global optimal solution and thus has good clustering performance.

Although optimization algorithms based on the Q function perform well in the community clustering, a number of issues remain unresolved due to the unpredictability of complex networks and the biased characteristic of Q function. Consider the following.

(1) Since the community detection results through the clustering algorithms based on optimization depend on the objective function to be optimized, "biased" objective function will inevitably lead to "biased" solution. The Q function currently widely used is a biased objective function [12], by which the results cannot completely and accurately reflect the real network structure. When the Q function reaches the global maximum, the clustering result is not optimal.

(2) With larger scale of complex networks, the calculation of the objective function and the iterative process become more complex, resulting in more and more time and resources consumed.

(3) Though the clustering algorithm based on heuristics method is able to handle the large-scale data in complex networks, compared to the optimization algorithm, it has lower clustering accuracy and cannot give high-precision clustering results.

To solve the above problems, we proposed a novel clustering algorithm based on the core influence of nodes. The algorithm combines heuristics method with optimization method. Its clustering process is designed to simulate the driven process of the cluster formation in sociology, to reflect the clustering process of nodes in the real network more accurately, and to achieve "no biased" precise clustering as far as possible.

The rest of this paper is organized as follows. In the next section, we introduce the clustering algorithm based on the core influence of nodes, then the experimental results and analysis are illustrated in Section 3, and, finally, a conclusion is drawn in Section 4.

2. Clustering Algorithm Based on the Core Influence of Nodes

The basic idea of the clustering algorithm based on the core influence of nodes is to identify the nodes with core influence based on the betweenness centrality theory, build the core structure of clusters with these nodes in the complex network through the evaluation function, and, finally, cluster the remaining nodes in the network using optimizing methods. Thus, clusters of the whole network can be obtained.

2.1. The Definition of the Core Influence of Nodes. The core influence of nodes in complex networks is denoted by the centrality of nodes. Centrality refers to the use of metric methods to evaluate the center position of a node in the network. It describes whether there are cores, how many cores there are, and how these cores are in the network.

Centrality has many definitions in complex networks, such as the degree centrality, the compactness centrality, the betweenness centrality, and the flow betweenness centrality. In order to reveal the role the nodes play in the transferring process of information, material, and energy in the complex network, this paper uses the betweenness [13] (the number of geodesics through the node) to define the centrality and the core influence of nodes.

Geodesic is defined as the path with least edges between two nodes. Thus, betweenness centrality of node x [14] is defined as

$$C_B(x) = \frac{2\sum_{i<j} g_{ij}(x)}{(n-1)(n-2)\, g_{ij}}, \quad (1)$$

where g_{ij} is the total number of geodesics between node i and j, $g_{ij}(x)$ is the number of geodesics through the node x between node i and j (the betweenness of node x), and $(n-1)(n-2)/2$ is the maximum value of the betweenness

of the node x (any geodesic between other two nodes goes through the node x).

Betweenness centrality partially describes the core influence of nodes in complex networks. However, betweenness centrality itself is a global evaluation parameter, which cannot accurately describe the relative influence of nodes in the local environment, especially in large-scale complex networks. Therefore, combining the betweenness centrality and local clustering features of nodes, the core influence of nodes is denoted as

$$C_A(x) = \frac{C_B(x)}{2E_x/(k_x(k_x-1))}, \quad (2)$$

where $C_B(x)$ is the betweenness centrality of the node x, k_x represents the number of neighbor nodes of the node x, E_x denotes the total edges between the neighbors.

The definition of the core influence above accurately describes how important a node is in its clustering environment. Higher core influence of a node indicates higher contribution and heavier load in the information dissemination process in a complex network. Meanwhile, different from the simple degree centrality, a node with the highest core influence is not probably the node with the maximum degree or a topological center in the network structure.

2.2. The Determination of Cluster's Core Structure. In complex networks, the core structure of clusters is usually not only a simple single node with high core influence but possibly also a certain structure composed of several active nodes with high influence [15]. In order to determine the core structure of a cluster by the core influence of nodes, the K function is used here as the evaluation function to determine the nodes that compose the core structure of clusters.

The goal of the K function is to determine whether the node can become the core of an independent cluster. It compares the actual connections and expected connections between a node with high core influence and the cluster it belongs to. The function is defined as

$$K(i) = \frac{m_i}{(d_i/d) \times \left(\left(d_q - d_i\right)/(d - d_i)\right) \times m}, \quad (3)$$

where m_i is the number of edges between node i and other nodes in the cluster that node i belongs to, m denotes the total number of edges in the whole network, d_i represents the degree of node i, d_q is the sum of degrees of nodes in the cluster, and d denotes the sum of degrees of nodes in the whole network. According to the definition of $K(i)$, there is a higher probability that a node becomes the core of an independent cluster when its $K(i)$ is smaller; while a node attached to a rather larger $K(i)$ plays a more influential role in its current cluster.

According to Fortunato and Barthélemy's study of the Q function's value range on a large number of real datasets [12], it can be estimated that the value range of K function is [1.96, 2.71] when a node is thought to be the core of an independent cluster. Besides, according to the Pareto rule, 20% of the nodes in the network with the highest core influence determine the main cluster structure framework. Plenty of datasets reveal that 20% of the nodes in the network with the highest core influence can determine the core cluster structure framework after being evaluated by K function, which is consistent with the Pareto rule.

2.3. Clustering Algorithm. After determining the core cluster structure, the algorithm clusters the remaining nodes by optimizing method. The remaining nodes are centralized by rearranging all nodes regarding their core influence. A "centralized" network can thus be obtained, where nodes are arranged from inside to outside. The objective "centralization function" that reflects the level of centralization is then defined as

$$C_A^g = \frac{\sum_{x \in W}\left(C_A^* - C_A(x)\right)}{(n-1)\max\left(C_A^* - C_A(x)\right)}, \quad (4)$$

where W represents a complex network and $C_A^* = \max_{x \in W} C_A(x)$ represents betweenness centrality of the node that has the maximum core influence.

The objective function shows that if all nodes have the same core influence, which indicates that the network is noncore, then $C_A^g = 0$; if the core influence of a node is 1, while other nodes remain 0, $C_A^g = 1$. Therefore, the higher the level of centralization of the network is, the greater the value of the objective function is.

The strategy to search and accept the candidate solution is as follows. Firstly, arrange all nodes descendingly according to their core influence. Then, change the structure of the cluster a node belongs to and then calculate the corresponding "centralization function." And accept the candidate solution that maximizes the sum of the value of the whole network's "centralization function." The process ends when all nodes are classified into their own respective cluster structure.

2.4. Algorithm Implementation. According to the algorithm, the actual steps of the clustering algorithm based on the core influence are as follows.

(1) Sort all nodes by betweenness centrality in descending order $\{n_i\}$, satisfying the requirement that when $i < j, C_i > C_j$.

(2) Set up three groups of nodes; $P[i]$ represents that the node has been clustered, $Q[i]$ represents that the node has not been clustered, and $R[i]$ represents that the node is in cluster-controversy. Initially, all nodes are in Q.

(3) Select the node n_1 with the highest degree and all the nodes connected to n_1 are defined as a cluster, while n_1 is the core of Cluster 1 and all the nodes are classified into the node group P.

(4) Judge whether n_2 is the core. If n_2 is not in the node group P, n_2 is the core of Cluster 2.

(5) If n_2 is within the node group P, use the criteria function K. If $K \in [1.96, 2.71]$, n_2 is the core of Cluster 2; otherwise, classify the adjacent nodes of n_2 into

Table 1: Neural Network dataset properties.

Properties	Values
Number of nodes	297
Average clustering coefficient	0.2924
Number of edges	2359
Diameter	5
Number of triangles	3241
Average shortest path length	2.4553

the cluster where n_2 is. Repeat step (4) and judge node $n_3, n_4 \ldots$.

(6) For all nodes connected to Cluster 2's core, classify those in Q into Custer 2 and transfer them into P, while transferring those originally of the P to R.

(7) Traverse the nodes in R through the optimization function "centric level," redetermine their respective cluster, and transfer the nodes from R to P after they have entered the selected cluster.

(8) Return to step (4) and iterate and traverse all the nodes.

3. Experimental Results and Analysis

For more objective and comprehensive evaluation, the algorithm is tested on three datasets (Neural Network [16], Political Blogs [17], and Email [18]) of different sizes and properties. The results are analysed and evaluated using the Conductance and Expansion results evaluation function in network, the Community Profile [19] (NCP). The two functions are defined as

$$\text{Conductance}: f(S) = \frac{c_S}{2m_S + c_S}, \qquad \text{Expansion}: f(S) = \frac{c_S}{n_S}, \tag{5}$$

where c_S represents the number of edges on the boundary of S, m_S denotes the total number of edges within the Cluster S and n_S is the total number of nodes in Cluster S. Lower value of the two evaluation functions signifies better clustering effect.

3.1. Neural Network Dataset Experiment. In this section, the algorithm is tested on the dataset "Neural Network." The dataset is a complex network of neurons in a living system, where each node represents a complete and independent neuron and the edge denotes the connection between neurons. The properties of the network are shown in Table 1.

The number of nodes and edges and the diameter describe the overall size of the network. The average clustering coefficient, the number of triangles, and the average shortest path length describe the relative tightness of the network and how obvious the clustering feature is.

The evaluating values of the clustering effect of two algorithms are shown in Figure 1. It can be calculated through

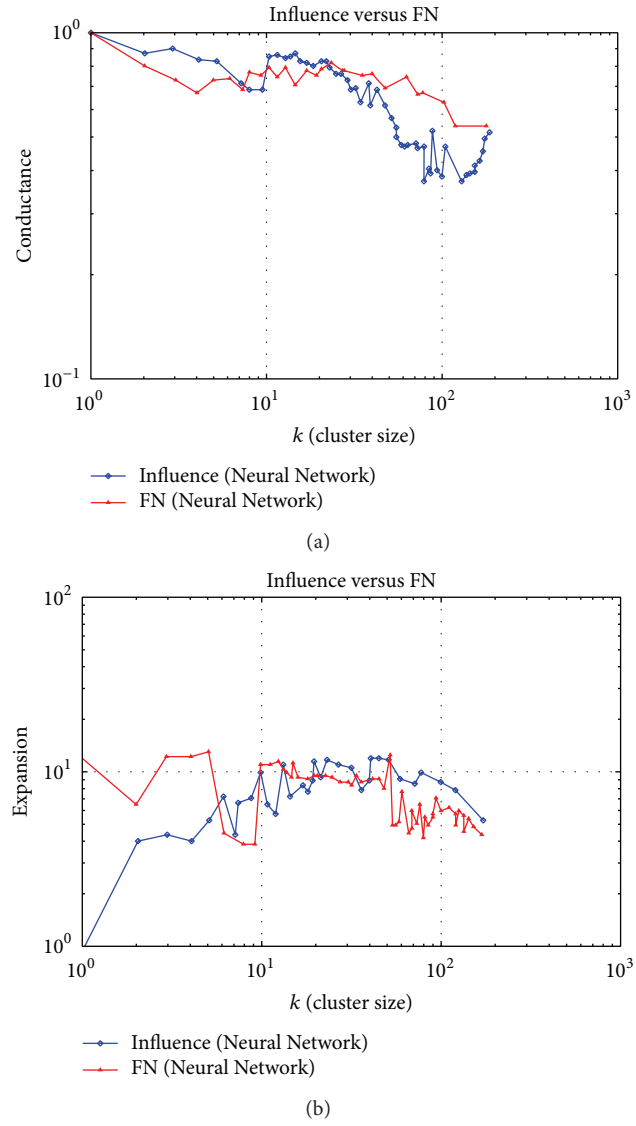


Figure 1: Evaluating values of the clustering effect of the Neural Network dataset.

the data shown in Figure 1(a) that the average Conductance of the influence algorithm is 0.540144, while the average of FN algorithm is 0.736532. The figure also shows that when the cluster size grows, the clustering effect of the influential algorithm improves, better than the FN algorithm. From Figure 1(b), it is calculated that the average value of the Expansion of the influence algorithm is 6.700091 and the average of FN algorithm is 8.205680. And, with the increase of the cluster size, the influence algorithm performs better than the FN algorithm in accuracy by 22.5%.

In this dataset, neurons have explicit functions and every neuron does not get global information. As a result, the FN algorithm cannot cluster precisely. The influence algorithm, proposed by us, however, digs out neurons with similar functional properties more precisely by considering the role every neuron plays in the process of information dissemination and gives the structural relationship among

Table 2: Political Blogs dataset properties.

Properties	Values
Number of nodes	1222
Average clustering coefficient	0.3203
Number of edges	16717
Diameter	8
Number of triangles	101043
Average shortest path length	2.7375

neurons of similar functions and among neurons clusters of different functions. The clustering results help medical researchers understand the mechanism of nervous system better so that they can analyze causes of neurological diseases and provide theoretical support for cures [20].

3.2. Political Blogs Dataset Experiment. In this section, the algorithm is tested on the dataset "Political Blogs." The dataset is a political blog network in complex social networks, where each node represents a politician and the edge denotes the real social relations between them. Compared with the Neural Network dataset, Political Blogs dataset has a larger scale, where the number of nodes increases by 3.1 times and the number of edges increases by 7 times. So, the connections between nodes are closer, and the clustering coefficient and the number of short circuits (triangle closure) increase in the network. On the other hand, the average shortest path between nodes becomes longer, indicating that the increase of the tightness of relationships between nodes is limited, though the network is larger. The properties of the network are shown in Table 2.

The evaluating values of the clustering effect of two algorithms are shown in Figure 2. It can be calculated through the data shown in Figure 2(a) that the average Conductance of the influence algorithm is 0.540144, while the average of FN algorithm is 0.736532. The value of Conductance of the influence algorithm is lower than the FN algorithm in 82.57% of all the cases. From Figure 2(b), it is calculated that the average value of the Expansion of the influence algorithm is 9.466124 and the average of FN algorithm is 16.379612. The clustering accuracy of the influence algorithm is better than the FN algorithm in 85.61% of all the cases.

In the comparison with Figure 1, the influence algorithm also keeps a high clustering accuracy, but the fluctuation range of the accuracy is wider. This indicates that as the complex network size increases, the local differences of the core influence and clustering feature of nodes increase. The influence algorithm reflects the local differences of nodes and reveals the significance of core influence nodes in the clustering process. It digs out the faction relationships among politicians more precisely.

The cluster analysis of the Political Blogs dataset by the influence algorithm is the theoretical basis of information diffusion and behavior spread in politics. For politicians, the clustering results help individuals to predict the support and resistance in the dissemination of their political opinion.

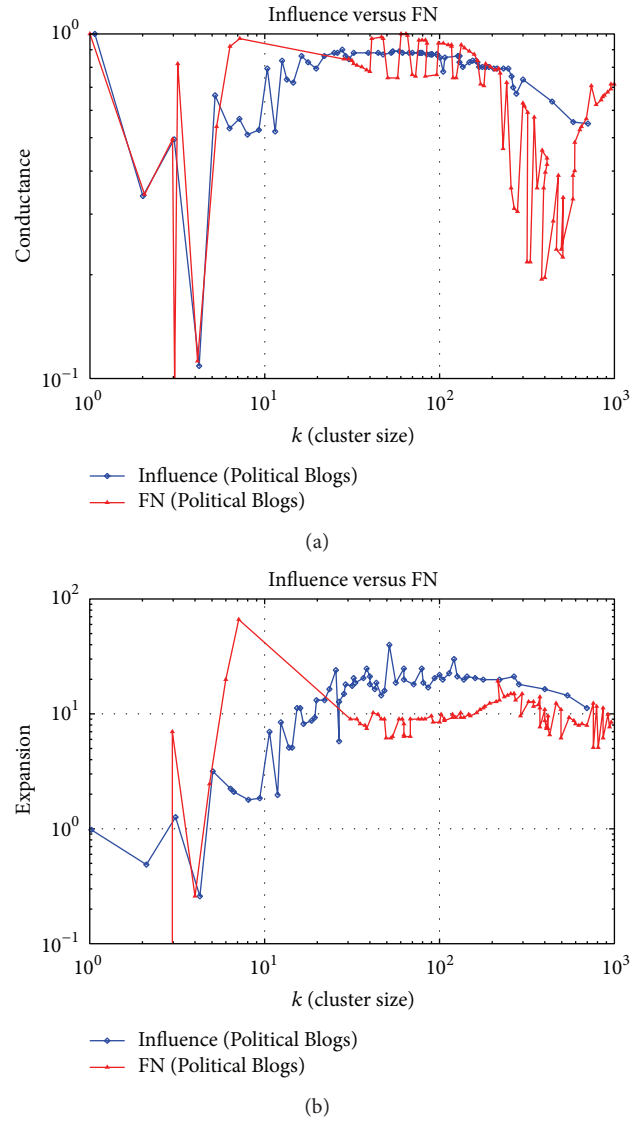


Figure 2: Evaluating values of the clustering effect of the Political Blogs dataset.

The results also help predict the probability of the pass of a political proposal and even the election result.

3.3. Email Dataset Experiment. In this section, the algorithm is tested on the dataset "Email" in the social system, which is established by receiving and sending emails. Each node represents an email address and two nodes are connected when they have email exchanges in history.

Compared with the first two datasets, the "Email" dataset contains fewer nodes and sparser connections. Thus, it has lower clustering coefficient and larger average value of shortest paths. In this case, the locality of nodes is stronger and the probability for nodes to grasp global information is smaller. The properties of the network are shown in Table 3.

The evaluating values of the clustering effect of two algorithms are shown in Figure 3. It can be calculated through the data shown in Figure 3(a) that the average Conductance

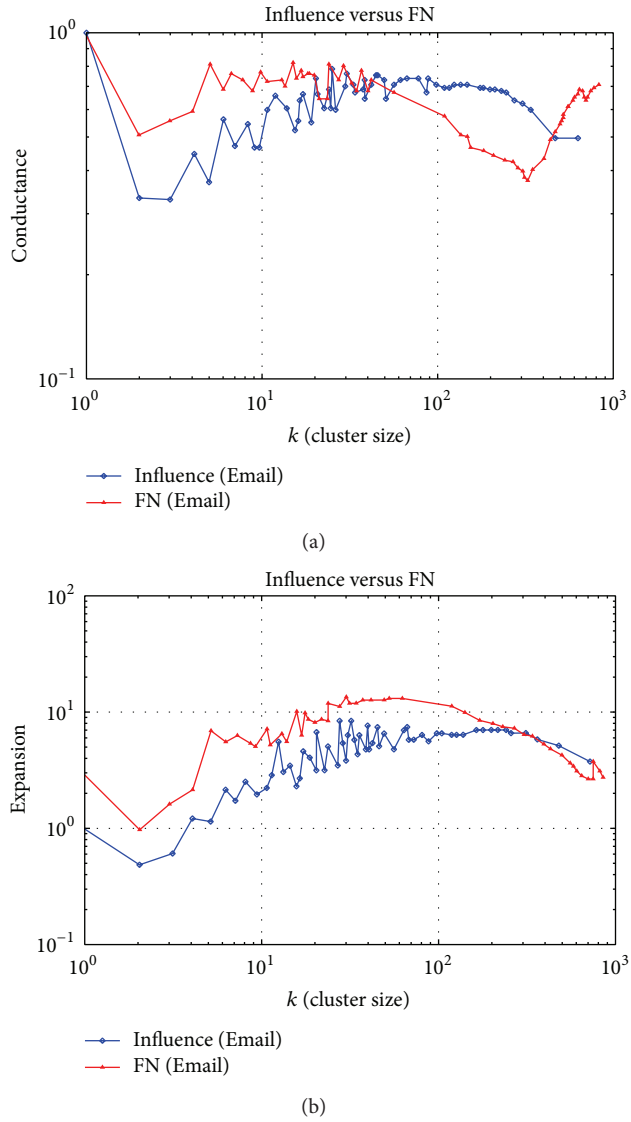


Figure 3: Evaluating values of the clustering effect of the Email dataset.

of the influence algorithm is 0.653043, while the average of FN algorithm is 0.664280. From Figure 3(b), it is calculated that the average value of the Expansion of the influence algorithm is 5.263551, while the average of FN algorithm is 4.619496. The clustering accuracy of the influence algorithm only has slight improvement compared to the FN algorithm. This is because the cluster coefficient becomes smaller with the expansion of clusters, leading to less and even the loss of difference of core influence among nodes. And less difference of core influence weakens the identity of the core cluster structure of the algorithm, indicating that the algorithm has limitations when processing sparse network with high homogeneity.

The experimental results on three datasets show that the clustering accuracy of the influence algorithm on large-scale complex networks increases variously compared to the FN algorithms. The effect is especially prominent for large-scale networks or networks with high heterogeneity. Studies have shown that when the size of a cluster is in the range of 50 to 100, the structure is relatively stable and real, and the effect of the clustering algorithm based on the core influence of nodes is much better than the FN algorithm in this interval.

Table 3: Email dataset properties.

Properties	Values
Number of nodes	1133
Average clustering coefficient	0.2202
Number of edges	5452
Diameter	8
Number of triangles	5453
Average shortest path length	3.6060

4. Conclusion

In this paper, to solve the biasness in traditional clustering methods, we proposed an algorithm based on the core influence of nodes. On the basis of the core influence of nodes, the algorithm simulates the driven process of cluster formation in sociology. It absorbs the advantages of both heuristic and optimizing algorithms and reflects the real clustering process in a more accurate way. The clustering experiments on different datasets conclude that the clustering accuracy of this algorithm is superior to the classic clustering algorithm (FN algorithm) in complex networks. Meanwhile, this algorithm runs faster and plays a positive role in revealing the real cluster structure of complex networks.

Future studies can be conducted in two directions. Firstly, improve the algorithm based on the core influence of nodes to achieve higher accuracy and prove the "unbiased" nature of its clustering results. Secondly, optimize the iterative strategy of the algorithm to further improve the clustering efficiency when handling large-scale networks.

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 the Research Fund of the State Key Laboratory of Software Development Environment under Grant no. BUAA SKLSDE-2012ZX-17, the National Natural Science Foundation of China under Grant nos. 61170296 and 61190125, the Program for New Century Excellent Talents in University under Grant no. NECT-09-0028, the Natural Science Foundation of Beijing, China, under Grant no. 4123101, and the Science Foundation of China University of Petroleum, Beijing (no. KYJJ2012-05-22).

References

[1] A. Vespignani, "Complex networks: the fragility of interdependency," *Nature*, vol. 464, no. 7291, pp. 984–985, 2010.

[2] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, "Limits of predictability in human mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010.

[3] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 23, pp. 8577–8582, 2006.

[4] M. Shiga, I. Takigawa, and H. Mamitsuka, "A spectral clustering approach to optimally combining numericalvectors with a modular network," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '07)*, pp. 647–656, August 2007.

[5] S. White and P. Smyth, "A spectral clustering approach to finding communities in graphs," in *Proceedings of the 5th SIAM Inteantional Conference on Data Mining*, pp. 76–84, SIAM, Philadelphia, Pa, USA, 2005.

[6] L. Donetti and M. A. Munoz, "Improved spectral algorithm for the detection of network commu-nities," in *Proceedings of the of the 8th International Conference on Modeling Cooperative Behavior in the Social Sciences*, vol. 779, pp. 104–107, American Institute of Physics, New York, NY, USA, 2005.

[7] M. E. J. Newman, "Detecting community structure in networks," *European Physical Journal B*, vol. 38, pp. 321–330, 2004.

[8] 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.

[9] G. W. Flake, S. Lawrence, C. Lee Giles, and F. M. Coetzee, "Self-organization and identification of web communities," *Computer*, vol. 35, no. 3, pp. 66–71, 2002.

[10] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Physical Review E*, vol. 69, no. 6, Article ID 066133, 2004.

[11] R. Guimera and L. Amaral, "Functional cartography of complex metabolic networks," *Nature*, vol. 433, pp. 895–900, 2005.

[12] S. Fortunato and M. Barthelemy, "Resolution limit in community detection," *Proceedings of the National Academy of Science*, vol. 104, pp. 36–41, 2007.

[13] W. Lin and J. J. Zhang, "Centrality of complex networks," *Complex System and Complexity Science*, vol. 3, p. 1, 2006.

[14] L. C. Freeman, "Centrality in social networks conceptual clarification," *Social Networks*, vol. 1, no. 3, pp. 215–239, 1978.

[15] Q. Wu, X. Qi, E. Fuller, and C. Zhang, "'Follow the Leader': a centrality guided clustering and its application to social network analysis," *The Scientific World Journal*, vol. 2013, Article ID 368568, 9 pages, 2013.

[16] L. K. Hansen and P. Salamon, "Neural network ensembles," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 10, pp. 993–1001, 1990.

[17] K. Wallsten, "Agenda setting and the blogosphere: an analysis of the relationship between mainstream media and political blogs," *Review of Policy Research*, vol. 24, no. 6, pp. 567–587, 2007.

[18] http://snap.stanford.edu/data/email-Enron.html, 2012.

[19] 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.

[20] S. Yin, S. Ding, A. Haghani, H. Hao, and P. Zhang, "A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process," *Journal of Process Control*, vol. 22, no. 9, pp. 1567–1581, 2012.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 802915, 9 pages
http://dx.doi.org/10.1155/2014/802915

Research Article

Energy-Efficient Routing Control Algorithm in Large-Scale WSN for Water Environment Monitoring with Application to Three Gorges Reservoir Area

Yuanchang Zhong,[1,2] Lin Cheng,[1] Liang Zhang,[1] Yongduan Song,[2] and Hamid Reza Karimi[3]

[1] *College of Communication Engineering, Chongqing University, Chongqing 400044, China*
[2] *School of Automation, Chongqing University, Chongqing 400044, China*
[3] *Department of Engineering, Faculty of Engineering and Science, University of Agder, N-4898 Grimstad, Norway*

Correspondence should be addressed to Yuanchang Zhong; zyc@cqu.edu.cn

Received 4 November 2013; Accepted 22 January 2014; Published 5 March 2014

Academic Editors: P. Garcia-Teodoro and Z. Sun

The typical application backgrounds of large-scale WSN (wireless sensor networks) for the water environment monitoring in the Three Gorges Reservoir are large coverage area and wide distribution. To maximally prolong lifetime of large-scale WSN, a new energy-saving routing algorithm has been proposed, using the method of maximum energy-welfare optimization clustering. Firstly, temporary clusters are formed based on two main parameters, the remaining energy of nodes and the distance between a node and the base station. Secondly, the algorithm adjusts cluster heads and optimizes the clustering according to the maximum energy-welfare of the cluster by the cluster head shifting mechanism. Finally, in order to save node energy efficiently, cluster heads transmit data to the base station in single-hop and multihop way. Theoretical analysis and simulation results show that the proposed algorithm is feasible and advanced. It can efficiently save the node energy, balance the energy dissipation of all nodes, and prolong the network lifetime.

1. Introduction

In recent years, the WSN technology has attracted extensive attention of the academia, industry and, government. It became one of the most competitive technologies among a lot of fields, such as, national defense and military, environment monitoring and forecasting, healthcare, smart home, building structure monitoring, complex mechanical control, urban transportation, space exploration, management of large workshop and warehouse, and safe monitoring of airport and large industrial parks [1, 2]. However, these sensor nodes in WSN are usually powered by batteries; the energy of nodes is limited and it is difficult to replace or recharge the batteries because of the wild environment and the huge number of nodes. The limited energy of nodes seriously affects the lifetime of WSN, which has constrained the large-scale application of WSN. Hence, it is important to improve the energy efficiency and prolong the network lifetime under the premise of limited energy, which is the key technical problems of WSN needing to be solved [3].

Designing an energy-efficient routing algorithm is the best way to prolong network lifetime of WSN and improve the quality of information transmission [4, 5]. Currently, a variety of routing protocols have been proposed to save energy. Existing routing protocols are generally divided into two categories: flat routing and hierarchical routing [6]. Flat routing is easy to implement, no additional cost of topology maintenance and packet routing, but it is not proper to wireless sensor networks. Hierarchical routing, also known as clustering routing, such as low-power Adaptive Clustering Hierarchy Protocol LEACH [7] and PEGASIS [8] protocols, has proposed the methods using cluster heads to form the clusters. Researches show that the hierarchical routing protocol is better than the planar routing protocol in the aspects of adaptability and energy efficiency. Thus hierarchical routing protocol received extensive attention [9, 10].

The application of wireless sensor networks for water environment monitoring in the Three Gorges Reservoir area is a typical case of large-scale wireless sensor networks research. And it has the typical characteristics of zonal distribution and large coverage area. Our research is based on the achievements we already obtained and deeply study on the existing routing algorithms, such as, LEACH and PARPEW [11] algorithm, and the focus is on the energy saving routing algorithm for large-scale WSN. So an improved energy-saving routing algorithm based on maximum energy-welfare optimization clustering has been proposed. In the simulation experiment, a rectangular area with greater aspect ratio is used to approach the actual banding distribution. Simulation results show that the performance of the proposed algorithm is superior to the other two algorithms'.

2. The Network Model of Large-Scale Wireless Sensor Networks

The Three Gorges Project is the world-famous water conservancy project, which brings the enormous economic and social benefits, while also triggering ecological environmental safety issues. Because that the self-purification capacity of static water is much lower than flowing water, the Three Gorges Reservoir water environment was gradually deteriorating and the water environment security became the focus of attention at home and abroad [12]. Currently, the environmental monitoring stations have been established in the Three Gorges Reservoir area to undertake a very important water environment monitoring tasks. However, the existing monitoring equipment, monitoring methods and means cannot meet the requirements, so it is urgent to strengthen the reservoir water environment monitoring capacity and improve the level of monitoring. According to this situation, the solution of building the Three Gorges Reservoir water environment monitoring system with WSN has been proposed [13].

Due to the large area and wide distribution of the Three Gorges Reservoir, which involves 26 counties (of which 22 are in Chongqing Municipality's jurisdiction) and some less accessible places, it shows a meandering tree structure [14]. Therefore, the distribution and networking method should be based on its unique serpentine tree structure. The distribution of the Three Gorges Reservoir and the structure model of the WSN water quality supervision system are shown in Figure 1.

Figure 1 shows that in such a large-scale wireless sensor network, nodes are far away from each other, and energy consumption by sending messages among nodes is very large, especially in some remote and inaccessible areas. So it is unrealistic to replenish their nodes energy frequently. Therefore, it is particularly important to improve energy efficiency and balance energy consumption of nodes in the wireless sensor networks. For these characteristics of the network, a new energy-saving routing algorithm has been proposed to solve this problem.

3. Related Researches

LEACH is the first to be proposed as a hierarchical routing algorithm, which combines the MAC protocol and the cluster-based energy efficient routing algorithm and performs well in extending the lifetime of the network. Dividing the network running into many rounds is its main idea, and in each round there are two steps: clusters establishment and stable transmission of data. During the establishment of clusters, each node generates a random value between 0 and 1. And if it is less than the threshold value $T(n)$, expressed as (1), it will be elected as the cluster head:

$$T(n) = \begin{cases} \dfrac{p}{1 - p\left(r \bmod \left(1/p\right)\right)}, & n \in G \\ 0 & n \notin G, \end{cases} \quad (1)$$

where, p is the percentage of the cluster heads occupied in the total nodes, r is the current round number, and G is the set of the nodes which have not been selected as cluster heads in the last $1/p$ rounds.

In the LEACH algorithm, the random rotation mechanism of cluster heads was used to balance the energy load to each node, so that the network lifetime has been prolonged. Compared with the normal planar multihop routing algorithm and static hierarchical algorithm, LEACH algorithm is outstanding, and the network lifetime can be prolonged by 15%. However, there are also some shortcomings, such as: (1) the remaining energy of nodes was not considered into the cluster heads selection; (2) uneven distribution of cluster heads and cluster sizes resulted from the random cluster heads selection mechanism, which causes the decline in the balance of network load; (3) when the network size is large, single-hop data transmission will lead to those cluster heads death, which are far away from the base station, so that the lifetime of the whole network will be affected.

Therefore, a lot of improved algorithms based on LEACH have been proposed to improve the network performance recently. In the LEACH-C algorithm [15], which was proposed by Heinzelman and so forth, the centralized control method was used and the base station obtained the cluster head by running a simulated annealing, so that the problem of uneven distribution of cluster heads in LEACH was solved. But it is not completely self-directed and is not good in expansibility with the centralized control method. To improve the quality of clustering, the distribution density, remaining energy of nodes, and other factors were taken into consideration in [16, 17]. But collecting information about neighbor nodes in the whole network increased the communication overhead and latency. In [18], the time interval was used to cluster instead of random number and threshold. But the distribution of the cluster head nodes has been ignored. A clustering algorithm based on energy optimization has been proposed in [19], which protected low energy nodes and reduced overhead of clustering by the competition parameters of cluster head, so that the network lifetime was prolonged. In [20], firstly, clusters were formed according to the optimization of the number of clusters, and then the data routing among clusters were completed by searching the shortest path based on the Kruskual algorithm, so that the

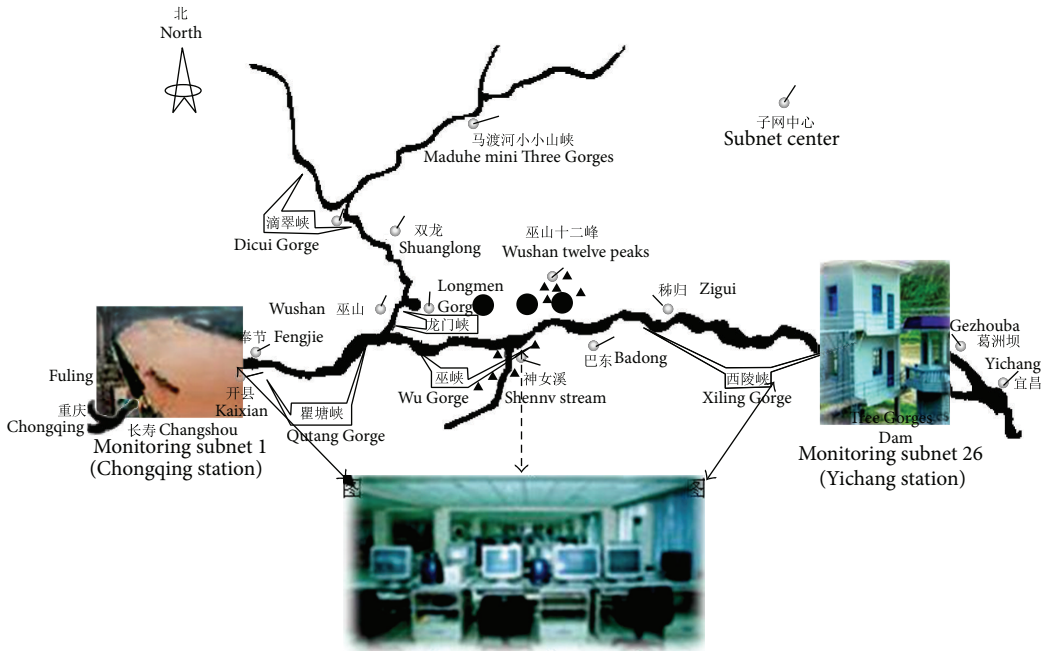


Figure 1: Distribution of the Three Gorges Reservoir and structure model of WSN water quality monitoring system.

energy efficiency of nodes has been improved. But the balance of the energy consumption of network was not mentioned.

In the social economics, the welfare reflects the equilibrium relationship between fairness and efficiency of the social crowd income [21]. In [21], a new metric for WSN is proposed, which combines energy balancing and energy efficiency, that is, EW (energy-welfare). The main idea of the PARPEW algorithm is to optimize clustering according to the predicted energy consumption of nodes and energy-welfare of clustering. So, good results such as extending network lifetime and balancing energy consumption have been achieved. Besides, it is simple to compute the energy welfare only through the local routing information. However, the shortcoming is that in the cluster heads adjustment, only the energy welfare was taken into consideration, while the effect of the remaining energy of nodes was ignored. Therefore, the situation that those nodes with low remaining energy were elected as the cluster head also existed. In addition, communication using single-hop will decrease the energy efficiency and is not conducive to network expansion.

4. The Improved Algorithm

4.1. The Network Model. Assume that the wireless network has the following properties.

(1) The position of the nodes will not be changed once deployed. The base station is located outside the area and full of energy.

(2) All nodes are energy limited and of the same structure and initial energy.

(3) All nodes have a unique ID.

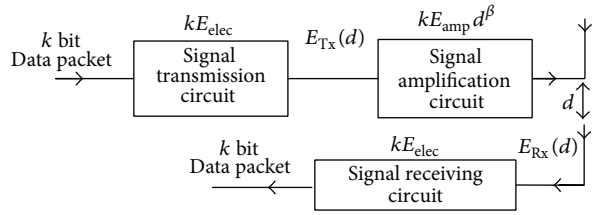


Figure 2: The first-order energy model.

(4) The transmit power can be adjusted according to the distance.

(5) All nodes can sense their remaining energy and compute the distance to the emission source according to the received signal strength.

4.2. The Energy Model. The energy model which is the same with it in [7] was used in this paper, as shown in Figure 2.

Figure 2 shows the data packet transmission between two nodes, in which the distance is d and the size of the packet is k bit. So the transmitter energy consumption is expressed as:

$$E_{\mathrm{Tx}}(k,d) = \begin{cases} kE_{\mathrm{elec}} + k\varepsilon_{fs}d^2, & d < d_0 \\ kE_{\mathrm{elec}} + k\varepsilon_{\mathrm{amp}}d^4, & d > d_0. \end{cases} \quad (2)$$

The energy consumption of the receiving terminal is shown in as:

$$E_{\mathrm{Rx}}(k) = kE_{\mathrm{elec}}. \quad (3)$$

In (2) and (3), the energy consumption of transmitting and receiving 1 bit data is denoted as E_{elec}. ε_{fs}, $\varepsilon_{\mathrm{amp}}$ are the energy

consumption coefficients of different channel propagation model. d_0 is the threshold value denoted as $d_0 = \sqrt{\varepsilon_{fs}/\varepsilon_{amp}}$, which is used to distinguish the free-space path loss model from the multipath fading model.

The energy consumption for integration of l data packets of k bit is expressed as:

$$E_{DA}(k) = l \times k \times E_{DA}, \tag{4}$$

where, E_{DA} is the energy consumption for integration of a data of 1 bit.

4.3. Algorithm Description. In the initial stages of networking, the TDMA time slot of the whole nodes was broadcasted by the base station at a time. Each node broadcasted its message to the whole nodes according to the TDMA time slot and then calculated the distance from the emission source according to the received signal strength, where the distance is denoted as $d(i, j)$, $i, j = 1, 2, \ldots, n$, and n is the number of the nodes in the network. And to provide convenience for next clustering optimization and routing, the distance was stored in the memory of the node. This step only runs once at the initial stage of networking, and the following operation model was the same with the LEACH algorithm.

4.3.1. Temporary Clustering. To reduce the energy consumption of nodes, it is better to select the nodes with more remaining energy near the base station. So, in this paper, the two factors were all taken into consideration when selecting the cluster heads. The improved threshold $T(n)$ is expressed as:

$$T(n) = \begin{cases} \dfrac{p}{1 - p\left(r \bmod (1/p)\right)} \left[E(n) + D(n)\right], & n \in G \\ 0 & n \notin G, \end{cases} \tag{5}$$

where $E(n)$ is the energy factor, denoted as $E(n) = E_{res}(n)/E_{init}$; $E_{res}(n)$, E_{init} are the remaining energy and initial energy of the node n, respectively. $D(n)$ is the distance factor, expressed as $D(n) = 1 - [d(n, BS)/d_{max}]$, $d(n, BS)$ is the distance between node n and base station, and d_{max} is the maximum of the distance from nodes to base station.

After the election of the temporary cluster head, the normal nodes sent application (including their own ID and the remaining energy of node) to those cluster heads, which costed their least communication energy. Thus, the temporary clustering was completed.

4.3.2. Cluster Heads Adjustment Based on Energy Welfare Maximization and Clustering Optimization

(1) Energy Welfare. Assume that the rth round of the cluster C has N nodes, and the remaining energy of the ith node is denoted as $E^r_{res}(i)$. According to the network model, the remaining energy after transmission of the node i can be predicted, which is expressed as $\overline{E}^r_{AT}$, shown as follows:

$$\overline{E}^r_{AT} = \frac{1}{N} \sum_{i \in N} E^r_{AT}(i). \tag{6}$$

Thus, the energy welfare of cluster C can be obtained [21], as expressed in the following equations:

$$\begin{aligned} EW^r_C(\varepsilon) &= \overline{E}^r_{AT} \times EE^r_C(\varepsilon), \\ EE^r_C(\varepsilon) &= 1 - I^r_C(\varepsilon), \\ I^r_C(\varepsilon) &= 1 - \left[\frac{1}{N} \sum_{i \in N} \left(\frac{E^r_{AT}(i)}{\overline{E}^r_{AT}} \right)^{1-\varepsilon} \right]^{1/(1-\varepsilon)}, \end{aligned} \tag{7}$$

where, $EE^r_C(\varepsilon)$ is the balance index of the energy consumption of node, and the greater the value, the more balanced the energy consumption. ε is the parameter representing the degree of hating inequality, and the greater the value of ε, the more disagreeable of inequality. So, the energy consumption of nodes with low power attracts more attention, and the typical value for ε is 1.5 [22].

The energy efficiency and the property of energy balance were both considered into the energy welfare. Therefore, increasing energy welfare can save node energy and balance the energy consumption of the network. The energy welfare instance is shown in Figure 3.

As is shown in Figure 3(a), there are three nodes in the cluster, and the energy required by the data transmission among nodes is given, and $E_{res}(i)$ is the remaining energy of the node i. To simplify the calculation, the energy consumed by cluster head for receiving data was set to be 1. The comparison between remaining energy after transmission and energy welfare of clustering of the three kinds of network models is shown in Figure 3(b). As can be seen in the figure, when the node 1 was selected as cluster head (CH), the energy efficiency is maximal, and the mean of remaining energy is 30.33 J. Considering the energy consumption balance index, when the node 3 was selected as cluster head, the energy welfare of clustering reached the maximum, and the energy efficiency and the property of energy consumption are the best. Therefore, to elect node 3 as cluster head is the most reasonable.

(2) Cluster Head Adjustment and Process of Clustering Optimization. Before cluster head adjustment, the candidate nodes for cluster head should be limited firstly. Assuming that the set of the candidate nodes for cluster head of the cluster C is Q, so

$$Q = \left\{ i \mid i \in C,\ E^r_{res}(i) > \overline{E}^r \right\}, \tag{8}$$

where, $E^r_{res}(i)$ is the remaining energy of node i in the rth round, and $\overline{E}^r$ is the mean value of the energy remaining of the nodes as follows:

$$\overline{E}^r = \frac{1}{N} \sum_{i \in N} E^r_{res}(i). \tag{9}$$

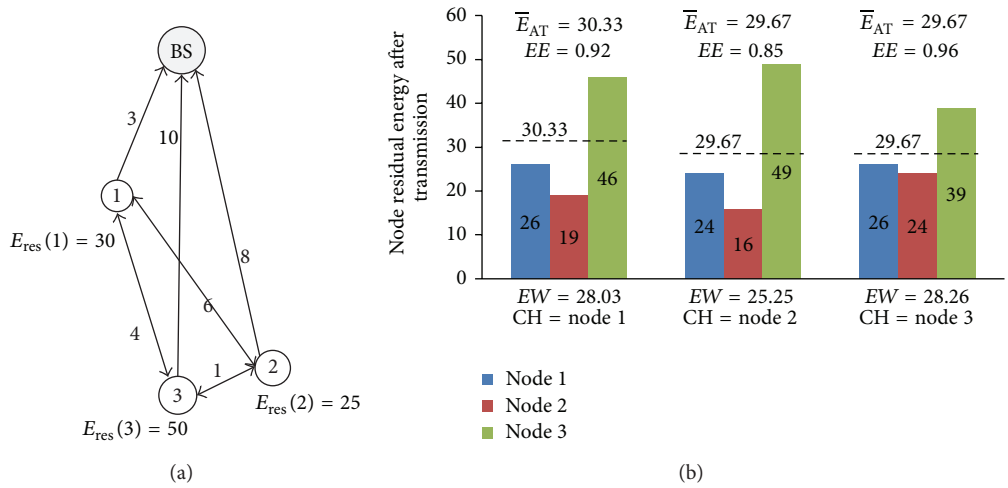


FIGURE 3: The instance of the energy welfare. (a) Energy consumption relationship between nodes. (b) Comparison between remaining energy of node and clustering welfare.

The processes of clustering optimization are shown as the following.

(1) Calculate the energy array according to the network model and the distance array $d(i, j)$, and then determine the set Q of candidate nodes for cluster head of cluster C.

(2) If the set Q is empty, the temporary cluster head will be selected as the real cluster head (RCH) directly. Otherwise, the energy welfare of cluster C is $EW_C^r|_{(\mathrm{RCH}=i)}$, $i \in Q$ according to equations (6)-(7) using each node in the set Q when the node i ($i \in Q$) is selected as RCH.

(3) Determine RCH according to the max energy-welfare of cluster C, that is $\max_{k\in Q} EW_C^r$, and the node k is RCH.

(4) After RCH is determined, the temporary cluster head broadcasts the TDMA time slot and control packets (including the ID of the temporary cluster head and RCH, and the remaining energy) of the cluster member nodes. Other nodes in the cluster become the normal nodes according to the control packets and send data to RCH according to the distributed TDMA time slot. At last the cluster head adjustment and clustering optimization are finished.

4.3.3. The Data Routing among Clusters. The transmission of data routing among nodes combines single-hop with multihop. If the distance from cluster head to base station is less than d_0, the cluster head communicates with base station directly. Otherwise, a relay will be chosen from the neighbor cluster heads to transmit data until the data reach the base station.

Firstly, the set of neighbor cluster heads of cluster C_i ($i = 1, 2 \ldots K$, K is the number of the cluster heads) is denoted as S which should meet the following condition:

$$S = \left\{ C_j \mid \alpha < \frac{d\left(C_j, \mathrm{BS}\right)}{d\left(C_i, \mathrm{BS}\right)} < 1,\ d\left(C_i, C_j\right) < d\left(C_j, \mathrm{BS}\right) \right\}, \tag{10}$$

where $d(C_i, C_j)$ is the distance from cluster head C_i to cluster head C_j, and $d(C_i, \mathrm{BS})$ is the distance from cluster head C_i to base station. α is the forward cost factor, and $0.8 < \alpha < 1$. Choosing a proper α can avoid the problem that those cluster heads of the same distance away from base station consume too much energy for transmitting data.

Then, using the greedy algorithm the cluster head C_i selects a relay which has the maximal weight function from the set S to transmit data. Considering the factors, such as, remaining energy of the node, the path costs, and the angle deviation between node and base station [23], the weight function is defined as follow:

$$W = u\frac{E_{\mathrm{res}}(j)}{E_{\mathrm{res}}(i)} + v\cos\theta + w\frac{d(C_i, \mathrm{BS})^2}{d\left(C_i, C_j\right)^2 + d\left(C_j, \mathrm{BS}\right)^2},$$
$$\cos\theta = \frac{d(C_i, \mathrm{BS})^2 + d\left(C_i, C_j\right)^2 - d(C_i, \mathrm{BS})^2}{2d\left(C_i, \mathrm{BS}\right) \times d\left(C_i, C_j\right)}, \tag{11}$$

where $E_{\mathrm{res}}(j)$ is the remaining energy of the cluster head C_j, θ is the angle deviation between node and base station, μ, ν, ω are the weighting coefficient, and $\mu + \nu + \omega = 1$. If the set S is empty, the cluster head C_i sends data to the base station directly. Otherwise, the cluster head C_i sends data to the relay. When all of the cluster heads find the relays, the data routing among clusters is completed.

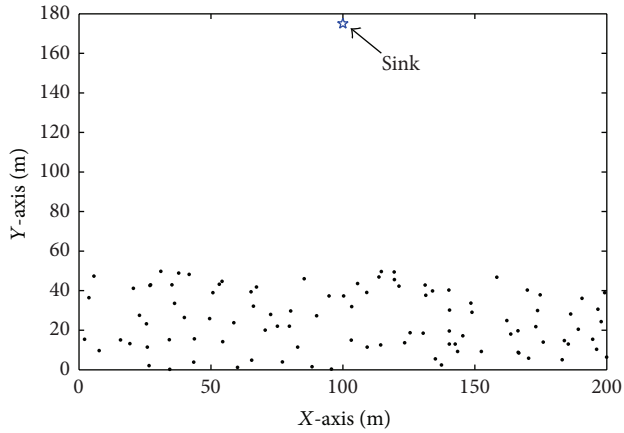


Figure 4: 100-node random network.

Table 1: The parameters of the network in simulations.

Parameters	Values
Initial energy of nodes E_{init}	0.5 J
Amplification coefficient of the free space model E_{fs}	10 pJ·m^2/b
Amplification coefficient of the multipath transmission model E_{amp}	0.0013 pJ·m^2/b
Data fusion rate E_{DA}	5 nJ/b
Circuit loss E_{elec}	50 nJ/b
Clustering probability of nodes p	0.05
Data packet length	4000 b
Control packet length	80 b

5. Simulation and Analysis

The performance evaluation of our proposed algorithm has been carried by using MATLAB simulation. We also compared the proposed algorithm with the LEACH algorithm and PARPEW algorithm. Taking into account the typical characteristics of zonal distribution of the Three Gorges Reservoir for water environment monitoring, in the simulation experiment, we used a rectangular area to approach the banding distribution, which has greater aspect ratio. Network simulation scenario is shown in Figure 4.

100 sensor nodes were deployed in a 200 m × 50 m rectangular area, and the base station was just above the area with the coordinates (100 m, 175 m). In the simulation, the forward price factor $\alpha = 0.9$, the weighting coefficients $\mu = 1/2$, $\nu = 1/6$, $\omega = 1/3$. Other simulation parameters of the network are shown in Table 1.

First, we compared the network performance among three algorithms, including the number of surviving nodes, nodes' energy consumption, and energy balance of network. Then, we analyzed the performance of our proposed algorithm in large-scale wireless sensor networks.

5.1. The Simulation of Lifecycle and Energy Consumption. Network lifetime is one of the most important criteria for evaluating the quality of routing protocol. Figure 5 shows the

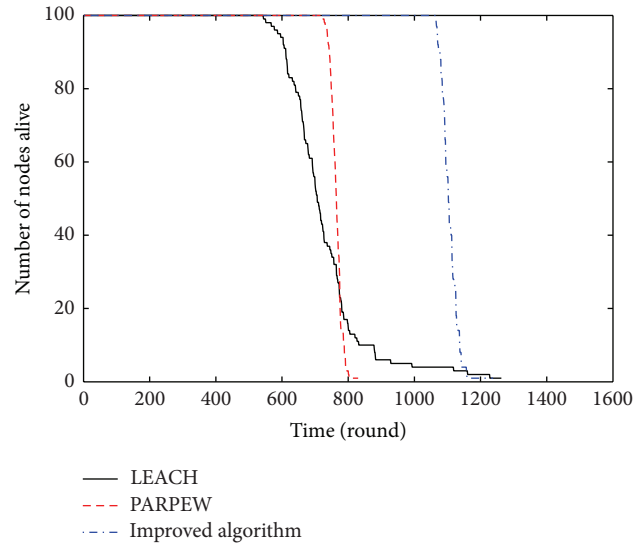


Figure 5: The relationship between the number of the survival nodes and the rounds.

Table 2: The statistical average of lifetime and energy consumption of network.

Algorithms	LEACH	PARPEW	The proposed
FND	542	713	1049
Total energy consumption at 500th round (J)	35.95	32.66	22.32

corresponding changes of the survival nodes' number during simulation. As can be seen from the figure, the network lifetime (defined as the number of rounds when the first node died) was obviously improved by the proposed algorithm in this paper with respect to LEACH and PARPEW algorithms. Since the proposed algorithm in this paper is based on the improved max energy-welfare method to adjust cluster head and optimize the clustering, the distribution of the cluster heads is more uniform and reasonable, and the energy consumption is efficiently balanced. Besides, cluster heads transmit data to the base station in single-hop and multi-hop way. It improves the energy efficiency of cluster heads and balances the energy consumption of cluster heads in different place.

The comparison curve of the total energy consumption of the network and the simulation time among the three algorithms is shown in Figure 6. The smaller slope of the curve indicates that the network has a slower rate of energy consumption and longer lifetime. In Figure 6, the slope of the curve of the proposed algorithm in this paper is minimal, and the total energy consumption of the network in any time is lower than that of the LEACH and PARPEW algorithms. So the energy consumption of the network has been reduced successfully.

To eliminate the effect of the chance factor, a number of simulations have been done to obtain the statistical average. The statistical average of lifetime and the total energy consumption at 500th round using different algorithms are shown in Table 2.

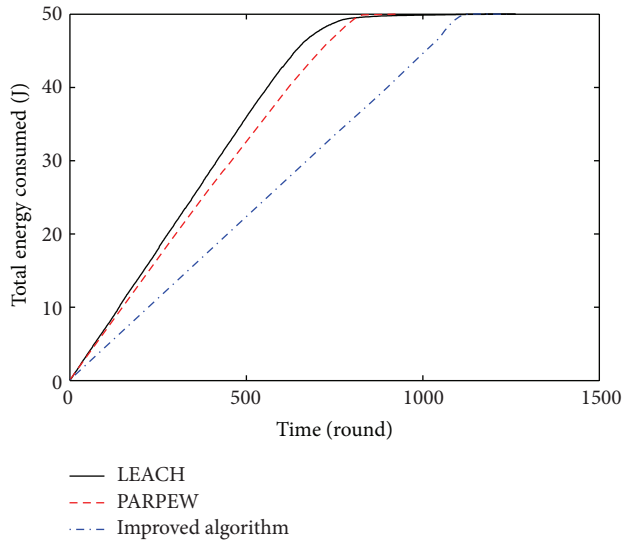


Figure 6: The changing curves of the energy consumption of network.

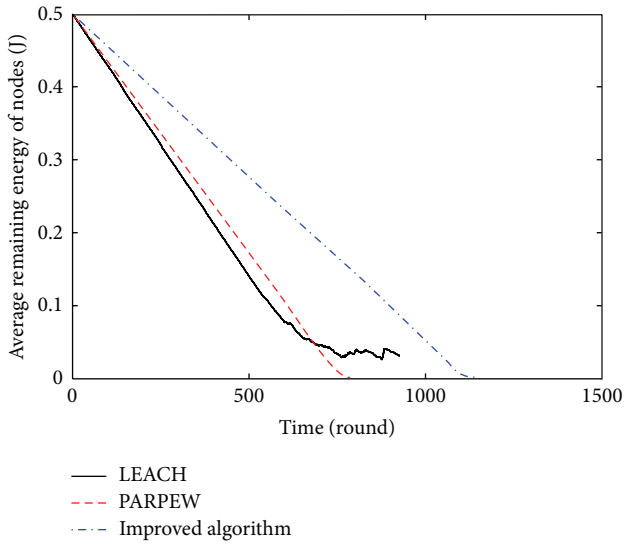


Figure 7: Comparison of the remaining energy average.

As can be seen from Table 2, compared with other two algorithms, the proposed algorithms can efficiently extend the lifetime of the network and reduce network energy consumption. The detailed comparative results are shown in Table 3.

5.2. The Simulation of Energy Balance. In this paper, the remaining energy mean and variance of nodes at a certain time were used to measure and further analyze the energy balance of the proposed algorithm. At time t, the greater the mean and the less the variance, the better the energy balance of the network. The comparison of the energy balance among the different algorithms is shown in Figures 7 and 8.

Figures 7 and 8 show that the average remaining energy of nodes in the network of the proposed algorithm in this

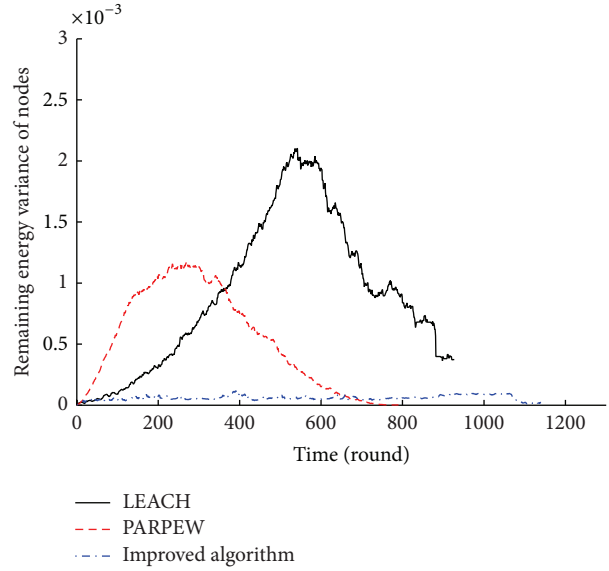


Figure 8: Comparison of the remaining energy variance.

paper has been greater than the mean energy of the other two algorithms, and the variance of the residual energy is much lower than the other two algorithms. There is no much change. Therefore, the proposed algorithm in this paper is effective in saving nodes' energy and balancing the network energy consumption. Its performance of energy balance is optimal.

5.3. The Simulation of Adaptability in Large-Scale WSN. Another important feature of the Three Gorges Reservoir water environment monitoring network is large coverage area and wide distribution. To further illustrate our proposed algorithm's good performance in large-scale wireless sensor network applications, we assumed three-network simulation environments as shown in Table 4. And we compared the network lifetime of three algorithms under the same conditions of the other parameters.

Figure 9 shows the comparison of the network lifetime of three algorithms under different environment conditions. As can be seen from the figure, in the three simulation environments, network lifetime of the proposed algorithm is superior to the other two algorithms'. And with the increasing of network size, the advantages of the proposed algorithm are more obvious in performance. This is mainly because that the algorithm uses the cluster heads optimization strategies and improved data transmission scheme to fully save and balance the energy consumption of nodes. Therefore, for the feature of large-scale water environment monitoring network application in Three Gorges Reservoir, our proposed algorithm has obvious advantages.

6. Conclusions

An improved energy saving routing algorithm based on maximum energy-welfare optimization clustering has been proposed. There are three aspects of the major improvements:

Table 3: The percentage of comparison of lifetime and energy consumption of network.

Comparison parameters	Comparison with LEACH	Comparison with PARPEW
FND (increased percentage)	93.5	47.1
Total energy consumption at 500th round (reduced percentage)	37.9	31.7

Table 4: Three kinds of simulation environments.

Simulation environment	Area (m^2)	Base station Location (m)	Number of nodes
1	100×100	(100, 175)	100
2	200×200	(100, 175)	400
3	400×400	(100, 175)	800

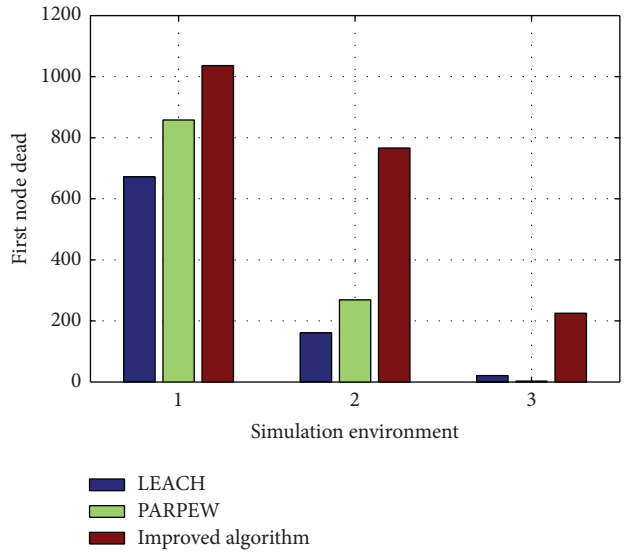


Figure 9: Impact of the simulation environment on the FND of the compared algorithms.

firstly, remaining energy and the distance from base station of nodes were both considered when selecting cluster heads. So it is more reasonable. Secondly, the cluster heads adjustment and clustering optimization based on improved maximum energy-welfare made the distribution of cluster heads more uniform, and the energy consumption of network has been balanced efficiently. Thirdly, the factors such as path costs, remaining energy, and angle deviation between node and base station were taken into consideration in the clustering routing construction, so that the energy efficiency of nodes has been increased. Simulation results show that compared with LEACH and PARPEW algorithms, the proposed algorithm in this paper can efficiently save energy of each node, balance the energy consumption, and prolong the lifetime of the network. It is suitable to the Three Gorges Reservoir water environment monitoring network with zonal structure. With the increasing size of network, this effect will be more obvious.

Conflict of Interests

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

Acknowledgments

The authors acknowledge The National Natural Science Foundation of China (no. 51374264) and The Scientific and Technological Project of Chongqing (no.CSTC2012ggyyjs40010).

References

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[2] H. Zhu and L. Sun, "Development status of wireless sensor network," *ZTE Communications*, vol. 15, no. 5, pp. 1–5, 2009.

[3] J. Hu, L. Shen, T. Song, and D. Ren, "Novel clustering algorithm for wireless sensor networks," *Journal on Communications*, vol. 29, no. 7, pp. 20–29, 2008.

[4] D. Goyal and M. R. Tripathy, "Routing protocols in wireless sensor networks: a survey," in *Proceedings of the 2nd International Conference on Advanced Computing and Communication Technologies (ACCT '12)*, pp. 474–480, Rohtak, India, January 2012.

[5] Q. Chen, Y. Zhong, and X. Zhang, "A pseudo genetic algorithm," *Neural Computing and Applications*, vol. 19, no. 1, pp. 77–83, 2010.

[6] C. Chen and C. Chuang, "A novel energy-efficient adaptive routing protocol for wireless sensor network," *Journal of the Chinese Society of Mechanical Engineers*, vol. 30, no. 1, pp. 59–65, 2009.

[7] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, pp. 3005–3014, IEEE, January 2000.

[8] S. Lindsey and C. Raghavendra, "PEGASIS: power-efficient gathering in sensor information systems," in *Proceedings of the IEEE Aerospace Conference*, vol. 3, pp. 1125–1129, 2002.

[9] S. Yi, J. Heo, Y. Cho, and J. Hong, "PEACH: power-efficient and adaptive clustering hierarchy protocol for wireless sensor networks," *Computer Communications*, vol. 30, no. 14-15, pp. 2842–2852, 2007.

[10] Z. Manap, B. M. Ali, C. K. Ng, N. K. Noordin, and A. Sali, "A review on hierarchical routing protocols for wireless sensor networks," *Wireless Personal Communications*, vol. 72, no. 2, pp. 1077–1104, 2013.

[11] P.-T. Yang and S. Lee, "A distributed reclustering hierarchy routing protocol using social welfare in wireless sensor networks,"

International Journal of Distributed Sensor Networks, vol. 2012, Article ID 681026, 14 pages, 2012.

[12] F. Lan, "Analysis for water quality variation in TGP reservoir area before and after impoundment," *Yangtze River*, vol. 39, no. 1, pp. 7–8, 2008.

[13] Y. Zhong, K. Guo, B. Xu, and J. Wang, "A hybrid network construction for water environmental monitoring of three Gorges reservoir," *World Sci-Tech R&D*, vol. 32, no. 6, pp. 737–740, 2010.

[14] F. Ji, "The general picture of the three Gorges region," *Chongqing Architecture*, vol. 9, no. 1, pp. 1–7, 2010.

[15] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[16] Y. Li, L. Sun, R. Wang, and H. Huang, "Improvement of LEACH algorithm in wireless sensor network," *Journal of Computer Research and Development*, vol. 48, no. 2, pp. 131–134, 2011.

[17] W. Luan, C. Zhu, B. Su, and C. Pei, "An improved routing algorithm on LEACH by combining node degree and residual energy for WSNs," in *Internet of Things*, vol. 312 of *Communications in Computer and Information Science*, pp. 104–109, Springer, Berlin, Germany, 2012.

[18] C. Li, X. Shen, H. Chen, and E. Sun, "Research and improvement of LEACH routing algorithm for wireless sensor networks," *Chinese Journal of Sensors and Actuators*, vol. 23, no. 8, pp. 1163–1167, 2010.

[19] T. Liu and Y. Wu, "Energy optimized approach based on clustering routing arithmetic for wireless senor networks," *Chinese Journal of Sensors and Actuators*, vol. 24, no. 5, pp. 764–770, 2011.

[20] Y. Zhong, B. Xu, K. Guo, and D. Zhou, "A partition multihop LEACH algorithm facing water environment monitoring of three Gorges reservoir area," *World Sci-Tech R&D*, vol. 2, no. 33, pp. 89–92, 2011.

[21] C. Ok, S. Lee, P. Mitra, and S. Kumara, "Distributed routing in wireless sensor networks using energy welfare metric," *Information Sciences*, vol. 180, no. 9, pp. 1656–1670, 2010.

[22] D. A. Seiver, "A note on the measurement of income inequality with interval data," *Review of Income and Wealth*, vol. 25, no. 2, pp. 229–233, 1979.

[23] T. N. Quynh, H. T. Trung, and V. Tran-Quang, "Improving energy efficiency for ARPEES routing protocol in wireless sensor networks," in *Proceedings of the International Conference onAdvanced Technologies for Communications (ATC '12)*, pp. 73–77, Hanoi, Vietnam, October 2012.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 916156, 6 pages
http://dx.doi.org/10.1155/2014/916156

Research Article

Channel Selection Based on Trust and Multiarmed Bandit in Multiuser, Multichannel Cognitive Radio Networks

Fanzi Zeng and Xinwang Shen

Key Laboratory for Embedded and Network Computing of Hunan Province, Hunan University, Changsha 410012, China

Correspondence should be addressed to Fanzi Zeng; zengfanzi@126.com

Received 18 November 2013; Accepted 19 December 2013; Published 23 February 2014

Academic Editors: H. R. Karimi, X. Yang, Z. Yu, and W. Zhang

This paper proposes a channel selection scheme for the multiuser, multichannel cognitive radio networks. This scheme formulates the channel selection as the multiarmed bandit problem, where cognitive radio users are compared to the players and channels to the arms. By simulation negotiation we can achieve the potential reward on each channel after it is selected for transmission; then the channel with the maximum accumulated rewards is formally chosen. To further improve the performance, the trust model is proposed and combined with multi-armed bandit to address the channel selection problem. Simulation results validate the proposed scheme.

1. Introduction

Radio spectrum is known to be a kind of valuable and limited resource. With the explosive growth in wireless applications, radio spectrum resources are almost exhausted. However, reliably experiments show that current spectrum is underutilized and that there exists spectrum opportunity over the space and time. To improve the utilization efficiency of spectrum, dynamic spectrum access (DSA) model based on cognitive radios [1] has been proposed, which allows secondary users (SUs) to exploit the spectrum opportunistically without interfering with primary users (PUs). In this paper, we focus on the critical issue of DSA in cognitive radio networks (CRNs): efficient channel selection strategy.

There exit many pieces of work on DSA strategy, most of which deal with spectrum sensing and channel sharing. In [1–3], the DSA strategies maximize the throughput of a single secondary user in a multichannel slotted primary network. But, there are always multiple secondary users seeking spectrum hole under licensed primary users in CRNs. So, it is very necessary and natural to address the problem of multiuser spectrum sharing in multichannel CRNs. For multi-channel CRNs, every secondary user should decide which channel must be selected to achieve highest throughput. Meanwhile, when multiple secondary users are contending for spectrum opportunities, they should take the best channel without colliding with other users. Effective channel selection for centralized and distributed system has been addressed extensively in the last decade. It is worth mentioning that for a centralized scheme, the communications between the fusion center and secondary users will raise the system reliability, but it also results in increasing the burden of the overall system, while the distributed scheme with lower complexity but no information exchange about the channel selection may lead to collisions among secondary users for some of them choosing the same channel.

In [4, 5], the authors analyzed the CRNs throughput using random sensing with negotiation, but it required an extra control channel for coordination among SUs. Hoang et al. [6] considered centralized power control and channel allocation in order to maximize the total throughput in CRNs, where each user requires the knowledge of all channel gains between PUs and SUs; in that way, it means a great deal of cooperation between PUs and SUs. The same problem has been introduced while jointly considering the result of spectrum sensing in [7]. Some papers have accrued interest in spectrum optimization for frequency selective channels (see [8, 9] and the reference therein). However, all of those pieces of work rely on central fusion controlling channel state information and statistics. In contrast, Li et al. [10] applied

analytical results for throughput for a novel slotted ALOHA-based distributed access CR system in a proposed channel selection scheme by considering the relationship between the system throughput and the number of the sensing channels. In [11], Leshem et al. suggested Gale-Shapley stable marriage theorem from game theory into channel allocation, where the user and the channel achieve the stable allocation in the cost of high complexaity. A channel-and-sensing-aware channel access (CSCA) policy is proposed for multichannel interweaves cognitive radio systems in [12], in which a minimum channel gain threshold is set on each channel to limit the probability that the channel is accessed by SUs. Therefore, we can draw a conclusion that if the SU chooses one or a subset of possible channels to access, it is crucial for DSA to design an efficient channel selection strategy to find the best channel. This problem can be formulated as a multi-armed bandit (MAB) problem (has been applied in [13, 14]). The paper [15] used the PAC-MAB formulation with optimal active sensing, which enables the SU to optimally balance energy between sensing and probing and data transmission. This motivates our present work where we transfer the channel selection problem into a MAB problem. Considering that cognitive radio systems are inherently distributed, in this paper, we consider a distributed solution to maximize the total throughput in the multiuser, multichannel CRNs.

The remainder of the paper is organized as follows. The system model and some basic definitions and assumptions used in this paper are presented in Section 2. In Section 3, we solve the problem of determining the optimal access cardinalities based on trust and MAB formulation. Section 4 is the numerical results of our algorithm. Finally, in Section 5, we conclude this paper and provide future work.

2. Problem Formulations

Suppose that there exist M secondary users (SUs), N available channels. Let $\sum_B \{b_1, b_2, \ldots b_M\}$ denote the SU set and let $\sum_S \{s_1, s_2, \ldots s_N\}$ denote the channels set. The channel selection problem is that SU chooses the suitable channel from the available channel list to transmit to maximize the total channel utility. It is assumed that each SU can select only one channel and access to different channel can obtain different gain. If $\sum_B$ is compared to the players and $\sum_S$ to the arms, the channel selection problem falls into the multi-arm bandit problem framework.

For the classical multi-armed bandit problems, the player repeatly selects an arm among a number of alternative arms and observes its associated reward to find an arm that maximizes the average cumulative reward, as close as possible to the ideal reward obtained if we were to try the "best" arm at all times.

Let $e_{i,j}$ denote channel transmission efficiency that ith SU selects jth channel to transmit,

$$e_{i,j} = t_{i,j} \cdot p_{i,j} \cdot w_j \cdot \log_2 \left(1 + \frac{\rho_{i,j} \cdot h_{i,j}}{N_0}\right), \quad (1)$$

where $t_{i,j}$ is the transmission duration of ith SU on jth channel before collision occurs; $p_{i,j}$ is the probability of the ith SU to access the channel j; w_j is the bandwidth of channel j; $\rho_{i,j} = \rho \cdot N_i \cdot d_i$ is the transmission power of the ith SU on channel j, ρ is the transmission power, N_i is the data packet size that ith SU wants to transmit, d_i represents the distance between ith SU and its receiver; $h_{i,j}$ is the transmission coefficient of channel j chosen by ith SU and N_0 is the power spectral density of noise.

Suppose each SU maintains a list it contains the channel transmission efficiency of all channels that have been selected. For $\forall s_i \in \Sigma_s$, let $H^{s_i} = (e_1^{s_i}, e_2^{s_i}, \ldots e_{n_i}^{s_i})$ denote the channel decision history, transmission efficiency of the simulated selection channel when simulated negotiation can be expressed as $\widehat{H}^{s_i} = (\hat{e}_1^{s_i}, \hat{e}_2^{s_i}, \ldots \hat{e}_{n_i}^{s_i})$, where n_i is the number of times channel s_i selected, where $e_j^{s_i}$ is the transmission efficiency gain after transmission, while $\hat{e}_j^{s_i}$ is transmission efficiency gain achieved by simulated negotiation.

3. Channel Selection Based on Multiarmed Bandit Problem

In above section, we formulate channel selection problem as MAB, where SU is compared to players and channels to arms. The player's purpose in MAB system is to maximize his total reward over a sequence of trials. Since each arm is assumed to have a different distribution of reward, the goal is to find the arm with the best expected return. The ultimate objective is to find the reward on each arm; this problem can be solved by simulated negotiation. Therefore, we can apply the algorithm of simulated negotiation into CRNs to obtain channel's reward distribution. In the condition of unknown channel's characteristics, the CR user tries to choose the best channel based on its data characteristic and decision history.

CR user chooses the negotiation target according to the probability $p_{i,j}$ of the ith SU to access the channel j, and let the channel transmission efficiency gain between the simulation negotiation and the precedent efficiency record denote the interaction reward; that is,

$$\Delta_j^{s_i} = \hat{e}_j^{s_i} - e_j^{s_i}; \quad (2)$$

then, the reward of selecting channel s_i at slot n is

$$r^{s_i}(n) = \sum_{j=1}^{n_i} \Delta_j^{s_i}. \quad (3)$$

Let $R^{s_i}(n)$ denote the total simulation negotiation reward of selecting channel s_i at slot n, which can be updated as follows:

$$R^{s_i}(n) = R^{s_i}(n-1) + r^{s_i}(n). \quad (4)$$

Typically, to solve the MAB problem, the reward on each arm is generated by some statistical assumptions for some circumstance, the distribution of each arm's reward is assumed to be a Gauss and time-invariant. But in fact, it is difficult or even impossible to determine the right statistical assumptions. In this paper, we use the nonstatistical assumptions technique to solve this classic MAB problem.

In order to achieve each arm's reward distribution, based on algorithm of Hedge(β) discussed in the literature [16] with the simulation of negotiation, apply the MAB technique into the channel selection algorithm to acquire each channel's reward distribution.

Let $\alpha > 0$ be the parameter to determine the probability of selecting the channel s_i.

The algorithm is as follows.

Initialization. for $i = 1, 2 \ldots N$, $R^{s_i}(1) = 0$

Then for $i = 1, 2 \ldots N$ do the following:

(1) According to the total simulation negotiation reward to calculate the probability of selecting channel s_i is $p_i(n)$:

$$p_i(n) = \frac{(1+\alpha)^{R^{s_i}(n)}}{\sum_{j=1}^{N}(1+\alpha)^{R^{s_j}(n)}}. \tag{5}$$

(2) Opt the channel with the maximum $p_i(n)$ as the next negotiation object; then, through simulation calculate the reward $r^{s_i}(n)$ of selecting channel s_i at slot n.

(3) Update the accumulated reward of channel s_i:

$$R^{s_i}(n) = R^{s_i}(n-1) + r^{s_i}(n). \tag{6}$$

Then choose the channel S^*:

$$S^* = \arg\max_{1 \le i \le N} \left(\sum_{j=1}^{N} p_i(j)\, r^{s_i}(j) \right). \tag{7}$$

However, this algorithm is defective; there may be a channel with a larger reward but its probability of being selected is relatively low. Therefore, we combine the probability distribution $p_i(n)$ generated by the above algorithm with a uniform distribution to form the new probability distribution $\widehat{p}_i(n)$. At the same time, reevaluate each channel's interactive reward to ensure the channel that owns greater rewards and has greater probability to be opted.

Consider the following:

$$\begin{aligned} \widehat{p}_i(n) &= (1-\gamma)\, p_i(n) + \frac{\gamma}{N}, \\ \widehat{r}^{s_i}(n) &= \left(\frac{\gamma}{N}\right) \times \frac{r^{s_i}(n)}{\widehat{p}_i(n)}. \end{aligned} \tag{8}$$

We can see that accumulated reward $\widehat{r}^{s_i}(n)$ is proportional to $r^{s_i}(n)/\widehat{p}_i(n)$, and the expected reward is also proportional to the actual reward; that is, $E[\widehat{r}^{s_i}(n)] = (\gamma/N) \times r^{s_i}(n)$; the expression of γ/N can guarantee $\widehat{r}^{s_i}(n) \in [0, 1]$. According to Lemma 4.2 introduced in the reference [17] can determine the parameters α, γ.

3.1. Channel Selection Based on Trust and Multiarmed Bandit. Through the simulated negotiation and updating the cumulative interaction reward, eventually, the user finds the best transmission channels. However, the following conditions will affect the accuracy and validity of the above algorithm: (a) before slot n, some channel may not be selected by SU which causes the negotiation history of these channels empty. Thus we cannot use (2) to calculate the interaction reward; (b) the negotiation history is too obsolete to present the current channel characteristic. Thus, in order to further improve efficiency and practicability of channel selection, in this paper, we introduce the trust model into multi-armed bandit problem to propose a channel selection algorithm.

Let $u_{i,j}$ denote channel utilization efficiency that jth SU transmit on ith channel, and $u_{i,j} = N_{i,j}/U_{i,j}$, where $N_{i,j}$ is the number of transmitted packets and $U_{i,j}$ is the channel utilization of the jth SU transmit on ith channel, which has been solved in [18].

Definition 1 (belief $B_{b\to s}$). Before the slot t, if the sensor b has finished k ($k \ge 1$) times transmission on channel s and the channel utilization efficiencies are $u_1, u_2, \ldots, u_k$, where the maximum utilization are $u_1^*, u_2^*, \ldots, u_k^*$, then the belief of the user on channel s is

$$B_{b\to s} = \frac{\sum_{i=1}^{k} u_i}{\sum_{i=1}^{k} u_i^*}. \tag{9}$$

Definition 2 (reputation $R_{b\to s}$). Before the slot t, if sensor b hasnot transmitted on channel s, we can achieve belief from other users, named sensor's reputation. It is the average of the whole sensor's belief on channel s except sensor b, which can be expressed as

$$R_{b\to s} = \sum_{i=1,b_i}^{L} r_i^b B_{b_i\to s}, \tag{10}$$

where L is the total number of sensors ever transmitted on channel s, r_i^b is the degree of trust of the sensor b on other sensor b_i, and $\sum_{i=1}^{L} r_i^b = 1$. In this paper, we assume that $r_1^b = r_2^b = \cdots r_L^b = 1/L$.

So, the trust on channel s is $T_{b\to s} = B_{b\to s} + R_{b\to s}$. Therefore, we'll combine trust-reputation model with reward distribution to solve best channel selection problem. Firstly, combine trust model with the interactive reward distribution to achieve the probability distribution of each channel; then, apply the average distribution into the channel's probability distribution; next, calculate the trust with the interactive reward and update the channel interactive reward; finally, select the appropriate negotiation target.

The outline of the algorithm is as follows.

Initialization. for $i = 1, 2 \ldots N$, $\sum_{j=1}^{N} p_i(j) r^{s_i}(j) = 0$

Then for $j = 1, 2 \ldots N$ do the following:

(1) Combine trust model with the interactive reward distribution to achieve the probability distribution of each channel.

For $j = 1, 2 \cdots N$ do the following:

$$\begin{aligned} p_j(n) &= \frac{(1+\alpha)^{R^{s_j}(n)}}{\sum_{i=1}^{N}(1+\alpha)^{R^{s_i}(n)}}, \\ \widehat{p}_j(n) &= \widehat{p}_j(n)\, B_{b_i\to s_j}. \end{aligned} \tag{11}$$

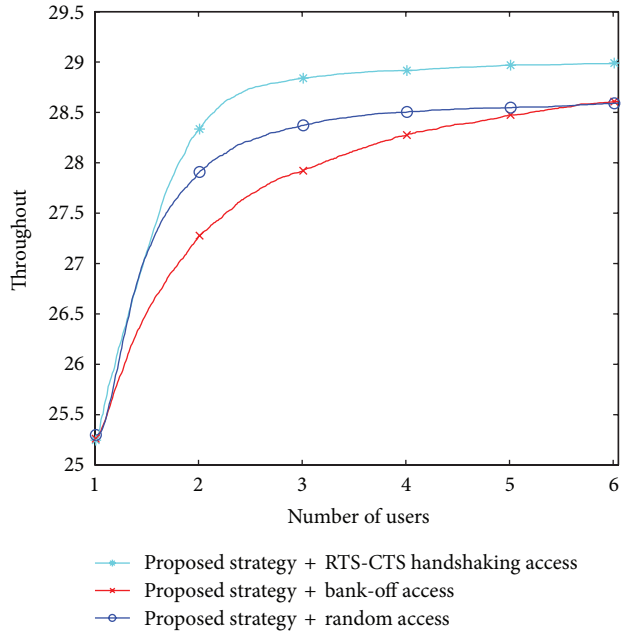


Figure 1: System throughput under different access strategy.

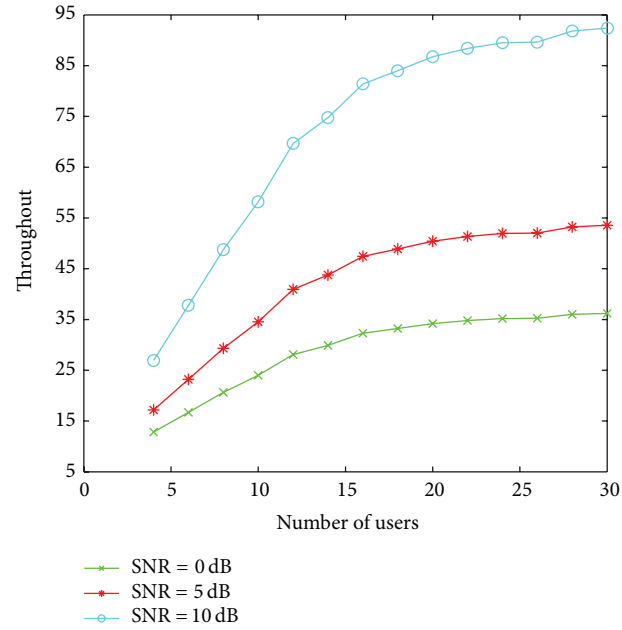


Figure 2: System throughput under different SNR condition.

(2) Apply the average distribution into the channel's probability distribution as follows:

$$\widehat{p}_j(n) = (1-\gamma)\, p_j(n) + \frac{\gamma}{N},$$
$$\widehat{r}^{s_i}(n) = \left(\frac{\gamma}{N}\right) \times \frac{r^{s_i}(n)}{\widehat{p}_j(n)}. \tag{12}$$

(3) Calculate the trust vector with the interactive reward as follows:

$$\widehat{r}^{s_j}(n) = \widehat{r}^{s_j}(n) \cdot T_{b \to s}. \tag{13}$$

(4) Update accumulative interactive reward of channel s_j as follows:

$$R^{s_j}(n) = R^{s_j}(n-1) + r^{s_j}(n). \tag{14}$$

(5) Select the best channel:

$$S^* = \arg\max_{1 \le i \le N} \left(\sum_{j=1}^{N} \widehat{p}_i(j)\, \widehat{r}^{s_i}(j) \right). \tag{15}$$

4. Simulation Results

In this section, we study the performance of the channel selection policy through simulation. 30 SUs associated with 40 channels are located within a circular ring area with radii between 200 m and 1 km, and the transmission coefficient $h_{i,j}$ is randomly selected in 0~1, but unchanged in the whole process of simulation. We run the simulation for 1000 times under the same conditions and each time our algorithm runs for 10000-time slots.

(1) When two or more than two users select a channel at the same time, it will lead to conflict. There are two methods to solve this problem: the back-off access method (all of the users quit the channel and find a new one) and random access method. In this paper, we apply RTS-CTS handshaking access; each user sends the preaccess signal before access in. The throughput curves of different access policies are shown in Figure 1. The RTS-CTS handshaking access achieves better throughput than the back-off access and the random access because it can avoid collisions by sending preaccess signal when the two users choose the same channel. Therefore, we jointly consider the proposed strategy and RTS-CTS handshaking access so as to improve the system throughput.

(2) We now turn to test the gain of the proposed strategy under different SNR conditions, as it is depicted in Figure 2. Note that for higher values of SNR the gain of the users is higher. The reason is that the good channel that is selected can be fully used for transmission.

(3) Here we compare the performance of three strategies in Figure 3: the strategy in [10], named ESS-based strategy, our proposed strategy, and the Gale-Shapley based strategy proposed in [11]. The results in the figure show the superiority of the proposed algorithm. It is noted that the system throughput first increases with the number of users, however, comes to saturation after a limited number of users. This occurs because the limited channel resource cannot bring more access opportunities; on the contrary,

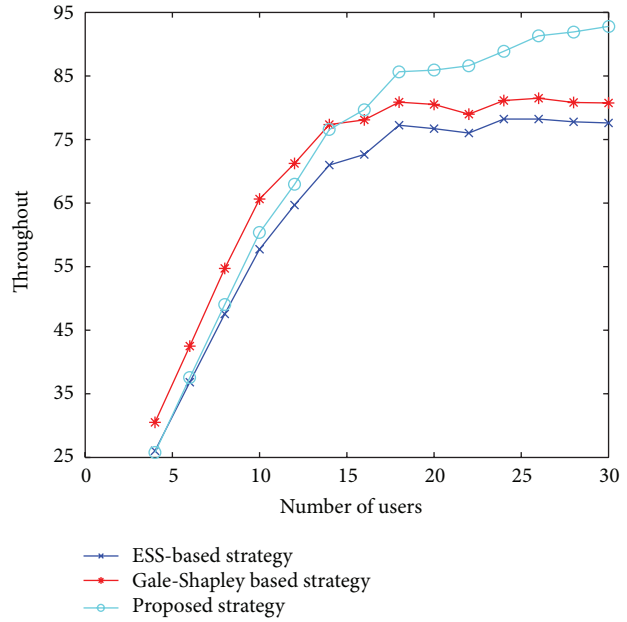


Figure 3: System throughput under different channel selection strategies.

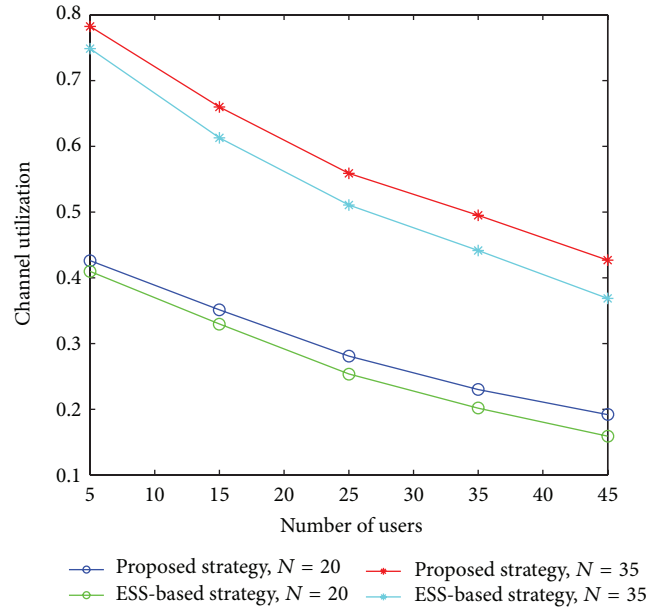


Figure 4: Comparison of channel utilization.

sometimes it may bring more collisions results in decreasing throughput.

(4) In Figure 4, we compare the channel utilization between ESS-based strategy and our proposed strategy with fixed M. We observe that the channel utilization degrades as the number of users increase. One potential cause is that along with the increase of the number of users, the conflict will increase and cause the waste of idle channel. We can also observe that our proposed scheme is more efficient than ESS-based strategy.

5. Conclusion

Multi-channel for multi-user to access in is a complicated course. In this paper, we proposed a distributed channel selection strategy based on the combination of trust-reputation model and multiarmed problem policies. Depending on the knowledge of the local observation and history decisions, a relatively efficient and channel selection strategy was obtained with the goal of maximizing the system throughput.

We provide numerical results with different scenarios regarding the system throughput; we show that higher values of SNR the gain higher throughput and the RTS-CTS handshaking access achieves better throughput than the back-off access and the random access. Furthermore, we compare our proposed channel selection algorithm with ESS-based and Gale-Shapley based strategy; simulation results show that this strategy performs better than the other two methods do in throughput gains or time overhead. As we all know, there exists a close relationship between the spectrum sensing and spectrum access; note that we did not consider this problem, which will be one of our future pieces of work.

Conflict of Interests

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

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant nos. 61370096 and 61173012 and the Key Project of Natural Science Foundation of Hunan Province under Grant no. 12JJA005.

References

[1] Q. Zhao and A. Swami, "A survey of dynamic spectrum access: signal processing and networking perspectives," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '07)*, pp. IV1349–IV1352, April 2007.

[2] Y. Yuan, P. Bahl, R. Chandra et al., "Knows: cognitive radio networks over white spaces," in *Proceedings of the 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pp. 416–427, April 2007.

[3] Q. Zhao, S. Geirhofer, L. Tong, and B. M. Sadler, "Opportunistic spectrum access via periodic channel sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 785–796, 2008.

[4] H. Su and X. Zhang, "Cross-layer based opportunistic MAC protocols for QoS provisionings over cognitive radio wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 118–129, 2008.

[5] H. Su and X. Zhang, "Design and analysis of a multi-channel cognitive MAC protocol for dynamic access spectrum networks," in *Proceedings of the IEEE Military Communications*

Conference (MILCOM '08), pp. 1–5, November 2008.

[6] A. T. Hoang, Y.-C. Liang, and M. H. Islam, "Power control and channel allocation in cognitive radio networks with primary users' cooperation," *IEEE Transactions on Mobile Computing*, vol. 9, no. 3, pp. 348–360, 2010.

[7] H. Mu and J. K. Tugnait, "Joint soft-decision cooperative spectrum sensing and power control in multiband cognitive radios," *IEEE Transactions on Signal Processing*, vol. 60, no. 10, pp. 5334–5346, 2011.

[8] A. Leshem and E. Zehavi, "Game theory and the frequency selective interference channel: a practical and theoretic point of view," *IEEE Signal Processing Magazine*, vol. 26, no. 5, pp. 28–40, 2009.

[9] E. G. Larsson, E. A. Jorswieck, J. Lindblom, and R. Mochaourab, "Game theory and the flat-fading gaussian interference channel: analyzing resource conflicts in wireless networks," *IEEE Signal Processing Magazine*, vol. 26, no. 5, pp. 18–27, 2009.

[10] X. Li, H. Liu, and S. Roy, "Throughput analysis for a multi-user, multi-channel ALOHA cognitive radio system," *IEEE Transactions on Wireless Communications*, vol. 11, no. 11, pp. 3900–3909, 2012.

[11] A. Leshem, E. Zehavi, and Y. Yaffe, "Multichannel opportunistic carrier sensing for stable channel access control in cognitive radio systems," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 1, pp. 82–95, 2012.

[12] S. Wang, C.-Y. Hsu, and Y.-W. P. Hong, "Distributed exploitation of spectrum and channel state information for channel reservation and selection in interweave cognitive radio networks," *IEEE Transactions on Wireless Communications* , vol. 12, no. 7, pp. 3458–3472, 2013.

[13] Q. Zhao, B. Krishnamachari, and K. Liu, "On myopic sensing for multi-channel opportunistic access: structure, optimality, and performance," *IEEE Transactions on Wireless Communications*, vol. 7, no. 12, pp. 5431–5440, 2008.

[14] L. Lai, H. E. Gamal, H. Jiang, and H. V. Poor, "Cognitive medium access: exploration, exploitation, and competition," *IEEE Transactions on Mobile Computing*, vol. 10, no. 2, pp. 239–253, 2011.

[15] T. V. Nguyen, H. Shin, T. Q. S. Quek, and M. Z. Win, "Sensing and probing cardinalities for active cognitive radios," *IEEE Transactions on Signal Processing*, vol. 60, no. 4, pp. 1833–1848, 2012.

[16] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[17] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "Gambling in a rigged casino: the adversarial multi-armed bandit problem," in *Proceedings of the IEEE 36th Annual Symposium on Foundations of Computer Science*, pp. 322–331, October 1995.

[18] P. Tu, X. Huang, and E. Dutkiewicz, "Subband sdaptive filtering for efficient spectrum utilization in cognitive radios," in *Proceedings of the 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom '08)*, pp. 1–4, May 2008.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 975054, 10 pages
http://dx.doi.org/10.1155/2014/975054

Research Article

A New Cooperative MIMO Scheme Based on SM for Energy-Efficiency Improvement in Wireless Sensor Network

Yuyang Peng[1] and Jaeho Choi[1,2]

[1] *Department of Electronic Engineering, Chonbuk National University, Jeonju 561-756, Republic of Korea*
[2] *CAIIT, Chonbuk National University, Jeonju 561-756, Republic of Korea*

Correspondence should be addressed to Yuyang Peng; yuyang@jbnu.ac.kr

Received 29 November 2013; Accepted 19 December 2013; Published 20 February 2014

Academic Editors: Z. Yu and W. Zhang

Improving the energy efficiency in wireless sensor networks (WSN) has attracted considerable attention nowadays. The multiple-input multiple-output (MIMO) technique has been proved as a good candidate for improving the energy efficiency, but it may not be feasible in WSN which is due to the size limitation of the sensor node. As a solution, the cooperative multiple-input multiple-output (CMIMO) technique overcomes this constraint and shows a dramatically good performance. In this paper, a new CMIMO scheme based on the spatial modulation (SM) technique named CMIMO-SM is proposed for energy-efficiency improvement. We first establish the system model of CMIMO-SM. Based on this model, the transmission approach is introduced graphically. In order to evaluate the performance of the proposed scheme, a detailed analysis in terms of energy consumption per bit of the proposed scheme compared with the conventional CMIMO is presented. Later, under the guide of this new scheme we extend our proposed CMIMO-SM to a multihop clustered WSN for further achieving energy efficiency by finding an optimal hop-length. Equidistant hop as the traditional scheme will be compared in this paper. Results from the simulations and numerical experiments indicate that by the use of the proposed scheme, significant savings in terms of total energy consumption can be achieved. Combining the proposed scheme with monitoring sensor node will provide a good performance in arbitrary deployed WSN such as forest fire detection system.

1. Introduction

Wireless sensor network (WSN) has been considered as one of the key techniques in many applications such as environment monitoring, industrial control, and so forth [1]. In a typical WSN, the power supply of each sensor node is from its battery which is energy limited and difficult to charge. Under this situation, minimizing the energy consumption to improve the energy efficiency becomes an alternative way to solve the energy constraint problem. Multiple-input multiple-output (MIMO) has been proved as a core technique to reduce the energy consumption of a wireless network [2]. However, in WSN, wireless sensor node is usually designed by using single transceiver antenna to realize a single-input single-output (SISO) transmission mechanism since the sensor node may not be able to equip using multiple antennas due to the small physical size. Consequently, it is difficult to directly apply MIMO scheme in WSN. Fortunately, the emergence of cooperative MIMO (CMIMO) [3, 4] brings the solution to solve this transmission problem. CMIMO, sometimes referred to as virtual MIMO, can achieve MIMO gains by use of collaboration among the single antennas embedded in each single node. Whereas, in order to realize collaboration, additional circuit and the local data exchange are required which result in the extra energy consumption. Therefore, in order to evaluate the energy consumption performance of CMIMO system, such extra energy also needs to be considered as well as the transmission energy.

Recently, a technique motivated by improving the spectral efficiency named spatial modulation (SM) [5] is proposed. As a working principle, it conveys the incoming information via the spatial antenna index and the MQAM/M-ray phase shift keying symbol in the signal constellation diagram and transmits the modulated symbol through a wireless channel by using a corresponding antenna specified by the antenna index. During the transmission, only one antenna is active

while others are sleeping which can effectively avoid the interchannel interference (ICI).

In this paper, in order to take full advantage of these approaches, a new scheme named CMIMO-SM is proposed which involves the joint utilization of cooperative MIMO and SM techniques in WSN for energy-efficiency improvement. We first model the energy consumption of the CMIMO-SM communication and compare the results with the conventional CMIMO. The energy consumption is compared over different transmission distances under the requirement of the same throughput and bit error rate (BER). Three system configurations are considered in both CMIMO-SM and CMIMO system for validating the performance of the proposed scheme, and then the proposed CMIMO-SM scheme is extended to a multihop clustered scenario in which the optimal hop length is derived mathematically by considering the transmitted load. Results from numerical experiments indicate that by use of the proposed scheme, significant performance in terms of energy consumption can be achieved. Also it is demonstrated that the total energy balancing performance of the proposed scheme can be improved.

The remainder of the paper is organized as follows. In Section 2, the related works will be introduced in terms of performance of different CMIMO. In Section 3, the necessary background information including CMIMO and SM are introduced. In Section 4, the proposed scheme and energy model are given. The performance of the proposed scheme and extended one is presented using simulations and numerical experiments in Section 5. Finally, in Section 6 the paper is concluded with a brief summary.

2. Related Work

In [3], for the first time, a CMIMO concept was proposed by Cui et al. for single hop transmission in WSN. It was shown that CMIMO can achieve real MIMO advantages in terms of energy efficient performance if the transmission distance is longer than the critical distance. A vertical Bell labs layered space-time (V-BLAST) based virtual MIMO is proposed in [4], which considers the training overhead requirement. The authors of [6] discussed the efficiency of cooperative transmission under space-time block code-encode (STBC) and the synchronization requirements. In [7], the authors have shown that the number of cooperative nodes at transmission and reception sides is supposed to be selected in order to reduce the energy consumption. The effect of cooperative transmitting area inside the cluster was discussed in [8]. CMIMO in a clustered WSN for energy efficiency was presented in [9]. The routing design to gain the advantages of CMIMO in energy saving is proposed in [10, 11]. In [12], the authors optimized energy consumption per unit transmit distance by selecting the number of cooperative nodes and the transmit energy consumption. The optimization of the cooperative transmission by using single parameter selection of cooperative nodes is introduced in [13, 14]. In [15, 16], the CMIMO with data aggregation technique for energy efficient WSN was presented. In WSN, information collected by the sensors needs to be transmitted to the sink or destination. If the destination is far away, the transmission requires multihop based technique. Finding the optimal hop to benefit CMIMO for achieving energy efficiency was investigated in [17, 18] and multihop hybrid virtual MIMO scheme for WSN was designed in [19]. However, none of these techniques take into account SM which is able to avoid ICI and improve spectral efficiency. To the best of our knowledge, we make the first attempt at changing the transmission way of CMIMO with SM based technique for WSN.

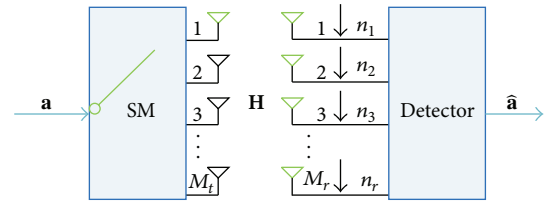


Figure 1: The model of spatial modulation.

3. Background

3.1. Cooperative MIMO. The basic idea of CMIMO started with a so called virtual antenna array (VAA) [20], a cooperative scheme wherein one or more antennas were embedded in one sensor node and several sensor nodes cooperate to emulate VAA system to achieve MIMO gains. Compared with real MIMO, CMIMO is more applicable in WSN. It allows small sensor nodes to achieve MIMO function without increasing their physical size. STBC as another important technique in MIMO field makes CMIMO perform well using the diversity approach. Consequently, CMIMO becomes a good candidate to figure out the energy problem in WSN, especially the WSN with energy constraint.

3.2. Spatial Modulation. Spatial modulation is a practical way for transmitting information by using amplitude/phase modulation and antenna index techniques. This 3 dimensional signal expression approach is able to make SM achieve high spectral efficiency. Furthermore, due to the fact that only one antenna is active during the transmission, the ICI is effectively avoided. The basic working principle is explained using Figure 1. A sequence of bits **a** gets into the SM system which consists of M_t transmitters and M_r receivers. The number of bits that can be transmitted using SM is $\log_2(MM_t)$, where $M = 2^b$ and b is constellation size. We can write this incoming bits **a** using a vector $\mathbf{x} = [x_1 x_2 \dots x_{Mt}]^T$, where a power constraint of unity is assumed. According to the working principle of SM, only one antenna is active during the transmission which results in a single nonzero element in **x**. The signal is transmitted from the corresponding antenna over the $M_r \times M_t$ MIMO channel **H** whose elements are assumed to be independent identically distributed (i.i.d.) complex Gaussian random variables with zero mean and unit variance, and M_r dimensional noise vector **n** are assumed to be i.i.d. complex Gaussian random variables with zero mean and variance of N_0. Then the received signal can

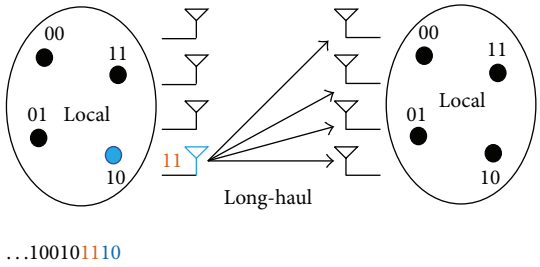


Figure 2: The structure of CMIMO-SM.

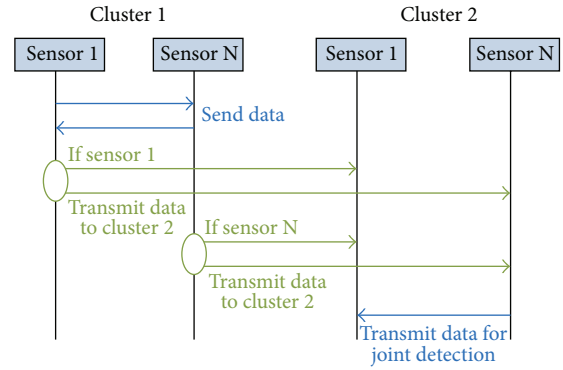


Figure 3: The communication process of CMIMO-SM.

be written as $\mathbf{y} = \mathbf{Hx} + \mathbf{n}$ and the estimated symbol can be extracted by using optimal maximum likelihood (ML) detector. Mathematically, the estimated symbol can be expressed by $\hat{\mathbf{x}} = \arg\min_{X\in\Lambda}\|\mathbf{y} - \mathbf{Hx}\|_F^2$, where $\hat{\mathbf{x}}$ indicates the estimated symbol vector, Λ is the set of all possible transmit symbols, and $\|\cdot\|_F$ denotes the Frobenius norm of the vector.

4. System Modeling

4.1. System Model. Let us consider a wireless sensor network in which each sensor node is equipped with a single antenna for data transmission. Typically, local sensor nodes inside the cluster collect the information and transmit it to the next relay cluster if the destination is far away. As we discussed in the last section, CMIMO can make single node with single antenna achieve MIMO performance and SM can provide high spectral efficiency and avoid ICI. Thus, if we allow CMIMO and SM techniques to be used among multiple nodes, the joint performance will be obtained. Such scenario is shown in Figure 2.

In our proposed model, each sensor node inside the cluster has a preassigned index using binary numbers to represent. In order to realize the MIMO function for saving energy, the local data exchange is necessary. The data flow inside the cluster is defined as local transmission while the data delivering between two clusters is defined as long-haul transmission. Suppose that M_t nodes on the transmitting side while M_r nodes on the receiving side. On the transmitting side, each sensor node broadcasts its information to all the other nodes inside the cluster using different time slots as the first stage. Once each node receives all the other information bits, the data sequence is ready to be transmitted through the MIMO channel. For each time instant, the data sequence is composed by the MQAM/MPSK modulated symbol part and antenna represented part. Only the MQAM/MPSK modulated symbols are transmitted and the symbol represented by corresponding antenna as the hidden information will be detected at receiver. On the receiver side, one destination node and $M_r - 1$ nodes join the cooperative reception. The whole process of CMIMO-SM can be explained by using Figure 3.

Example: consider a CMIMO-SM based WSN with 4 sensor nodes on the transmitter and each sensor node has an index which is similar to the instance in Figure 2. It is assumed that 1110 is the data sequence to transmit after local data flow and modulation is 4QAM. According to our proposed approach, only bits 11 will be modulated using 4QAM and transmitted via the corresponding antenna 10 while the bits 10 as the antenna index will be detected at the receiver.

4.2. Energy Consumption Model. In order to evaluate the energy performance of the proposed scheme, the energy model is first discussed. From [3] it is known that the total average power consumption of a normal communication system can be categorized into two main components: the power consumption of all the power amplifiers P_{PA} and the power consumption of all the circuit blocks P_c. P_{PA} depends on the output transmission power P_{out} and has a linear relation:

$$P_{\text{PA}} = (1 + \alpha)\, P_{\text{out}}, \tag{1}$$

where α equals to $\xi/\eta - 1$ with ξ being the peak to average ratio (PAR) and η being the drain efficiency of the RF power amplifiers. For MQAM, $\xi = 3(M^{1/2}-1)/(M^{1/2}+1)$. P_{out} can be calculated as below if the channel is a square-law path loss [21]:

$$P_{\text{out}} = \overline{E}_b R_b \times \frac{(4\pi d)^2}{G_t G_r \lambda^2} M_l N_f, \tag{2}$$

where $\overline{E}_b$ is the average energy per bit required for a given bit error rate (BER) at receiver, d is the transmission distance, G_t and G_r are the transmitter and receiver antenna gains, respectively, λ is the carrier wavelength, M_l is the link margin compensating the hardware process variations and other additive interference or background noise, and N_f is the receiver noise figure. It should be noted that N_f is given by $N_f = N_r/N_0$, where N_r is the power spectral density (PSD) of the total effective noise at the receiver input and N_0 is the single-sided thermal noise PSD at room temperature with a value $N_0 = -171$ dBm/Hz. In (2), depending on the number of transmit and receive antennas, $\overline{E}_b$ can be calculated using the SNR value and the PSD of the thermal noise N_0 for the BER requirement.

The total circuit power consumption for an M_t transmit and M_r receive antennas system is given by

$$p_c \approx M_t \left(P_{\mathrm{DAC}} + P_{\mathrm{mix}} + P_{\mathrm{filt}}\right) + 2P_{\mathrm{syn}} + M_r \left(P_{\mathrm{LNA}} + P_{\mathrm{mix}} + P_{\mathrm{IFA}} + P_{\mathrm{filr}} + P_{\mathrm{ADC}}\right), \tag{3}$$

where P_{DAC}, P_{mix}, P_{LNA}, P_{IFA}, P_{filt}, P_{filr}, P_{ADC}, and P_{syn} are the power consumption values for the DAC, the mixer, the low noise amplifier (LNA), the intermediate frequency amplifier (IFA), the active filters at the transmitter side, the active filters at receiver side, the ADC, and the frequency synthesizer, respectively. The values of P_{DAC}, P_{ADC}, and P_{IFA} can be calculated using the mode introduced in [22].

The total energy consumption per bit can be expressed as

$$E_{\mathrm{bt}} = \frac{P_{\mathrm{PA}} + P_c}{R_b}. \tag{4}$$

According to (1) and (2), the total energy consumption per bit can be rewritten as

$$E_{\mathrm{bt}} = (1+\alpha)\,\overline{E}_b \times \frac{(4\pi d)^2}{G_t G_r \lambda^2} M_l N_f + \frac{P_c}{R_b}. \tag{5}$$

According to the transmission process described in Section 4.1, the energy consumption per bit of the proposed scheme $E_{\mathrm{bt,csm}}$ consists of two components: energy consumption in the local phase E_l and energy consumption in the long-haul phase E_{lh}, that is,

$$E_{\mathrm{bt,csm}} = E_l + E_{\mathrm{lh}}. \tag{6}$$

For the local energy consumption, there are two communication phases: (1) energy consumption of data exchange inside cluster in transmitter side E_i^t; (2) energy consumption of data collection for joint detection inside cluster in receiver side E_j^r. Note that for the local phase communication, cooperative transmission is not used and $M_t = M_r = 1$, namely SISO communication. For the long-haul energy consumption, CMIMO-SM is utilized and note that the transmit antenna number M_t is always equal to one in the circuit power consumption, since only one antenna is active during each time instant. Assume that each sensor node has N_i bits to transmit; and then the energy consumption per bit for E_l is given by

$$E_l = \frac{\sum_{i=1}^{M_t} N_i E_i^t + \sum_{j=1}^{M_r-1} E_j^r \sum_{i=1}^{M_t} N_i}{\sum_{i=1}^{M_t} N_i}. \tag{7}$$

After adding E_{lh} to E_l, the energy consumption per bit can be expressed as

$$E_{\mathrm{bt,csm}} = \frac{\sum_{i=1}^{M_t} N_i E_i^t + E_{\mathrm{lh}} \sum_{i=1}^{M_t} N_i + \sum_{j=1}^{M_r-1} E_j^r \sum_{i=1}^{M_t} N_i}{\sum_{i=1}^{M_t} N_i}, \tag{8}$$

where E_i^t, E_j^r, and E_{lh} can be calculated according to (5).

In this paper, for a realistic case we take into account the detection energy as well. The detection energy can be calculated by using operation complexity in terms of complex multiplication and addition. Processing ability of the multiplications and additions depends on the CPU speed and can be different for different devices. We refer to the TelosB mote [23] as an instance. It requires 8 cycles and 4 cycles for dealing with multiplication and addition, respectively, and E_J Joule per cycle [24]. When SM applies ML at the receiver, $2M_tM_r + (M_t+1)M$ complex multiplications and $M_t(M_r-1)$ complex additions are needed for one symbol detection [25]. So the energy consumption per bit for detection can be calculated as

$$E_d = \frac{E_J \times \left\{8\left[2M_tM_r + \left(M_t+1\right)M\right] + 4\left[M_t\left(M_r-1\right)\right]\right\}}{\log_2\left(M_tM\right)}. \tag{9}$$

And then the energy consumption per bit for CMIMO-SM approach is given by

$$E_{\mathrm{bt,csm}} = \frac{\sum_{i=1}^{M_t} N_i E_i^t + E_{\mathrm{lh}} \sum_{i=1}^{M_t} N_i + \sum_{j=1}^{M_r-1} E_j^r \sum_{i=1}^{M_t} N_i}{\sum_{i=1}^{M_t} N_i} + E_d. \tag{10}$$

For energy consumption evaluation and comparison, the reference CMIMO introduced in [3] is used in this paper. For the case of CMIMO, energy consumption per bit in the local phase is the same as that in the proposed scheme. For the long-haul phase, Alamouti approach [26] is utilized and when Alamouti applies ML at the receiver, $0.5M_r$ complex multiplications and $2M_r - 1$ complex additions are needed for one symbol detection [5]. The same calculating way as CMIMO-SM, the energy consumption per bit for detection is given as

$$E_{\mathrm{dc}} = \frac{E_J \times \left[8\left(0.5M_r\right) + 4\left(2M_r - 1\right)\right]}{\log_2 M}. \tag{11}$$

The total energy consumption per bit for CMIMO is given as

$$E_{\mathrm{bt,c}} = \frac{\sum_{i=1}^{M_t} N_i E_i^{\mathrm{tc}} + E_{\mathrm{lhc}} \sum_{i=1}^{M_t} N_i + \sum_{j=1}^{M_r-1} E_j^{\mathrm{rc}} \sum_{i=1}^{M_t} N_i}{\sum_{i=1}^{M_t} N_i} + E_{\mathrm{dc}}, \tag{12}$$

where $E_i^{\mathrm{tc}}$and E_j^{rc} represent the local transmission energy cost per bit for cooperation on the transmitter side and joint detection on the receiver side, respectively. The energy consumption per bit in long-haul transmission by using the Alamouti approach is denoted as E_{lhc}. E_i^{tc}, E_j^{rc}, and E_{lhc} can be calculated according to (5).

5. Simulations and Numerical Experiments

5.1. For Long-Haul Communication. In our simulation, Monte Carlo simulations are carried out to find $\overline{E}_b$. Specifically, ten thousand randomly generated channel samples are taken and averaged to find the desired BER and then invert to get the required value of $\overline{E}_b$. Assume that the rate of 2 bits/s/Hz, 3 bits/s/Hz, and 4 bits/s/Hz can be supported. Figures 4 and 5

Table 1: Modulation configurations for 2 bits/s/Hz, 3 bits/s/Hz, and 4 bits/s/Hz transmissions using CMIMO-SM and CMIMO.

	2 bits/s/Hz	3 bits/s/Hz	4 bits/s/Hz
CMIMO-SM	BPSK	4QAM	8QAM
CMIMO	4QAM	8QAM	16QAM

Table 2: α configurations for 2 bits/s/Hz, 3 bits/s/Hz, and 4 bits/s/Hz transmissions using CMIMO-SM and CMIMO.

	2 bits/s/Hz	3 bits/s/Hz	4 bits/s/Hz
CMIMO-SM	0.4706	1.8571	3.0937
CMIMO	1.8571	3.0937	4.1429

Table 3: Circuit power consumption P_c for 2 bits/s/Hz, 3 bits/s/Hz, and 4 bits/s/Hz transmissions using CMIMO-SM and CMIMO.

	2 bits/s/Hz	3 bits/s/Hz	4 bits/s/Hz
CMIMO-SM		0.2732 W	
CMIMO		0.3214 W	

Table 4: α, and P_c values for local communication using CMIMO-SM and CMIMO.

	α	P_c
CMIMO-SM/CMIMO	0.4706	0.2107 W

represent the BER performance under different SNR values of SM and Alamouti, respectively.

$\overline{E}_b$ is calculated using the SNR value plotted in Figures 4 and 5 and the PSD of the thermal noise N_0 for the BER requirement 10^{-3}. The configurations of modulation and α of CMIMO-SM and CMIMO in long-haul stage are tabulated in Tables 1 and 2, respectively. Total circuit power consumption P_c is calculated using (3) and listed in Table 3.

5.2. For Local Communication. In the local phase, both CMIMO-SM and CMIMO utilize BPSK as the modulation scheme and SISO as transmission approach. The same approach introduced in Section 5.1 is used to simulate and calculate the parameters which are listed in Table 4.

In this simulation, the following values have been used: $B = 10$ kHz, $f_c = 2.5$ GHz, $P_{\text{mix}} = 30.3$ mW, $P_{\text{filt}} = 2.5$ mW, $P_{\text{filr}} = 2.5$ mW, $P_{\text{LNA}} = 20$ mW, $P_{\text{synth}} = 50$ mW, $M_l = 40$ dB, $N_f = 10$ dB, $G_tG_r = 5$ dBi, and $\eta = 0.35$, $E_J = 1.215$ nJ, $N_i = 20$ kb. For the fair comparison, we give the same setup $M_t = M_r = 2$ as in [3].

5.3. Energy Consumption Comparison. Since the analytical and simulation results are too complicated to offer obvious information, it is expected that some comparisons can be given from our results. It is assumed that the average distance between two adjacent nodes inside the cluster is about 2 meters. The energy consumption per bit comparisons for

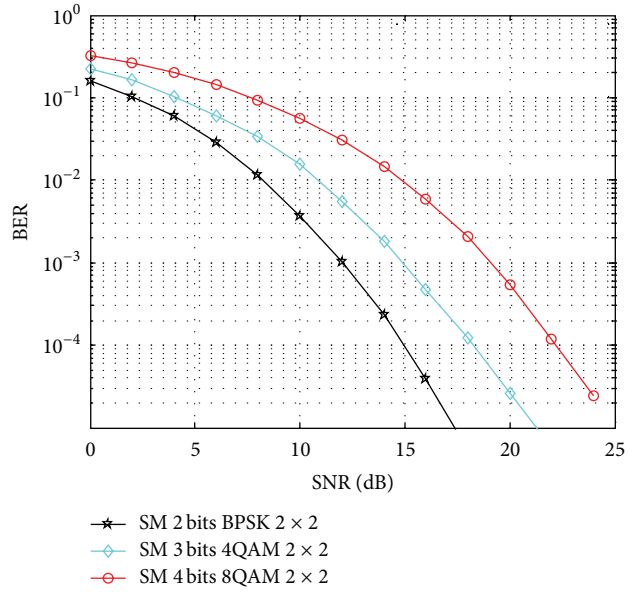


Figure 4: The received SNR versus BER in SM with 3 different modulations.

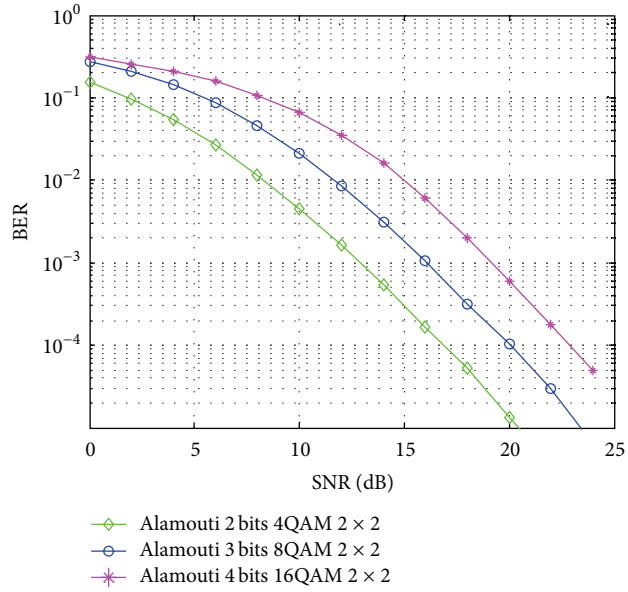


Figure 5: The received SNR versus BER in Alamouti with 3 different modulations.

2 bits, 3 bits, and 4 bits systems between CMIMO-SM and CMIMO are presented in Figures 6, 7, and 8, respectively.

From above plots, we see that the proposed CMIMO-SM beats CMIMO due to the advanced transmission scheme. Also, for both systems, as the transmission distance increases, energy consumption per bit increases. Moreover, for both systems, the energy consumption per bit decreases as the transmission rate increases from 2 bits/s/Hz to 4 bits/s/Hz. This can be explained by the reason that circuit power working in a shorter time will bring lower energy.

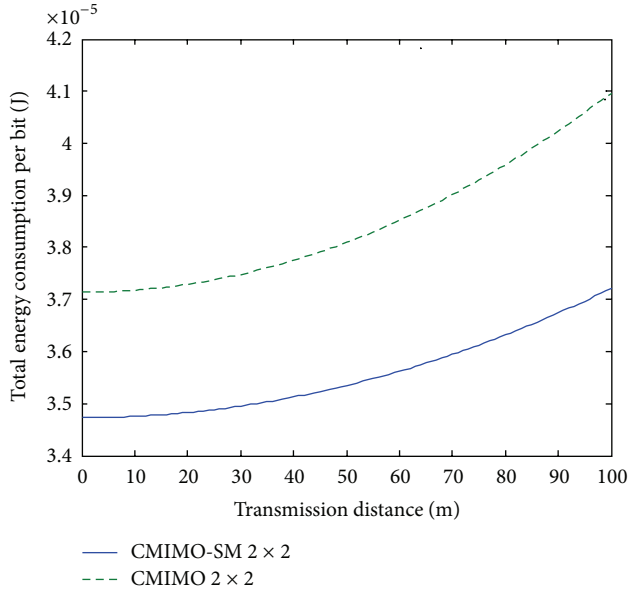


Figure 6: Total energy consumption per bit over d, 2 bits transmission.

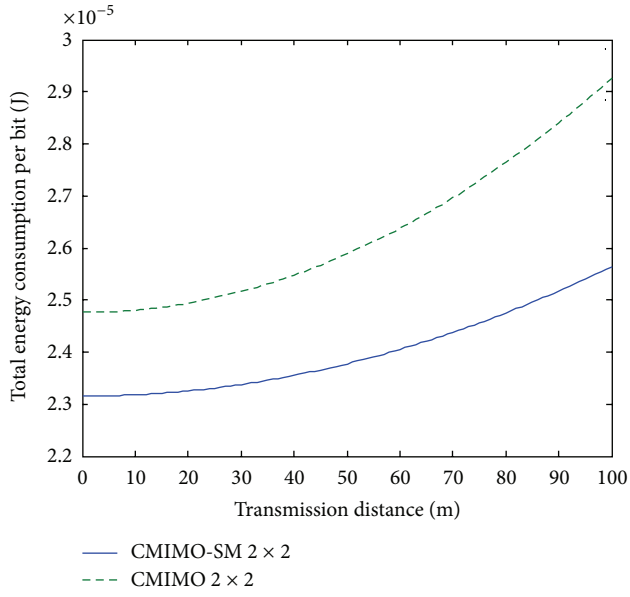


Figure 7: Total energy consumption per bit over d, 3 bits transmission.

5.4. Energy Consumption in Multihop Scenario. Now a multihop WSN composed of n clusters and a destination is considered and shown in Figure 9. The proposed CMIMO-SM scheme is applied to this multihop WSN for saving energy. Inside the cluster, the longest distance amongst the nodes is defined as d_{local}. The long-haul distance between the nearest nodes of different clusters is defined as d_i ($i = 1, 2, \ldots n$) which is assumed to be much larger than d_{local}. Let d_i represent the optimal transmission distance and then, for

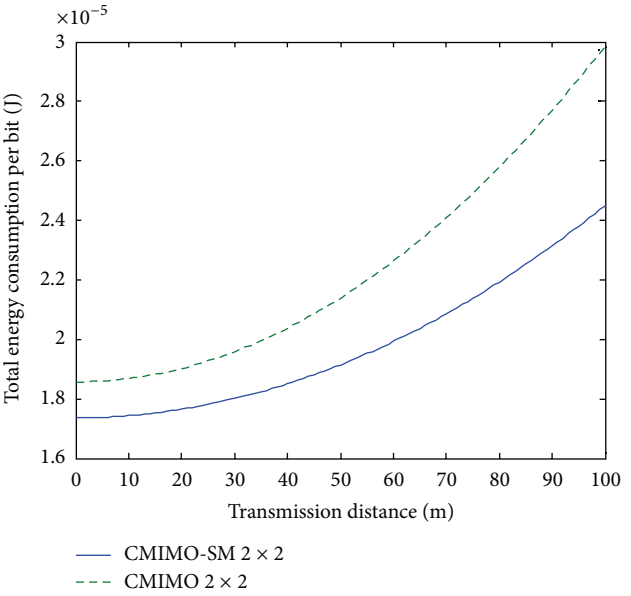


Figure 8: Total energy consumption per bit over d, 4 bits transmission.

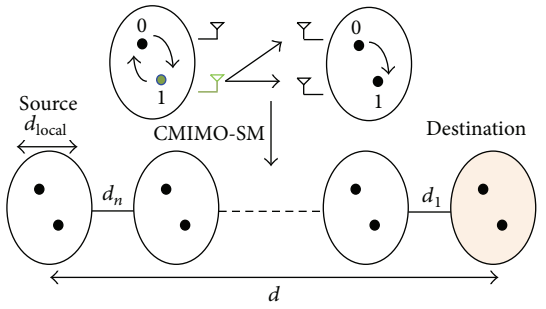


Figure 9: The structure of multihop CMIMO-SM.

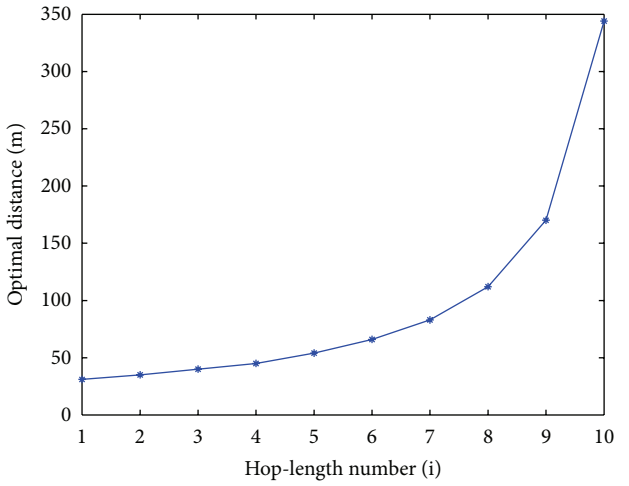


Figure 10: Optimal distances versus hop-length numbers.

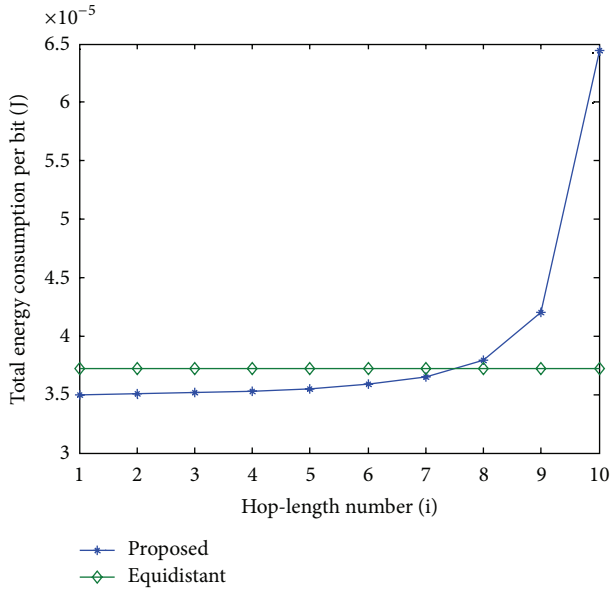


Figure 11: Total energy consumption per bit for each cluster, 2 bits transmission.

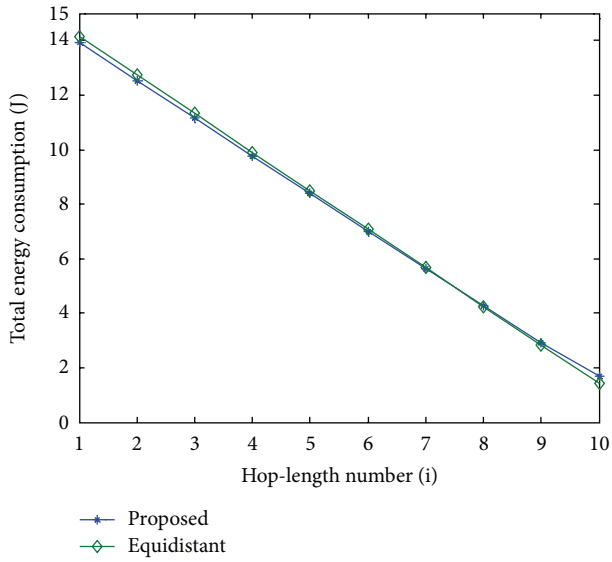


Figure 12: Total energy consumption for each cluster, 2 bits transmission.

a transmission distance d_i, energy consumption per bit of long-haul is given by

$$E_{\text{bt}}^{(i)} = (1+\alpha)\,\overline{E}_b \times \frac{(4\pi d_i)^2}{G_t G_r \lambda^2} M_l N_f + \frac{P_c}{R_b}. \tag{13}$$

It will be assumed that two sensor nodes in one cluster are used in order to reduce the complexity of the calculation. However, it is important to note that the calculation can in principle be extended to any cluster size. For all nodes transmitting scenario, the total energy consumption can be expressed as

$$\begin{aligned} E_{\text{total}} = {} & \sum_{i=1}^{n} \left(\sum_{i=1}^{M_t} N_i E_i^t + \sum_{j=1}^{M_r-1} E_j^r \sum_{i=1}^{M_t} N_i + \sum_{i=1}^{M_t} N_i E_d \right) \\ & + 2\sum_{i=1}^{n} (n+1-i) \\ & \times \left[(1+\alpha)\,\overline{E}_b \times \frac{(4\pi)^2 \times (d_i + d_{\text{local}})^2}{G_t G_r \lambda^2} M_l N_f + \frac{P_c}{R_b} \right] N_i. \end{aligned} \tag{14}$$

In order to minimize the total energy consumption, the optimal transmission distance d_i from the source to the destination needs to be derived. First, through observation of the transmission distance in the proposed model, the constraint $\sum_{i=1}^{n} d_i = d - nd_{\text{local}}$ is obtained; and then the optimization problem can be formulated as

$$\begin{aligned} \min_{d_i,\, i=1,2,\ldots n} \quad & 2\sum_{i=1}^{n} (n+1-i) \\ & \times \left[(1+\alpha)\,\overline{E}_b \times \frac{(4\pi)^2 \times (d_i + d_{\text{local}})^2}{G_t G_r \lambda^2} M_l N_f \right. \\ & \left. + \frac{P_c}{R_b} \right] N_i \\ \text{s.t} \quad & \sum_{i=1}^{n} d_i + nd_{\text{local}} = d \\ & d_i > 0, \; i = 1, 2, \ldots, n \\ & n \text{ is a positive integer.} \end{aligned} \tag{15}$$

Proposition 1. *Under the all nodes transmission case, given the total distance between source and destination d, the optimal transmission distance d_i is dependent on d and decreases as the cluster approaches to the destination.*

Proof. See the appendix. □

Therefore, setting hop distances using (A.4) for all hop-lengths is sufficient for achieving the minimum energy consumption.

The numerical results of the optimization and multihop based CMIMO-SM are presented in Figure 10. As an example of numerical results, suppose that the source and destination are d meters apart and $n-1$ clusters act as relays to forward the information. Assume that n is chosen to be 10, d and d_{local} are set to be 1000 meters and 2 meters, respectively. The optimal transmission distances are obtained using (A.4) and plotted in Figure 10. It can be observed that for each hop-length, we can find a best transmission value and also it can be seen that as the distance to the destination increases, the hop-length also increases. The reduction of energy consumption

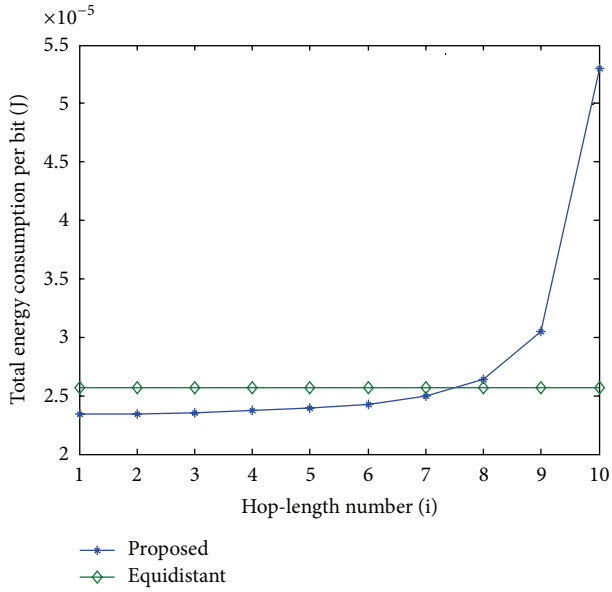


Figure 13: Total energy consumption per bit for each cluster, 3 bits transmission.

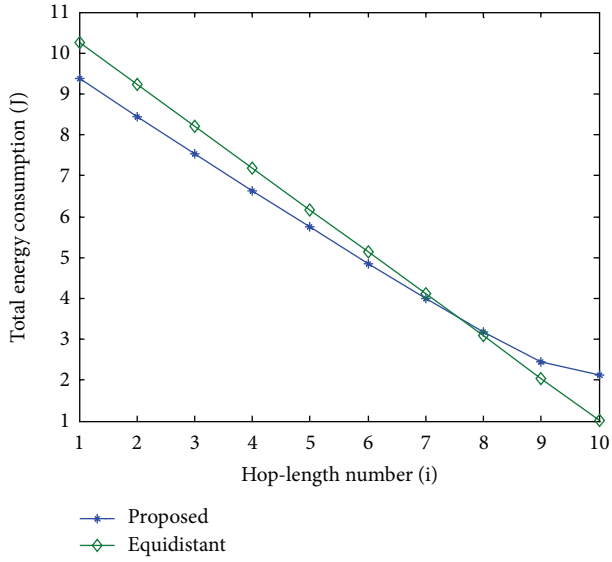


Figure 14: Total energy consumption for each cluster, 3 bits transmission.

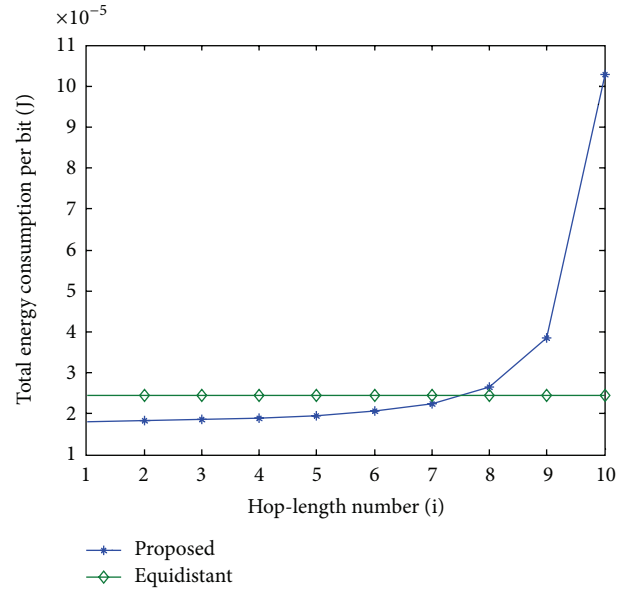


Figure 15: Total energy consumption per bit for each cluster, 4 bits transmission.

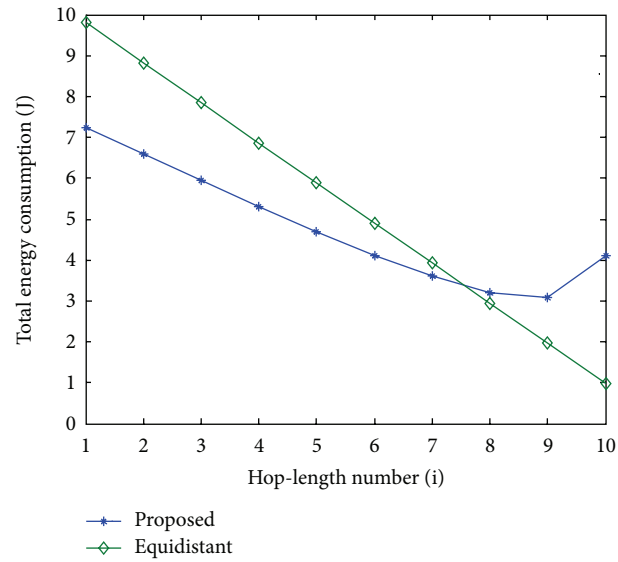


Figure 16: Total energy consumption for each cluster, 4 bits transmission.

can be further obtained using this approach instead of the equidistant approach.

In order to evaluate the performance of the proposed scheme, an equidistant scheme as a reference is compared with the proposed scheme in terms of energy consumption. In Figures 11, 13, and 15, the total energy consumption per bit versus hop-length number is plotted as a calculation aim for the proposed scheme and reference scheme in terms of three different system configurations. It can be seen that the majority of clusters in the proposed scheme have less total energy consumption per bit when compared with the equidistant scheme because transmission distances within these hops are relatively short for the proposed scheme, while the opposite reason can be explained to the minority of the clusters. By using this approach, the total energy consumption is effectively reduced and balanced. After calculating total energy consumption of each cluster, the results in Figures 12, 14, and 16 show that the proposed scheme offers a good energy balance property compared to the equidistant scheme for a given transmission distance.

6. Conclusions

A new cooperative transmission scheme CMIMO-SM has been proposed in this paper. The system model was designed and the energy consumption was analyzed in a cluster to cluster communication environment under three different system configuration cases. The superiority of using this scheme for minimization of total energy consumption is validated by simulations and numerical experiments. Results demonstrated that the proposed scheme always provides energy saving compared with when using the conventional one. Later, we integrated the CMIMO-SM into a multihop scenario to further optimize the total energy consumption. The results indicated that optimal hop scheme can reduce and balance energy consumption significantly, and thus the feasibility of using CMIMO-SM in optimal multihop scheme to boost the joint performance was proved. Therefore, it is concluded that the proposed scheme can be used as a guideline to design energy efficient communications in arbitrary deployed WSN for the reduction of energy consumption and extension of the lifetime.

Appendix

Proof of Proposition

To solve the problem in (15) under the constraint $\sum_{i=1}^{n} d_i = d - nd_{\text{local}}$, we set

$$L = 2\sum_{i=1}^{n}(n+1-i) \times \left[(1+\alpha)\,\overline{E}_b \times \frac{(4\pi)^2 \times (d_i + d_{\text{local}})^2}{G_t G_r \lambda^2} M_l N_f + \frac{P_c}{R_b}\right] N_i + w\left(d - nd_{\text{local}} - \sum_{i=1}^{n} d_i\right), \tag{A.1}$$

where w is a Langrage multiplier. According to the method of Langrage multiplier, we have

$$\frac{\partial L}{\partial d_i} = 0, \qquad \sum_{i=1}^{n} d_i = d - nd_{\text{local}} \tag{A.2}$$

by taking partial derivatives on (A.1) with respect to d_i and equating to 0. Solving (A.2) and each distance d_i can be determined as follows:

$$d_i = \left(\frac{w}{4\,(1+a)\,\overline{E}_b \left((4\pi)^2 / G_t G_r \lambda^2\right) M_l N_f\,(n+1-i)}\right) - d_{\text{local}}. \tag{A.3}$$

Since $\sum_{i=1}^{n} d_i = d - nd_{\text{local}}$, (A.3) can be derived as

$$d_i = \frac{d + nd_{\text{local}}}{\sum_{f=1}^{n} \left(1/\left(n+1-f\right)\right)(n+1-i)} - d_{\text{local}}. \tag{A.4}$$

Conflict of Interests

The authors declare no conflict of interests.

Acknowledgment

This research was supported partly by BK21 and CBNU Research Fund 2013.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[2] D. N. Nguyen and M. Krunz, "Cooperative MIMO in wireless networks: recent developments and challenges," *IEEE Network*, vol. 27, no. 4, pp. 48–54, 2013.

[3] S. Cui, A. J. Goldsmith, and A. Bahai, "Energy-efficiency of MIMO and cooperative MIMO techniques in sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, pp. 1089–1098, 2004.

[4] S. K. Jayaweera, "Virtual MIMO-based cooperative communication for energy-constrained wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 5, no. 5, pp. 984–989, 2006.

[5] R. Y. Mesleh, H. Haas, S. Sinanović, C. W. Ahn, and S. Yun, "Spatial modulation," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 4, pp. 2228–2241, 2008.

[6] X. Li, M. Chen, and W. Liu, "Application of STBC-Encoded cooperative transmissions in wireless sensor networks," *IEEE Signal Processing Letters*, vol. 12, no. 2, pp. 134–137, 2005.

[7] T.-D. Nguyen, O. Berder, and O. Sentieys, "Cooperative MIMO schemes optimal selection for wireless sensor networks," in *Proceedings of the 65th IEEE Vehicular Technology Conference (VTC '07)*, pp. 85–89, Dublin, Ireland, April 2007.

[8] Z. Huang, K. Kobayashi, M. Katayama, and T. Yamazato, "A study on cluster lifetime of single-hop wireless sensor networks with cooperative MISO scheme," *IEICE Transactions on Communications*, vol. 94, no. 10, pp. 2881–2885, 2011.

[9] Z. Zhou, S. Zhou, S. Cui, and J.-H. Cui, "Energy-efficient cooperative communication in a clustered wireless sensor network," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3618–3628, 2008.

[10] Y. Yuan, Z. He, and M. Chen, "Virtual MIMO-based cross-layer design for wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 3, pp. 856–864, 2006.

[11] S. Cui and A. J. Goldsmith, "Cross-layer design of energy-constrained networks using cooperative MIMO techniques," *Signal Processing*, vol. 86, no. 8, pp. 1804–1814, 2006.

[12] B. Li, W. Wang, Q. Yin, R. Yang, Y. Li, and C. Wang, "A new cooperative transmissions metric in wireless sensor networks to minimize energy consumption per unit transmit distance," *IEEE Communications Letters*, vol. 16, no. 5, pp. 626–629, 2012.

[13] M. Islam and J. Kim, "Cooperative technique based on sensor selection in wireless sensor network," *Advances in Electrical and Computer Engineering*, vol. 9, no. 1, pp. 56–62, 2009.

[14] M. R. Islam and J. Kim, "Channel estimated cooperative MIMO in wireless sensor network," *IETE Technical Review*, vol. 25, no. 5, pp. 234–243, 2008.

[15] Y. Gai, L. Zhang, and X. Shan, "Energy efficiency of cooperative MIMO with data aggregation in wireless sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '07)*, pp. 792–797, March 2007.

[16] Q. Gao, Y. Zou, J. Zhang, and X. Peng, "Improving energy-efficiency in a wireless sensor network by combing cooperative MIMO with data aggregation," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3618–3628, 2008.

[17] Z. Huang, H. Okada, K. Kobayashi, and M. Katayama, "A study on cluster lifetime in multi-hop wireless sensor networks with cooperative MISO scheme," *Journal of Communications and Networks*, vol. 14, no. 4, pp. 443–450, 2012.

[18] J. Zhang, L. Fei, Q. Gao, and X.-H. Peng, "Energy-efficient multihop cooperative MISO transmission with optimal hop distance in wireless Ad Hoc networks," *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3426–3435, 2011.

[19] J. Chung, J. Kim, and D. Han, "Multihop hybrid virtual MIMO scheme for wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 9, pp. 4049–4078, 2012.

[20] M. Dohler, A. Aghvami, F. Said, and S. Ghorashi, "Improvements in or Relating to Electronic Data Communication Systems," patent pub. no. WO 03/003672, June 2001.

[21] T. S. Rappaport, *Wireless Communications, Principle and Practice*, Prentice Hall, Upper Saddle River, NJ, USA, 2nd edition, 2001.

[22] S. Cui, A. J. Goldsmith, and A. Bahai, "Energy-constrained modulation optimization," *IEEE Transactions on Wireless Communications*, vol. 4, no. 5, pp. 2349–2360, 2005.

[23] TelosB Datasheet, http://www.memsic.com/.

[24] J. J. Ahmad, H. A. Khan, and S. A. Khayam, "Energy efficient video compression for wireless sensor networks," in *Proceedings of the 43rd Annual Conference on Information Sciences and Systems (CISS '09)*, pp. 629–634, Baltimore, Md, USA, March 2009.

[25] H. Xu, "Simplified maximum likelihood-based detection schemes for M-ray quadrature amplitude modulation spatial modulation," *IET Communications*, vol. 6, pp. 1356–1363, 2012.

[26] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 8, pp. 1451–1458, 1998.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 241469, 11 pages
http://dx.doi.org/10.1155/2014/241469

Research Article

Augmented Lagrange Based on Modified Covariance Matching Criterion Method for DOA Estimation in Compressed Sensing

Weijian Si, Xinggen Qu, and Lutao Liu

Department of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China

Correspondence should be addressed to Lutao Liu; liulutao@msn.com

Received 27 November 2013; Accepted 19 December 2013; Published 11 February 2014

Academic Editors: H. R. Karimi, Z. Yu, and W. Zhang

A novel direction of arrival (DOA) estimation method in compressed sensing (CS) is presented, in which DOA estimation is considered as the joint sparse recovery from multiple measurement vectors (MMV). The proposed method is obtained by minimizing the modified-based covariance matching criterion, which is acquired by adding penalties according to the regularization method. This minimization problem is shown to be a semidefinite program (SDP) and transformed into a constrained quadratic programming problem for reducing computational complexity which can be solved by the augmented Lagrange method. The proposed method can significantly improve the performance especially in the scenarios with low signal to noise ratio (SNR), small number of snapshots, and closely spaced correlated sources. In addition, the Cramér-Rao bound (CRB) of the proposed method is developed and the performance guarantee is given according to a version of the restricted isometry property (RIP). The effectiveness and satisfactory performance of the proposed method are illustrated by simulation results.

1. Introduction

Direction of arrival (DOA) estimation of multiple narrowband sources has been an interesting research topic in array signal processing. Its applications span many fields including radar, communication systems, and medical imaging [1, 2]. Many effective algorithms are proposed for DOA estimation, which are mainly classified into three categories. The beamforming algorithms such as MVDR [3] and MEM [4] obtain a nonparametric spatial spectrum by optimizing the filter weights. The subspace algorithms such as MUSIC [5] and ESPRIT [6] and their derivatives [7, 8] exploit the orthogonality of signal subspace and noise subspace for DOA estimation. The subspace fitting algorithms including Maximum Likelihood (ML) [9] and Weighted Subspace Fitting (WSF) [10] solve a multidimensional nonlinear optimization problem to obtain a precise estimation, but a good initialization is required to ensure global convergence and computational complexity is very high. All these algorithms focus on two important issues: resolution (i.e., the ability to correctly resolve two closely spaced sources) and precision (i.e., the deviation from true DOA) which are considered to be the theoretical bases of evaluating certain algorithms. Many high resolution methods suffer from serious performance degradation in the scenarios with low SNR, small number of snapshots, and closely spaced corrected sources. More recently, many research applications involving compressed sensing (CS) [11], especially DOA estimation [12, 13], have become more and more popular in the signal processing. Moreover, the restricted isometry property (RIP) based on the perfect theoretical framework given with modern probability theory by Donoho [11] and Candés et al. [14] provides the performance guarantee in CS. Hence, in this paper, DOA estimation is posed as a joint sparse recovery problem where we recover jointly sparse sources from multiple measurement vectors (MMV) under CS framework.

CS is an emerging area which can break through the limit of Nyquist sampling theorem. On one hand, CS can simultaneously capture and store compressed or sparse source at a rate much lower than that of Nyquist sampling. On the other hand, it can recover the original source using nonadaptive linear projection measurements onto a suitable measurement matrix which satisfies the RIP. In CS, the joint sparse recovery aims to find a common support shared by unknown sparse

vectors from MMV, which is obtained by the measurement matrix. Support denotes the indices of the nonzero elements in the unknown sparse vectors. A sparse solution can be obtained as long as the support is determined.

CS theory has been widely applied to DOA estimation according to the source sparsity, which results from the fact that there are much fewer source directions than all potential directions in the spatial domain. The DOA estimation methods in CS are attractive since they have much better estimation performance than conventional estimation methods. In [15], Gürbüz et al. firstly formulate the DOA estimation problem under CS framework in the time domain. Wang et al. [16] propose a new architecture to estimate DOA by exploiting compressive sampling in the spatial domain. Stoica et al. [17] make full use of covariance matching criterion and present a semiparametric/sparse estimation method and its derivative called LIKES [18] for the separable model. Malioutov et al. [19] present the l_1-SVD algorithm for DOA estimation which combines the SVD of the data matrix with a sparse recovery method based on l_1-norm minimization. A new class of subspace-base algorithms such as compressive MUSIC (CS-MUSIC) [20] and subspace-augmented MUSIC (SA-MUSIC) [21] is proposed in recent years.

The RIP and various modified versions of it have been used as a foundation of performance guarantee [21–24] for the joint sparse recovery. The performance guarantee of MUSIC based on joint sparse recovery is given for identifying the unique support in a favorable case [25]. However, in the unfavorable case where the number of measurement vectors is smaller than the sparsity or the covariance matrix tends to lose rank due to correlated sources, performance guarantee fails. Lee et al. [21] propose a new performance guarantee in terms of a version of the RIP under such unfavorable conditions. The performance guarantees of other methods such as greedy algorithms and convex relaxation have been developed to find the sparse solution in [23, 24]. However, the guarantees of these methods cannot be simply extended to the MMV case to obtain a better bound for the sparse solution.

In this paper, we propose a novel augmented Lagrange based on modified covariance matching criterion method called AULMC for DOA estimation in CS. This method utilizes the minimization of the modified covariance matching criterion which is acquired by using regularization method to add penalties in order to obtain the stable sparse parameter estimation, especially in the low sparsity case. This minimization problem is shown to be a semidefinite program (SDP) and transformed into the constrained quadratic programming problem for the sake of reducing computational complexity. The augmented Lagrange function is formed to solve this problem by the use of augmented Lagrange method. AULMC has a number of advantages over the other methods. For example, it provides more precise estimation and higher resolution in the scenarios with low SNR, small number of snapshots, and closely spaced correlated sources, and it does not need priori knowledge about the number of sources or to choose the regularization parameter of the l_1-optimization framework which is very difficult to select in the DOA estimation. In addition, we give a detailed derivation process of the closed-form expression for the Cramér-Rao bound (CRB) of the new method and discuss an explicit condition that guarantees performance of the new method. This performance guarantee is given in terms of weaker version of the RIP which is referred to as weak-1 RIP. Simulation results illustrate the performance of the proposed method.

It is worth noting that covariance matching criterion has been used for DOA estimation [26]. In [26], a sparse iterative covariance-base estimation method, abbreviated as SPICE, is proposed. Our approach is different from SPICE because it utilizes modified covariance matching criterion which can guarantee the stability of solution even if the source sparsity is rather low. In the future work, we will focus on the application of our approach to data-driven design [27–30]. Now we briefly summarize the contributions of this work as follows.

(i) A modified covariance matching criterion is proposed by adding penalties according to the regularization method.

(ii) The original SDP problem is transformed into the constrained quadratic programming problem. The motivation to transform the problem is that it can reduce computational complexity.

(iii) Augmented Lagrange based on modified covariance matching criterion method is devised to solve the resulting programming problem.

(iv) The CRB and performance guarantee of the new method are given in detail.

The rest of this paper is organized as follows. In Section 2, we formulate the DOA estimation problem. Section 3 presents a modified covariance matching criterion. A novel augmented Lagrange based on modified covariance matching criterion method for DOA estimation is described in detail in Section 4; the performance of which is analyzed in Section 5. The performance of the proposed method is evaluated by simulations in Section 6. Conclusions are provided in Section 7.

2. Problem Formulation

Consider L narrow-band sources $\widehat{s}_1, \widehat{s}_2, \ldots, \widehat{s}_L$ impinging on a sensor array that consisted of P omnidirectional sensors from far field. At time instant t, the array received source can be given by

$$\mathbf{x}(t) = \sum_{k=1}^{L} \mathbf{a}(\theta_k)\, \widehat{s}_k(t) + \mathbf{n}(t), \tag{1}$$

where $\mathbf{n}(t) \in \mathscr{C}^{P\times 1}$ denotes a noise term, $\theta_k \in \Omega$ is the unknown direction, of the kth source where Ω denotes the entire spatial range and $\mathbf{a}(\theta_k)$ is the $P \times 1$ steering vector. Although the DOA estimation of the single snapshot, which is a typical single measurement vector (SMV) model, has its value, the number of snapshots is larger than one in the most practical applications. Correspondingly, the multiple snapshots model is a MMV model.

Let $\{\widetilde{\theta}_k\}_{k=1}^{K}$ denote a fine grid which covers Ω where there exist K ($K \gg L$) potential directions of the sources $s_1, s_2, \ldots, s_K$ so that the true directions $\{\theta_k\}_{k=1}^{L}$ are aligned or are close to the grids. This means that if $\widetilde{\theta}_{k_1}, \widetilde{\theta}_{k_2}, \ldots, \widetilde{\theta}_{k_L}$ are equal to $\theta_1, \theta_2, \ldots, \theta_L$, respectively, we have

$$s_k = \begin{cases} \widehat{s}_l, & k = k_l \ (l = 1, 2, \ldots, L), \\ 0, & \text{elsewhere.} \end{cases} \tag{2}$$

Hence, the multiple snapshots model can be written as the following sparse form:

$$\mathbf{x}(t) = \sum_{k=1}^{K} \mathbf{a}\left(\widetilde{\theta}_k\right) s_k(t) + \mathbf{n}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t), \quad t = t_1, t_2 \ldots t_M, \tag{3}$$

where $\mathbf{A} = [\mathbf{a}(\widetilde{\theta}_1) \quad \mathbf{a}(\widetilde{\theta}_2) \quad \cdots \quad \mathbf{a}(\widetilde{\theta}_K)]$ is the $P \times K$ manifold matrix corresponding to all the potential directions which is also referred to as an overcomplete dictionary in CS. $\mathbf{s}(t) = [s_1(t) \ \ s_2(t) \ \cdots \ s_K(t)]^T$ is a L-sparse vector since it has at most L nonzero elements in K elements, and L is defined as sparsity where the operator $[\cdot]^T$ denotes transport. $\{\mathbf{s}(t_i)\}_{i=1}^{M}$ are jointly L-sparse if they share a common support. Hence, the matrix $\mathbf{S} = [\mathbf{s}(t_1) \ \ \mathbf{s}(t_2) \ \cdots \ \mathbf{s}(t_M)] \in \mathscr{C}^{K \times M}$ has no more than L nonzero rows in order to be called row L-sparse. The MMV problem is that of identifying the row support of the unknown matrix $\mathbf{S}$ from the matrix $\mathbf{Y} \in \mathscr{C}^{N \times M}$ that consisted of MMV which is given by

$$\mathbf{Y} = [\mathbf{y}(t_1) \quad \mathbf{y}(t_2) \quad \cdots \quad \mathbf{y}(t_M)] = \mathbf{\Phi}\mathbf{A}\mathbf{S} + \mathbf{\Phi}\mathbf{N} \tag{4}$$

with a common measurement matrix $\mathbf{\Phi}$ of the size $N \times P$ with $N < P$ where N is the number of nonadaptive linear projection measurements, such as random Gaussian matrix or random partial Fourier matrix, and noise matrix $\mathbf{N}$.

3. Modified Covariance Matching Criterion

In this section, the modified covariance matching criterion is developed in the CS scenario. A conventional covariance matrix of the compressed measurement source with the size $N \times N$ is given by

$$\mathbf{R}_y = E[\mathbf{y}(t)\mathbf{y}^*(t)] = \mathbf{\Phi}\mathbf{A}\mathbf{R}_s\mathbf{A}^*\mathbf{\Phi}^* + \mathbf{\Phi}\mathbf{R}_n\mathbf{\Phi}^*, \tag{5}$$

where $\mathbf{R}_s = E[\mathbf{s}(t)\mathbf{s}^*(t)]$ is a $K \times K$ covariance matrix of the sparse source whose off-diagonal elements denote the source correlation and diagonal elements denote the source powers. Since the powers are one-to-one corresponding to all the potential directions and our focus is on the source angle parameter estimation, $\mathbf{R}_s$ can be reduced to a diagonal matrix $\mathbf{R}_s(\widetilde{\boldsymbol{\theta}}) = \operatorname{diag}\left(\widetilde{\theta}_1 \ \ \widetilde{\theta}_2 \ \cdots \ \widetilde{\theta}_K\right)$. According to (5), the measurement matrix can change the covariance matrix of the noise to render the noise colored even if the noise is white. Therefore, this adverse factor must be considered before recovering jointly sparse sources. In addition, the operators $(\cdot)^*$ and $E[\cdot]$ denote conjugate transpose and expectation, respectively.

Since $\mathbf{\Phi}\mathbf{R}_n\mathbf{\Phi}^*$ is a positive definite Hermitian matrix, a prewhitened method is given by the Cholesky decomposition. Let $\mathbf{B}$ be the Cholesky factor that satisfies

$$\left(\mathbf{\Phi}\mathbf{R}_n\mathbf{\Phi}^*\right)^{-1} = \mathbf{B}^*\mathbf{B}, \tag{6}$$

where $\mathbf{B} \in \mathscr{C}^{N \times N}$ is an upper triangular matrix of positive line. Hence, a prewhitened process is implemented by multiplying $\mathbf{y}(t)$ by $\mathbf{B}$ in order to obtain a pure source whose covariance matrix is written as

$$\widetilde{\mathbf{R}}_y = \mathbf{B}\mathbf{\Phi}\mathbf{A}\mathbf{R}_s\mathbf{A}^*\mathbf{\Phi}^*\mathbf{B}^* + \mathbf{I}_N = \mathbf{C}\mathbf{R}_s\mathbf{C}^* + \mathbf{I}_N, \tag{7}$$

where $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K]$ and $\mathbf{I}_N$ is an identity matrix of the size $N \times N$. It is important to note that the unknown covariance matrix of the noise is transformed into the known identity matrix after the prewhitened process. Therefore, the prewhitened method improves the robustness to the noise. Then, the covariance matrix of the compressed measurement source denoising is realized by

$$\mathbf{R} = \widetilde{\mathbf{R}}_y - \mathbf{I}_N = \mathbf{C}\mathbf{R}_s\mathbf{C}^*. \tag{8}$$

The parameter $\widetilde{\boldsymbol{\theta}}$ can be estimated by a class of the covariance matching estimation techniques (COMET) based on covariance matching criterion [31]. This parameter estimation method is well known to be a large-sample approximation of ML method and provide a more attractive solution than ML estimator.

The principle of COMET is that of using the right data to minimize its data model by the weighted least-squares (WLS) method. However, the lower the source sparsity is, the more likely it is to be ill-posed for the covariance matrix estimation error meaning that the optimal solution obtained directly by minimizing the conventional covariance matching criterion is instable. Hence, we employ regularization method to add penalties in order to sufficiently exploit prior knowledge to reduce the solution space for determining the stable optimal solution. The modified covariance matching criterion is proposed as the following form:

$$\left[\operatorname{vec}\left(\widehat{\mathbf{R}}\right) - \operatorname{vec}\left(\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)\right)\right]^* \mathbf{\Gamma}^* \mathbf{P}^{-1} \mathbf{\Gamma} \left[\operatorname{vec}\left(\widehat{\mathbf{R}}\right) - \operatorname{vec}\left(\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)\right)\right] + \alpha \operatorname{Tr}\left[\mathbf{R}^{-1}\left(\widetilde{\boldsymbol{\theta}}\right) - \widehat{\mathbf{R}}^{-1}\right] + \beta \operatorname{Tr}\left[\frac{\mathbf{R}^{-1}\left(\widetilde{\boldsymbol{\theta}}\right)\widehat{\mathbf{R}}^{-1}}{4}\right], \tag{9}$$

where parameter $\alpha \geq 0$ controls the solution smoothness (guarantee precision), parameter $\beta \geq 0$ controls the solution scale (guarantee sparsity), the inverses of the matrix $\mathbf{\Gamma}$, the sample covariance matrix $\widehat{\mathbf{R}}$, and $\mathbf{R}(\widetilde{\boldsymbol{\theta}})$ are assumed to exist, and the matrix $\mathbf{P}^{-1}$ is the inverse of the covariance matrix of the residuals, $\widetilde{\boldsymbol{\varepsilon}} = \mathbf{\Gamma}\operatorname{vec}(\widehat{\mathbf{R}} - \mathbf{R}(\widetilde{\boldsymbol{\theta}}))$. Since $\widehat{\mathbf{R}}$ is equal to $\mathbf{R}(\widetilde{\boldsymbol{\theta}})$ as $M \to \infty$, it follows from [32] that $\operatorname{vec}(\widehat{\mathbf{R}} - \mathbf{R}(\widetilde{\boldsymbol{\theta}}))$ satisfies the asymptotic normal distribution with mean zero and variance $N^{-1}\mathbf{R}^T(\widetilde{\boldsymbol{\theta}}) \otimes \widehat{\mathbf{R}}$. In addition, the operators $\operatorname{Tr}[\cdot]$, $\operatorname{vec}(\cdot)$, and $\otimes$ denote matrix trace, column stacking operation,

and Kronecker product, respectively. Then, the matrix $\mathbf{P}$ can be given by

$$\begin{aligned}\mathbf{P} &= E\left[\widetilde{\boldsymbol{\varepsilon}}\left(\widetilde{\boldsymbol{\theta}}\right)\widetilde{\boldsymbol{\varepsilon}}^{*}\left(\widetilde{\boldsymbol{\theta}}\right)\right] \\ &= \boldsymbol{\Gamma} E\left[\operatorname{vec}\left(\widehat{\mathbf{R}}-\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)\right)\operatorname{vec}^{*}\left(\widehat{\mathbf{R}}-\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)\right)\right]\boldsymbol{\Gamma}^{*} \\ &= \boldsymbol{\Gamma} N^{-1}\mathbf{R}^{T}\left(\widetilde{\boldsymbol{\theta}}\right)\otimes\widehat{\mathbf{R}}\boldsymbol{\Gamma}^{*} = \boldsymbol{\Gamma} N^{-1}\mathbf{D}\boldsymbol{\Gamma}^{*}.\end{aligned} \tag{10}$$

We consider the normalized data model in (9) and choose the regularization parameters $\alpha = \beta = N$ based on perceptual criterion [33]. By substituting (10) into (9), we have

$$\begin{aligned}&\operatorname{vec}^{*}\left(\widehat{\mathbf{R}}-\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)\right)\mathbf{D}^{-1}\operatorname{vec}\left(\widehat{\mathbf{R}}-\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)\right) \\ &\qquad + \operatorname{Tr}\left[\mathbf{R}^{-1}\left(\widetilde{\boldsymbol{\theta}}\right)-\widehat{\mathbf{R}}^{-1}\right] + \frac{\operatorname{Tr}\left[\mathbf{R}^{-1}\left(\widetilde{\boldsymbol{\theta}}\right)\widehat{\mathbf{R}}^{-1}\right]}{4}.\end{aligned} \tag{11}$$

By the properties of vec, $\otimes$, and Tr, the data model can be further simplified to

$$\begin{aligned}&\operatorname{vec}^{*}\left(\widehat{\mathbf{R}}-\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)\right)\mathbf{D}^{-1}\operatorname{vec}\left(\widehat{\mathbf{R}}-\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)\right) \\ &\qquad + \operatorname{Tr}\left[\mathbf{R}^{-1}\left(\widetilde{\boldsymbol{\theta}}\right)\left(\widehat{\mathbf{R}}-\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)\right)\widehat{\mathbf{R}}^{-1}\right] + \frac{\operatorname{Tr}\left[\mathbf{R}^{-1}\left(\widetilde{\boldsymbol{\theta}}\right)\widehat{\mathbf{R}}^{-1}\right]}{4} \\ &= \left\|\widehat{\mathbf{R}}^{-1/2}\left(\widehat{\mathbf{R}}-\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)+\frac{\mathbf{I}_N}{2}\right)\mathbf{R}^{-1/2}\left(\widetilde{\boldsymbol{\theta}}\right)\right\|^{2},\end{aligned} \tag{12}$$

where $\|\cdot\|$ denotes the Frobenius norm for matrices or the Euclidean norm for vectors. The data model in (12) is considered to be the modified covariance matching criterion. It can be seen from (12) that a positive definite Hermitian matrix is added to the covariance matrix estimation error by applying penalties according to regularization method in order to guarantee the stability of solution.

4. DOA Estimation

In this section, we will utilize the minimization of the modified covariance matching criterion to estimate parameter $\widetilde{\boldsymbol{\theta}}$. Let

$$\widetilde{\boldsymbol{\theta}}_{s} = \arg\min_{\widetilde{\boldsymbol{\theta}}}\left\|\widehat{\mathbf{R}}^{-1/2}\left(\widehat{\mathbf{R}}-\mathbf{R}\left(\widetilde{\boldsymbol{\theta}}\right)+\frac{\mathbf{I}}{2}\right)\mathbf{R}^{-1/2}\left(\widetilde{\boldsymbol{\theta}}\right)\right\|^{2} \tag{13}$$

be the optimal solution of $\widetilde{\boldsymbol{\theta}}$ in the structure model of (8). Our goal is to utilize the modified covariance matching criterion for an estimate that is asymptotically equal to $\widetilde{\boldsymbol{\theta}}_{s}$. By the properties of the trace and the Hermitian matrix, a derivation process is shown as follows, where we omit the dependence on $\widetilde{\boldsymbol{\theta}}$ for notational convenience:

$$\begin{aligned}f &= \operatorname{Tr}\left[\widehat{\mathbf{R}}^{-1}\left(\widehat{\mathbf{R}}-\mathbf{R}+\frac{\mathbf{I}}{2}\right)\mathbf{R}^{-1}\left(\widehat{\mathbf{R}}-\mathbf{R}+\frac{\mathbf{I}}{2}\right)\right] \\ &= \operatorname{Tr}\left[\widehat{\mathbf{R}}^{-1}\mathbf{R}-\widehat{\mathbf{R}}^{-1}+\left(\widehat{\mathbf{R}}^{1/2}+\frac{1}{2}\widehat{\mathbf{R}}^{-1/2}\right)\right. \\ &\qquad \left.\cdot\mathbf{R}^{-1/2}\left(\widehat{\mathbf{R}}^{1/2}+\frac{1}{2}\widehat{\mathbf{R}}^{-1/2}\right)\mathbf{R}^{-1/2}\right].\end{aligned} \tag{14}$$

Due to

$$\operatorname{Tr}\left[\widehat{\mathbf{R}}^{-1}\mathbf{R}\right] = \sum_{k=1}^{K}\left(\mathbf{c}_{k}^{*}\widehat{\mathbf{R}}^{-1}\mathbf{c}_{k}\right)\widetilde{\theta}_{k} \tag{15}$$

we can deduce that the minimization of f is equal to the minimization of h:

$$\begin{aligned}h = &\operatorname{Tr}\left[\left(\widehat{\mathbf{R}}^{1/2}+\frac{1}{2}\widehat{\mathbf{R}}^{-1/2}\right)\mathbf{R}^{-1/2}\left(\widehat{\mathbf{R}}^{1/2}+\frac{1}{2}\widehat{\mathbf{R}}^{-1/2}\right)\mathbf{R}^{-1/2}\right] \\ &+\sum_{k=1}^{K}\left(\mathbf{c}_{k}^{*}\widehat{\mathbf{R}}^{-1}\mathbf{c}_{k}\right)\widetilde{\theta}_{k}.\end{aligned} \tag{16}$$

Then, we will demonstrate that the minimization of h in (16) with respect to $\{\widetilde{\theta}_{k}\}_{k=1}^{K}$ is an SDP. To prove this fact, One assumes

$$\widehat{\mathbf{R}}^{1/2}+\frac{1}{2}\widehat{\mathbf{R}}^{-1/2} = \begin{bmatrix}\mathbf{r}_1 & \mathbf{r}_2 & \cdots & \mathbf{r}_N\end{bmatrix}. \tag{17}$$

The h in (16) can be rewritten as

$$h = \sum_{k=1}^{N}\mathbf{r}_{k}^{*}\mathbf{R}^{-1}\mathbf{r}_{k} + \sum_{k=1}^{K}\left(\mathbf{c}_{k}^{*}\widehat{\mathbf{R}}^{-1}\mathbf{c}_{k}\right)\widetilde{\theta}_{k}. \tag{18}$$

By the Schur complement, let $\{x_k\}_{k=1}^{N}$ be auxiliary variables satisfying

$$\begin{bmatrix}x_k & \mathbf{r}_{k}^{*} \\ \mathbf{r}_{k} & \mathbf{R}\end{bmatrix} \geq 0. \tag{19}$$

The equivalent form of this minimization problem is expressed as

$$\begin{aligned}\min_{x_k,\ \widetilde{\theta}_k}\quad &\sum_{k=1}^{N}x_k + \sum_{k=1}^{K}\left(\mathbf{c}_{k}^{*}\widehat{\mathbf{R}}^{-1}\mathbf{c}_{k}\right)\widetilde{\theta}_{k} \\ \text{s.t.}\quad &\widetilde{\theta}_{k}\geq 0, \quad k=1,2,\ldots,K, \\ &\begin{bmatrix}x_k & \mathbf{r}_{k}^{*} \\ \mathbf{r}_{k} & \mathbf{R}\end{bmatrix}\geq 0, \quad k=1,2,\ldots,N.\end{aligned} \tag{20}$$

It is easy to see that (20) is an SDP [34]. Many software packages can solve an SDP, but solving (16) as an SDP is not a good choice because SDP solvers have generally rather high computational complexity for the values of N, M, and K in the DOA estimation. To solve it effectively, we transform it into the constrained quadratic programming problem for reducing computational complexity, as described next.

Since a consistent estimation of $\widetilde{\boldsymbol{\theta}}$ is given by (15), we can reformulate the minimization of h in (16) as the following constrained minimization by the Schur inequality of the trace:

$$\begin{aligned}\min_{\theta_i\geq 0}\quad &\operatorname{Tr}\left[\left(\widehat{\mathbf{R}}^{1/2}+\frac{1}{2}\widehat{\mathbf{R}}^{-1/2}\right)\mathbf{R}^{-1/2}\left(\widehat{\mathbf{R}}^{1/2}+\frac{1}{2}\widehat{\mathbf{R}}^{-1/2}\right)\mathbf{R}^{-1/2}\right] \\ \text{s.t.}\quad &\mathbf{W}\mathbf{z} = \mathbf{1}_K,\end{aligned} \tag{21}$$

where $\mathbf{W}$ is a $K \times K$ diagonal matrix of $[\mathbf{C}^* \widehat{\mathbf{R}}^{-1} \mathbf{C}]^{-1/2}$ and $\mathbf{z} = [z_1 \ z_2 \ \cdots \ z_K]^T$ is a K-dimensional vector with $z_i = \theta_i^{-1/2}$, $i = 1, 2, \ldots, K$. By substituting (8) into (21), the objective function in (21) can be rewritten as

$$\begin{aligned} &\mathrm{Tr}\left[\left(\widehat{\mathbf{R}}^{1/2} + \frac{1}{2}\widehat{\mathbf{R}}^{-1/2}\right)(\mathbf{C}^*)^{-1/2}\mathbf{R}_s^{-1/2}\mathbf{C}^{-1/2}\right. \\ &\quad \left.\cdot\left(\widehat{\mathbf{R}}^{1/2} + \frac{1}{2}\widehat{\mathbf{R}}^{-1/2}\right)(\mathbf{C}^*)^{-1/2}\mathbf{R}_s^{-1/2}\mathbf{C}^{-1/2}\right] \\ &= \mathrm{Tr}\left[\mathbf{C}^{-1/2}\left(\widehat{\mathbf{R}}^{1/2} + \frac{1}{2}\widehat{\mathbf{R}}^{-1/2}\right)(\mathbf{C}^*)^{-1/2}\mathbf{R}_s^{-1/2}\right. \\ &\quad \left.\cdot\mathbf{C}^{-1/2}\left(\widehat{\mathbf{R}}^{1/2} + \frac{1}{2}\widehat{\mathbf{R}}^{-1/2}\right)(\mathbf{C}^*)^{-1/2}\mathbf{R}_s^{-1/2}\right]. \end{aligned} \tag{22}$$

Based on the following equation:

$$\mathrm{Tr}\left[\mathbf{MT}(\mathbf{d})\mathbf{NT}(\mathbf{d})\right] = \mathbf{d}^*\left(\mathbf{M}^T \odot \mathbf{N}\right)\mathbf{d}, \tag{23}$$

where $\odot$ denotes the Schur-Hadamard product, $\mathbf{M}$ and $\mathbf{N}$ are both $K \times K$ matrices, $\mathbf{T}(\mathbf{d}) = \mathrm{diag}(d_1 \ \ d_2 \ \cdots \ d_K)$, and $\mathbf{d} = [d_1 \ \ d_2 \ \cdots \ d_K]^T$, the objective function (22) can be written as

$$\mathrm{Tr}\left[\mathbf{Q}\mathbf{R}_s^{-1/2}\mathbf{Q}\mathbf{R}_s^{-1/2}\right] = \mathbf{z}^*\mathbf{V}\mathbf{z}, \tag{24}$$

where $\mathbf{Q} = \mathbf{C}^{-1/2}(\widehat{\mathbf{R}}^{1/2} + (1/2)\widehat{\mathbf{R}}^{-1/2})(\mathbf{C}^*)^{-1/2}$ and $\mathbf{V} = \mathbf{Q}^T \odot \mathbf{Q}$. Hence, the minimization problem in (21) is transformed into the following form:

$$\begin{aligned} \min_{z} \quad & f(\mathbf{z}) = \mathbf{z}^*\mathbf{V}\mathbf{z} \\ \text{s.t.} \quad & \mathbf{\Pi}(\mathbf{z}) = \mathbf{W}\mathbf{z} - \mathbf{1}_K = \mathbf{0}_K \end{aligned} \tag{25}$$

which is a typical constrained quadratic programming problem.

It is well known that an important class of methods for solving the constrained quadratic programming problem is to form the auxiliary function. To solve (25), we form the following augmented Lagrange function with respect to (25):

$$\mathbf{p}_\sigma(\mathbf{z}, \overline{\boldsymbol{\mu}}) = f(\mathbf{z}) + \overline{\boldsymbol{\mu}}^*\mathbf{\Pi}(\mathbf{z}) + \frac{1}{2}\sigma\|\mathbf{\Pi}(\mathbf{z})\|^2, \tag{26}$$

where $\overline{\boldsymbol{\mu}}$ is the asymptotical solution of the Lagrange multiplier of (25) and σ is a penalty factor.

By setting the gradient and the Hessian matrix of (26) with respect to $\mathbf{z}$ to zero, we have

$$\begin{aligned} \nabla_{\mathbf{z}}\mathbf{p}_\sigma(\mathbf{z}, \overline{\boldsymbol{\mu}}) &= \nabla_{\mathbf{z}} l(\mathbf{z}, \overline{\boldsymbol{\mu}}) + \sigma\mathbf{q}(\mathbf{z})\mathbf{\Pi}(\mathbf{z}), \\ \nabla^2_{\mathbf{zz}}\mathbf{p}_\sigma(\mathbf{z}, \overline{\boldsymbol{\mu}}) &= \nabla^2_{\mathbf{zz}} l(\mathbf{z}, \overline{\boldsymbol{\mu}}) + \sigma\sum_{i=1}^{K}\mathbf{\Pi}_i(\mathbf{z})\nabla^2\mathbf{\Pi}(\mathbf{z}) \\ &\quad + \sigma\mathbf{q}(\mathbf{z})\mathbf{q}^*(\mathbf{z}), \end{aligned} \tag{27}$$

where $l(\mathbf{z}, \overline{\boldsymbol{\mu}}) = f(\mathbf{z}) + \overline{\boldsymbol{\mu}}^*\mathbf{\Pi}(\mathbf{z})$ and $\mathbf{q}(\mathbf{z}) = \nabla\mathbf{\Pi}^*(\mathbf{z}) = [\nabla\Pi_1(\mathbf{z}) \ \ \nabla\Pi_2(\mathbf{z}) \ \cdots \ \nabla\Pi_K(\mathbf{z})]$. One assumes that

$$\mathbf{b}(\mathbf{z}, \overline{\boldsymbol{\mu}}) = \nabla^2_{\mathbf{zz}} l(\mathbf{z}, \overline{\boldsymbol{\mu}}) + \sigma\sum_{i=1}^{K}\mathbf{\Pi}_i(\mathbf{z})\nabla^2\mathbf{\Pi}(\mathbf{z}). \tag{28}$$

Applying the Newton method, we obtain

$$\nabla^2_{\mathbf{zz}}\mathbf{p}_{\sigma_k}(\mathbf{z}_k, \overline{\boldsymbol{\mu}}_k)(\mathbf{z}_{k+1} - \mathbf{z}_k) = -\nabla_{\mathbf{z}}\mathbf{p}_{\sigma_k}(\mathbf{z}_k, \overline{\boldsymbol{\mu}}_k). \tag{29}$$

For notational convenience, we assume that $\mathbf{q}_k = \mathbf{q}(\mathbf{z}_k)$, $\nabla f_k = \nabla f(\mathbf{z}_k)$, $\mathbf{\Pi}_k = \mathbf{\Pi}(\mathbf{z}_k)$, $\mathbf{b}_k = \mathbf{b}(\mathbf{z}_k, \overline{\boldsymbol{\mu}}_k)$, $\mathbf{d}_k = \mathbf{z}_{k+1} - \mathbf{z}_k$ and we get the following equation by (29):

$$\mathbf{b}_k\mathbf{d}_k + \sigma_k\mathbf{q}_k\mathbf{q}_k^*\mathbf{d}_k + \nabla_{\mathbf{z}} l(\mathbf{z}_k, \overline{\boldsymbol{\mu}}_k) + \sigma_k\mathbf{q}_k\mathbf{\Pi}_k = \mathbf{0}. \tag{30}$$

Assuming that the inverse of $\mathbf{b}_k$ exists, by left-multiplying (30) by $(1/\sigma_k)\mathbf{q}_k^*\mathbf{b}_k^{-1}$, we have

$$\begin{aligned} &\left(\frac{\mathbf{I}_K}{\sigma_k} + \mathbf{q}_k^*\mathbf{b}_k^{-1}\mathbf{q}_k\right)\mathbf{q}_k^*\mathbf{d}_k \\ &\quad = -\mathbf{q}_k^*\mathbf{b}_k^{-1}\mathbf{q}_k\mathbf{\Pi}_k - \frac{1}{\sigma_k}\mathbf{q}_k^*\mathbf{b}_k^{-1}\nabla_{\mathbf{z}} l(\mathbf{z}_k, \overline{\boldsymbol{\mu}}_k). \end{aligned} \tag{31}$$

It follows from (31) that

$$\mathbf{q}_k^*\mathbf{b}_k = -\mathbf{\Pi}_k + \frac{\boldsymbol{\alpha}_k}{\sigma_k}, \tag{32}$$

where $\boldsymbol{\alpha}_k$ satisfies

$$\left(\frac{\mathbf{I}_K}{\sigma_k} + \mathbf{q}_k^*\mathbf{b}_k^{-1}\mathbf{q}_k\right)\boldsymbol{\alpha}_k = -\mathbf{q}_k^*\mathbf{b}_k^{-1}\nabla_{\mathbf{z}} l(\mathbf{z}_k, \overline{\boldsymbol{\mu}}_k) + \mathbf{\Pi}_k. \tag{33}$$

By substituting (32) into (30), we have

$$\mathbf{d}_k = -\mathbf{b}_k^{-1}\left(\mathbf{q}_k\boldsymbol{\alpha}_k + \nabla_{\mathbf{z}} l(\mathbf{z}_k, \overline{\boldsymbol{\mu}}_k)\right). \tag{34}$$

Both the multiplier factor and the penalty factor are determined with difficulty for utilizing the augmented Lagrange function to solve the constrained quadratic programming problem. Hence, an updated sequence for the multiplier factor is given in terms of Proposition 1.

Proposition 1. *One assumes that $\boldsymbol{\mu}(\mathbf{z})$ is the optimal solution of the problem $\min_{\boldsymbol{\mu}\in\mathscr{C}^K}\|\mathbf{q}(\mathbf{z})\boldsymbol{\mu} + \nabla f(\mathbf{z})\|^2$. Then, the following equation holds:*

$$\boldsymbol{\mu}(\mathbf{z}_k) = \overline{\boldsymbol{\mu}}_k + \boldsymbol{\alpha}_k + \boldsymbol{\beta}_k\mathbf{d}_k, \tag{35}$$

where $\boldsymbol{\beta}_k = (\mathbf{q}_k^*\mathbf{q}_k)^{-1}\mathbf{q}_k^*\mathbf{b}_k$.

Proof. See Appendix A. □

It can be deduced from Proposition 1 that $\overline{\boldsymbol{\mu}}_k + \boldsymbol{\alpha}_k$ is referred to as the next iteration of $\overline{\boldsymbol{\mu}}_k$. We apply a heuristic update sequence for the penalty factor to achieve a stable solution. If the kth iterative solution $\mathbf{z}_k$ is closer to the feasible region than the previous solution $\mathbf{z}_{k-1}$, the penalty factor is decreased. Inversely, we increase the penalty factor when $\mathbf{z}_k$ is not closer to the feasible region.

The specific steps of solving by the augmented Lagrange method are given as follows.

Initialization: set $k = 1$, $\mathbf{z}_1 \in \mathscr{C}^K$, $\sigma_1 > 0$, $\tau \in (0, 1)$, and $\overline{\boldsymbol{\mu}}_1 = -(\mathbf{q}_1^*\mathbf{q}_1)^{-1}\mathbf{q}_1^*\nabla f_1$.

(1) Calculate $\boldsymbol{\alpha}_k$ and $\mathbf{d}_k$ in terms of (33) and (34). If $\|\mathbf{d}_k\| \simeq 0$, $\mathbf{z}_k$ is the KKT point of the problem (25); then stop iteration.

(2) We use Armijo search method to find the maximum of t_k satisfying

$$\mathbf{p}_{\sigma_k}\left(\mathbf{z}_k + t_k\mathbf{d}_k, \overline{\boldsymbol{\mu}}_k\right) \le \mathbf{p}_{\sigma_k}\left(\mathbf{z}_k, \overline{\boldsymbol{\mu}}_k\right) + \tau t_k \nabla \mathbf{p}^*_{\sigma_k}\left(\mathbf{z}_k, \overline{\boldsymbol{\mu}}_k\right)\mathbf{d}_k. \tag{36}$$

(3) Set $\mathbf{z}_{k+1} = \mathbf{z}_k + t_k\mathbf{d}_k$ and update the multiplier factor and penalty factor, respectively:

$$\overline{\boldsymbol{\mu}}_{k+1} = \overline{\boldsymbol{\mu}}_k + \boldsymbol{\alpha}_k,$$
$$\sigma_{k+1} = \begin{cases} 2\sigma_k & \text{if } \left\|\boldsymbol{\Pi}\left(z_{k+1}\right)\right\| > \left\|\boldsymbol{\Pi}\left(z_k\right)\right\|, \\ \dfrac{1}{2}\sigma_k & \text{if } \left\|\boldsymbol{\Pi}\left(z_{k+1}\right)\right\| \le \left\|\boldsymbol{\Pi}\left(z_k\right)\right\|. \end{cases} \tag{37}$$

(4) Set $k = k + 1$ and return to step (1).

5. Performance Analysis

5.1. Cramér-Rao Bound. In this subsection, the closed-form expression for the CRB of the proposed method with complex white Gaussian noise after the prewhitened process is illustrated. The bound of the noise variance estimation can be computed separately as $\mathrm{CRB}_n = 1/NM$ (see [35]). The remaining parameters consist of an unknown vector $\boldsymbol{\varphi} = \left[\widetilde{\boldsymbol{\theta}}^T \quad \mathbf{s}^T\right]^T$. It is not an easy task to get the CRB of the unknown parameters. However, fortunately, [36, 37] have provided a critical inspiration for the derivation in this paper.

First, the likelihood function is given by

$$L\left(\frac{\mathbf{y}}{\mathbf{s}\left(t_j\right)}, \widetilde{\boldsymbol{\theta}}\right) = \frac{1}{(2\pi)^{MN}(1/2)^{MN}} \cdot \exp\left\{-\sum_{j=1}^{M}\left(\mathbf{y}\left(t_j\right) - \mathbf{B}\boldsymbol{\Phi}\mathbf{A}\mathbf{s}\left(t_j\right)\right)^* \times \left(\mathbf{y}\left(t_j\right) - \mathbf{B}\boldsymbol{\Phi}\mathbf{A}\mathbf{s}\left(t_j\right)\right)\right\}. \tag{38}$$

Thus, the log-likelihood function is

$$\ln L\left(\frac{\mathbf{y}}{\mathbf{s}\left(t_j\right)}, \widetilde{\boldsymbol{\theta}}\right) = \text{const} - \sum_{j=1}^{M}\left(\mathbf{y}\left(t_j\right) - \mathbf{B}\boldsymbol{\Phi}\mathbf{A}\mathbf{s}\left(t_j\right)\right)^* \times \left(\mathbf{y}\left(t_j\right) - \mathbf{B}\boldsymbol{\Phi}\mathbf{A}\mathbf{s}\left(t_j\right)\right). \tag{39}$$

Then, the partial derivatives of (39) with respect to $\widetilde{\boldsymbol{\theta}}$, $\mathbf{s}_r(t_j) = \mathrm{Re}\{\mathbf{s}(t_j)\}$ and $\mathbf{s}_i(t_j) = \mathrm{Im}\{\mathbf{s}(t_j)\}$ are given by

$$\frac{\partial \ln L}{\partial \widetilde{\boldsymbol{\theta}}} = 2\sum_{j=1}^{M}\mathrm{Re}\left\{\mathbf{S}_j^*\mathbf{U}^*\boldsymbol{\Phi}^*\mathbf{B}^*\mathbf{n}\left(t_j\right)\right\},$$
$$\frac{\partial \ln L}{\partial \mathbf{s}_r\left(t_j\right)} = 2\,\mathrm{Re}\left\{\mathbf{A}^*\boldsymbol{\Phi}^*\mathbf{B}^*\mathbf{n}\left(t_j\right)\right\}, \tag{40}$$
$$\frac{\partial \ln L}{\partial \mathbf{s}_i\left(t_j\right)} = 2\,\mathrm{Im}\left\{\mathbf{A}^*\boldsymbol{\Phi}^*\mathbf{B}^*\mathbf{n}\left(t_j\right)\right\},$$

where $\mathbf{S}_j = \mathrm{diag}(\mathbf{s}(t_j))$ and $\mathbf{U} = [d\mathbf{a}(\widetilde{\theta}_1)/d\widetilde{\theta}_1 \quad d\mathbf{a}(\widetilde{\theta}_2)/d\widetilde{\theta}_2 \quad \cdots \quad d\mathbf{a}(\widetilde{\theta}_K)/d\widetilde{\theta}_K]$. Following [35, 37], we can obtain the Fisher information matrix (FIM) as follows:

$$\mathrm{FIM} = \boldsymbol{\Lambda} - \begin{bmatrix}\boldsymbol{\Delta}_{1r} & \boldsymbol{\Delta}_{1i} & \boldsymbol{\Delta}_{2r} & \cdots & \boldsymbol{\Delta}_{Mi} & \boldsymbol{\Delta}_{Mr}\end{bmatrix} \cdot \begin{bmatrix} \mathbf{G}_r & -\mathbf{G}_i & 0 & \cdots & 0 \\ \mathbf{G}_i & \mathbf{G}_r & & \ddots & \vdots \\ 0 & & \ddots & & 0 \\ \vdots & \ddots & & \mathbf{G}_r & -\mathbf{G}_i \\ 0 & \cdots & 0 & \mathbf{G}_i & \mathbf{G}_r \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\Delta}_{1r} \\ \boldsymbol{\Delta}_{1i} \\ \boldsymbol{\Delta}_{2r} \\ \vdots \\ \boldsymbol{\Delta}_{Mr} \\ \boldsymbol{\Delta}_{Mi} \end{bmatrix}, \tag{41}$$

where

$$E\left\{\frac{\partial \ln L}{\partial \widetilde{\boldsymbol{\theta}}}\left(\frac{\partial \ln L}{\partial \widetilde{\boldsymbol{\theta}}}\right)^T\right\} = \boldsymbol{\Lambda} = 2\sum_{j=1}^{M}\mathrm{Re}\left\{\mathbf{S}_j^*\mathbf{U}^*\boldsymbol{\Phi}^*\mathbf{B}^*\mathbf{B}\boldsymbol{\Phi}\mathbf{U}\mathbf{S}_j\right\},$$

$$E\left\{\frac{\partial \ln L}{\partial \mathbf{s}_r\left(t_k\right)}\left(\frac{\partial \ln L}{\partial \mathbf{s}_i\left(t_p\right)}\right)^T\right\} = -\mathbf{G}_i^{-1} = -\mathrm{Im}\left\{\mathbf{G}^{-1}\right\} = -\mathrm{Im}\left\{2\mathbf{A}^*\boldsymbol{\Phi}^*\mathbf{B}^*\mathbf{B}\boldsymbol{\Phi}\mathbf{A}\right\}\delta_{kp},$$

$$E\left\{\frac{\partial \ln L}{\partial \mathbf{s}_r\left(t_k\right)}\left(\frac{\partial \ln L}{\partial \mathbf{s}_r\left(t_p\right)}\right)^T\right\} = E\left\{\frac{\partial \ln L}{\partial \mathbf{s}_i\left(t_k\right)}\left(\frac{\partial \ln L}{\partial \mathbf{s}_i\left(t_p\right)}\right)^T\right\} = \mathbf{G}_r^{-1} = \mathrm{Re}\left\{\mathbf{G}^{-1}\right\} = \mathrm{Re}\left\{2\mathbf{A}^*\boldsymbol{\Phi}^*\mathbf{B}^*\mathbf{B}\boldsymbol{\Phi}\mathbf{A}\right\}\delta_{kp},$$

$$E\left\{\frac{\partial \ln L}{\partial \mathbf{s}_r\left(t_j\right)}\left(\frac{\partial \ln L}{\partial \widetilde{\boldsymbol{\theta}}}\right)^T\right\} = \mathrm{Re}\left\{\boldsymbol{\Delta}_j\right\} = \boldsymbol{\Delta}_{jr} = \mathrm{Re}\left\{2\mathbf{A}^*\boldsymbol{\Phi}^*\mathbf{B}^*\mathbf{B}\boldsymbol{\Phi}\mathbf{U}\mathbf{S}_j\right\}, \quad j = 1, 2, \ldots, M,$$

$$E\left\{\frac{\partial \ln L}{\partial \mathbf{s}_i\left(t_j\right)}\left(\frac{\partial \ln L}{\partial \widetilde{\boldsymbol{\theta}}}\right)^T\right\} = \operatorname{Im}\left\{\boldsymbol{\Delta}_j\right\} = \boldsymbol{\Delta}_{ji} = \operatorname{Im}\left\{2\mathbf{A}^*\boldsymbol{\Phi}^*\mathbf{B}^*\mathbf{B}\boldsymbol{\Phi}\mathbf{U}\mathbf{S}_j\right\}, \quad j = 1, 2, \ldots, M. \tag{42}$$

It is well known that

$$\begin{bmatrix}\boldsymbol{\Delta}_r^T & \boldsymbol{\Delta}_i^T\end{bmatrix}\begin{bmatrix}\mathbf{G}_r & -\mathbf{G}_i\\ \mathbf{G}_i & \mathbf{G}_r\end{bmatrix}\begin{bmatrix}\boldsymbol{\Delta}_r\\ \boldsymbol{\Delta}_i\end{bmatrix} = \operatorname{Re}\left\{\boldsymbol{\Delta}^*\mathbf{G}\boldsymbol{\Delta}\right\}. \tag{43}$$

It can be deduced from (41) and (43) that

$$\operatorname{CRB}\left(\widetilde{\boldsymbol{\theta}}\right) = \operatorname{FIM}^{-1} = \frac{1}{2}\left\{\sum_{j=1}^{M}\operatorname{Re}\left\{\mathbf{S}_j^*\mathbf{U}^*\mathbf{H}^*\left(\mathbf{I} - \mathbf{H}\mathbf{A}(\mathbf{H}\mathbf{A})^+\right)\mathbf{H}\mathbf{U}\mathbf{S}_j\right\}\right\}^{-1}, \tag{44}$$

where $\mathbf{H} = \mathbf{B}\boldsymbol{\Phi}$ is a $N \times P$ matrix and $(\cdot)^+$ denotes pseudoinverse. Note that the CRB in CS is affected not only by the conventional factors, for example, SNR, array structure and signal relative location, but also by the measurement matrix.

5.2. Performance Guarantee. In CS, the RIP has been deeply studied for the joint sparse recovery by minimizing the l_1 norm. We say that the matrix $\mathbf{C} \in \mathscr{C}^{N\times K}$ obeys the RIP of the order L if there exists a constant $\delta \in (0, 1)$ satisfying

$$(1-\delta)\|\mathbf{s}\|_2^2 \ll \|\mathbf{C}\mathbf{s}\|_2^2 \ll (1+\delta)\|\mathbf{s}\|_2^2. \tag{45}$$

Therefore, all submatrices of $\mathbf{C}$ with $\mathbf{L}$ columns are uniformly well conditioned. The restricted isometry constant (RIC) of the order $\mathbf{L}$, described as $\delta_L(\mathbf{C})$, is the smallest δ that satisfies (45) and $\delta_L(\mathbf{C})$ satisfies

$$\delta_L(\mathbf{C}) = \max_{|J'|=L}\left\|\mathbf{C}_{J'}^*\mathbf{C}_{J'} - \mathbf{I}_L\right\|, \tag{46}$$

where J' is a L-dimensional subsupport of $J = [1, 2, \ldots, K]$ and $\mathbf{C}_{J'}$ denotes the submatrix of $\mathbf{C}$ with columns indexed by J'. Note that the condition satisfying the RIP is so demanding that its applications are limited. Therefore, we should make use of a new version of the RIP, which is called the weak-1 RIP [38], to control the size of the recovery error. The weak-1 RIP is given in the following form:

$$(1-\alpha)\|\mathbf{v}\|_2^2 \ll \|\mathbf{C}\mathbf{v}\|_2^2 \ll (1+\alpha)\|\mathbf{v}\|_2^2 \tag{47}$$

for all $\mathbf{v}$ supported on U, where the cardinality of the set U is $L+1$. If the matrix $\mathbf{C}$ satisfies the weak-1 RIP, it can be deduced that $0 \le \alpha \le \phi_{L+1}(\mathbf{C}_U)$, where $\phi_k(\mathbf{C}_U)$ denotes the kth largest eigenvalue of $\mathbf{C}_U$. The corresponding weak-1 RIC is given by

$$\alpha_{L+1}^w(\mathbf{C}; U) = \min\phi_{L+1}\left(\mathbf{C}_U\right). \tag{48}$$

In this paper, when the estimation quality is imperfect, especially in the unfavorable case, the support is no longer identified by the algebraic property of $\mathbf{R}$. Hence, a new performance guarantee is given in terms of the weak-1 RIP in the following proposition.

Proposition 2. *One assumes that* $\mathbf{F} = \widehat{\mathbf{R}}^{1/2} + (1/2)\widehat{\mathbf{R}}^{-1/2}$ *and* $\mathbf{R}^{-1} = \boldsymbol{\Psi}\mathbf{R}_s^{-1}(\widetilde{\boldsymbol{\theta}})\boldsymbol{\Psi}^*$ *where* $\boldsymbol{\Psi} = [\psi_1\ \psi_2\ \ldots\ \psi_K]$. $\widehat{\boldsymbol{\theta}}$ *is an estimation that is asymptotically equal to* $\widetilde{\boldsymbol{\theta}}_s$ *such that* $|\operatorname{tr}(\mathbf{R}^{-1}(\widehat{\boldsymbol{\theta}})) - \operatorname{tr}(\mathbf{R}^{-1}(\widetilde{\boldsymbol{\theta}}_s))| \le \eta$ *for* $\eta \in (0, 1/2\|\mathbf{F}\|_2^2)$. *Let* J_0 *and* $J_1 = \{j_i\}_{i=1}^L$ *be the L-dimensional supports that consisted of the indexes of L largest elements in* $\widetilde{\boldsymbol{\theta}}_s$ *and* $\widehat{\boldsymbol{\theta}}$, *respectively. If the matrix* $\mathbf{R}$ *satisfies*

$$\alpha_{L+1}^w\left(\mathbf{R}; J_0\right) > \kappa \tag{49}$$

for

$$\kappa \ge \frac{L\|\mathbf{F}\|_2^2\sum_{j\in J_0}\boldsymbol{\psi}_j^*\boldsymbol{\psi}_j}{1 - 2\eta\|\mathbf{F}\|_2^2} \tag{50}$$

the support can be identified.

Proof. See Appendix B. □

It follows from Proposition 2 that the performance guarantee of the proposed method requires a mild condition in the unfavorable case. However, in the favorable case, the performance guarantee only requires a much milder condition, $\alpha_{L+1}^w(\mathbf{R}; J_0) > 0$, which is an algebraic condition.

6. Simulation Results

In this section, the performance of the proposed method is illustrated by several simulation results and compared with that of CS-MUISC, SPICE, and CRB under the condition that the number of sources is unknown. We consider the spatial signal impinging on the uniform linear array (ULA) with interspacing $\lambda/2$ where λ denotes the wavelength of source. In the ULA case, the steering vector corresponding to the DOA equal to θ_k is given by

$$\mathbf{a}\left(\theta_k\right) = \left[1\ \ e^{-j\pi\sin(\theta_k)}\ \ \cdots\ \ e^{-j(P-1)\pi\sin(\theta_k)}\right]^T, \tag{51}$$

where the number of the array elements is set to be $P = 8$.

In the simulation, the average root mean square error (RMSE) of the DOA estimation with 50 Monte Carlo runs is defined as the significant performance index:

$$\text{RMSE} = \left[\sum_{m=1}^{50}\sum_{l=1}^{L}\frac{\left(\overline{\theta}_{lm} - \theta_l\right)^2}{50L}\right]^{1/2}, \tag{52}$$

where $\overline{\theta}_{lm}$ is the estimate of θ_l in the mth run.

The resolution of the grid is closely related to the precision of the DOA estimation. A coarse grid can lead to poor precision, but a too fine grid increases computational complexity. Therefore, an adaptive grid refinement method is used to

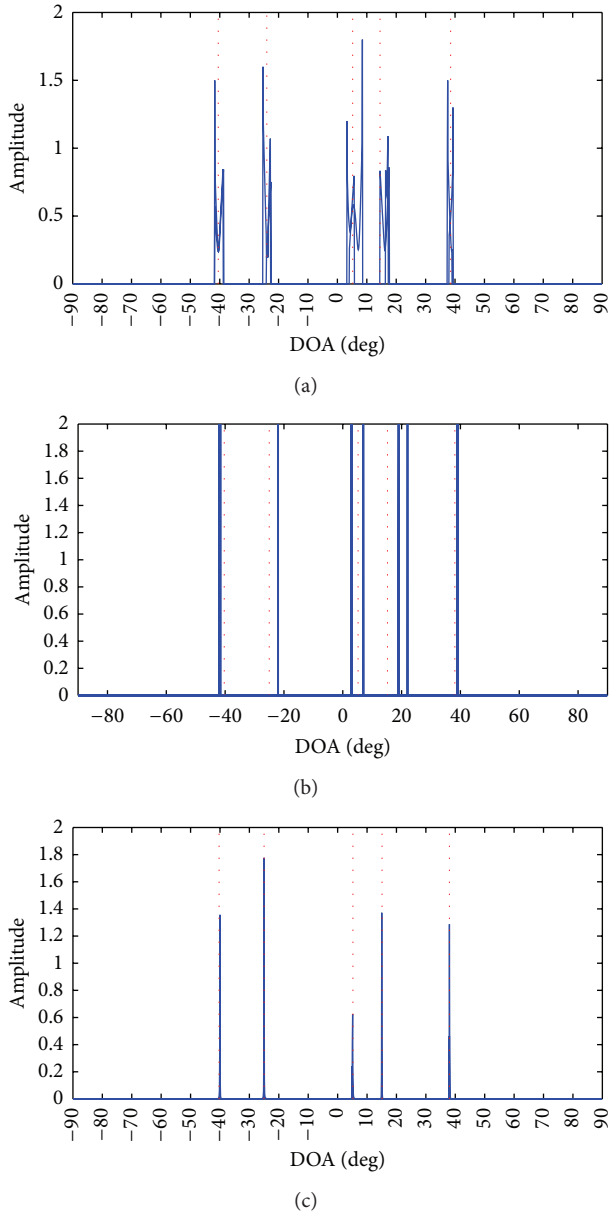


Figure 1: Superimposed spatial spectra of CS-MUSIC, SPICE, and AULMC in 10 Monte Carlo runs, where the red vertical dashed lines denote the true DOAs. (a) CS-MUSIC, (b) SPICE and (c) AULMC.

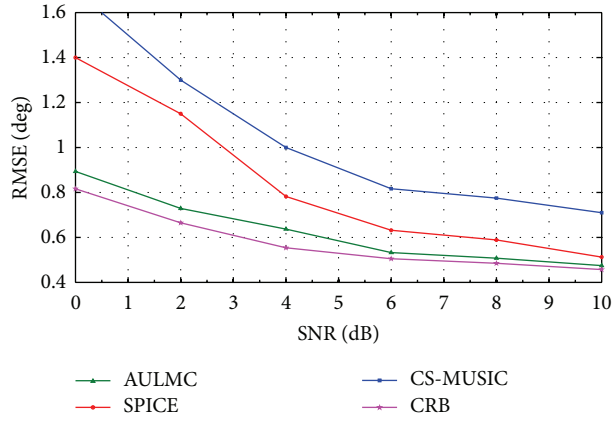


Figure 2: RMSE of the DOA estimation versus SNR for 50 snapshots.

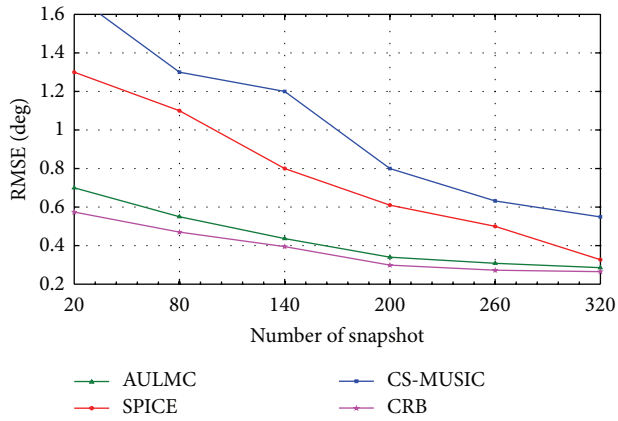


Figure 3: RMSE of the DOA estimation versus number of snapshots for 5 dB SNR.

balance the tradeoff between precision and computational complexity. In the simulation, we make a coarse grid with 1° step in the range of −90° to 90° and perform a local fine grid in the vicinity of locations obtained by using the coarse grid.

In the first simulation, we display the superimposed spatial spectra of three algorithms in 10 Monte Carlo runs in the scenario with low SNR, small number of snapshots, and five sources impinging from $[-40.3^\circ\ -25^\circ\ 5.2^\circ\ 15.2^\circ\ 38.1^\circ]$, respectively where two most closely spaced sources are correlated and the remaining sources are uncorrelated. The spatial spectra are shown in Figure 1 with 3 dB SNR and 50 snapshots. The following facts can be acquired from Figure 1 as follows: the spatial spectrum obtained by CS-MUSIC suffers from severe interference at the true directions, especially at two most closely spaced correlated sources whose bias is clearly seen from the insert. Two most closely spaced correlated sources can be resolved by SPICE (note that peaks in the spatial spectrum are much larger than 2, but they are cut off at 2 to use the same scale as the other figures in Figure 1), but SPICE can yield false peaks and slight bias in the vicinity of the correlated sources and uncorrelated sources, respectively. The proposed method AULMC yields a nearly ideal spatial spectrum and provides a precise estimation for all the sources. In summary, AULMC outperforms CS-MUSIC and SPICE in terms of the spatial spectrum in the scenario with low SNR, small number of snapshots, and closely spaced correlated sources.

We analyze the RMSE of three algorithms under different conditions in the second simulation. The source model is the same as the first simulation. Figure 2 shows the RMSE as a function of SNR of all the algorithms and CRB in 50 Monte Carlo runs for the fixed number of snapshots 50, whereas the RMSE versus number of snapshots is shown in Figure 3 for the fixed SNR 5 dB in 50 Monte Carlo runs. Based on Figures 2

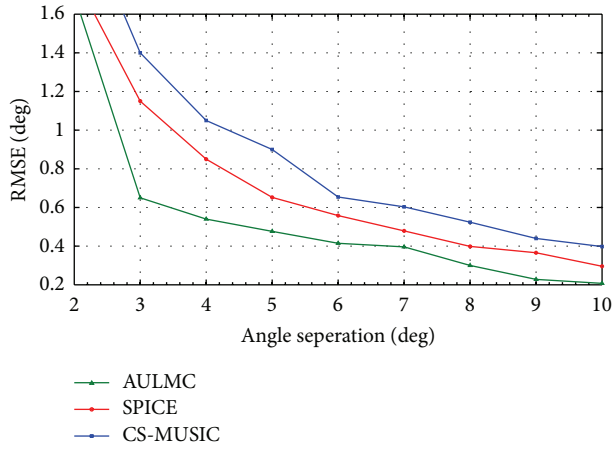


Figure 4: RMSE of the DOA estimation versus angle separation, where the SNR is 13 dB and the number of snapshots is 100.

Table 1: Comparison of computation time.

Number of snapshots	Time (sec)			
	AULMC	CS-MUSIC	SPICE	SDP solver
50	0.0651	0.0569	0.0601	0.7526
75	0.0776	0.0604	0.0714	0.8871
100	0.0860	0.0640	0.0819	1.0268
125	0.0974	0.0691	0.0923	1.3018

and 3, we can draw the conclusions that the RMSE of AULMC is smaller than those of other two algorithms and AULMC has the more significant performance advantages than the other two algorithms, especially in the scenarios with low SNR or small number of snapshots. One possible explanation is that AULMC can give the stable estimation in every Monte Carlo run. It can be also seen that the RMSE is close to the CRB with the increase of SNR and the number of snapshots.

In Figure 4, we display the relation between the RMSE and angel separation of correlated sources which can illustrate the resolving ability. Let two correlated sources at angles 20° and $20^\circ+\Delta\theta$, where the step of $\Delta\theta$ is 1°, be impinged on the ULA. The SNR is 13 dB and the number of snapshots is 100. It can be seen from Figure 4 that when angle separation is 2°, AULMC fails; however, AULMC can still provide a precise estimation as long as the angle separation is no less than 3° and has higher resolution than the other two algorithms.

Finally, the computation time of different algorithms versus number of snapshots is shown in Table 1 for comparing the efficiency of these algorithms and SDP solver. Two correlated sources impinge on the ULA at 20° and 25°. The SNR is fixed at 13 dB. The computation time is obtained by the MATLAB 7.8 (R2009a) on a 2.8 GHz 4 GB PC. For AULMC, the computation time is mainly spent on the iterations of augmented Lagrange.

It can be seen from Table 1 that the computation time of SDP solver is the longest, and although the computation time of AULMC is longer than that of other two algorithms, it is comparable. Moreover, it is worth noting that the performance of AULMC is much better than that of CS-MUSIC or SPICE.

7. Conclusion

A novel augmented Lagrange based on modified covariance matching criterion method for DOA estimation is proposed in CS. It is proved that the problem of minimizing the modified covariance matching criterion is an SDP, which can be transformed into the constrained quadratic programming problem solved by the augmented Lagrange method. A detailed derivation for the CRB and a theoretical performance guarantee for identifying the support are provided. Simulation results show that AULMC outperforms CS-MUSIC and SPICE in terms of the spatial spectrum and has more precise estimation as well as higher resolution, especially in the scenarios with low SNR, small number of snapshots, and closely spaced correlated sources.

Appendices

A. Proof of Proposition 1

Due to $\boldsymbol{\mu}(\mathbf{z}_k) = -(\mathbf{q}_k^*\mathbf{q}_k)^{-1}\mathbf{q}_k^*\nabla f_k$, we have

$$\mathbf{q}_k^*\mathbf{q}_k\boldsymbol{\mu}(\mathbf{z}_k) = -\mathbf{q}_k^*\nabla f_k. \tag{A.1}$$

With (34) and (A.1), we get

$$\mathbf{q}_k^*\mathbf{q}_k\left[\boldsymbol{\mu}(\mathbf{z}_k) - \overline{\boldsymbol{\mu}}_k - \boldsymbol{\alpha}_k\right] = \mathbf{q}_k^*\mathbf{b}_k\mathbf{d}_k. \tag{A.2}$$

Therefore, it is implied by (A.2) that (35) holds and the proof of Proposition 1 is completed.

B. Proof of Proposition 2

The support J_0 can be identified if

$$\max_{j\in J_0}\frac{\mathrm{Tr}\left[\mathbf{F}\mathbf{R}^{-1/2}\left(\widehat{\boldsymbol{\theta}}\right)\mathbf{F}\mathbf{R}^{-1/2}\left(\widehat{\boldsymbol{\theta}}\right)\right]}{\mathrm{Tr}\left[\mathbf{F}^2\right]} < \min_{j\in J\setminus J_0}\frac{\mathrm{Tr}\left[\mathbf{F}\mathbf{R}^{-1/2}\left(\widehat{\boldsymbol{\theta}}\right)\mathbf{F}\mathbf{R}^{-1/2}\left(\widehat{\boldsymbol{\theta}}\right)\right]}{\mathrm{Tr}\left[\mathbf{F}^2\right]}. \tag{B.1}$$

By the Cauchy-Schwarz inequality of the trace, we have

$$\begin{aligned}&\left|\frac{\mathrm{Tr}\left[\left(\mathbf{F}\mathbf{R}^{-1/2}\left(\widehat{\boldsymbol{\theta}}\right)\right)^2\right] - \mathrm{Tr}\left[\left(\mathbf{F}\mathbf{R}^{-1/2}\left(\widetilde{\boldsymbol{\theta}}_s\right)\right)^2\right]}{\mathrm{Tr}\left[\mathbf{F}^2\right]}\right| \\ &\quad\le \frac{\mathrm{Tr}\left[\mathbf{F}^2\right]\cdot\left|\mathrm{Tr}\left[\mathbf{R}^{-1}\left(\widehat{\boldsymbol{\theta}}\right)\right] - \mathrm{Tr}\left[\mathbf{R}^{-1}\left(\widetilde{\boldsymbol{\theta}}_s\right)\right]\right|}{\mathrm{Tr}\left[\mathbf{F}^2\right]} \\ &\quad= \left|\mathrm{Tr}\left[\mathbf{R}^{-1}\left(\widehat{\boldsymbol{\theta}}\right)\right] - \mathrm{Tr}\left[\mathbf{R}^{-1}\left(\widetilde{\boldsymbol{\theta}}_s\right)\right]\right| \le \eta.\end{aligned} \tag{B.2}$$

Then, for all $j \in J_0$ we have

$$\begin{aligned}\frac{\mathrm{Tr}\left[\left(\mathbf{F}\mathbf{R}^{-1/2}\left(\widehat{\boldsymbol{\theta}}\right)\right)^2\right]}{\mathrm{Tr}\left[\mathbf{F}^2\right]} &\le \eta + \frac{\mathrm{Tr}\left[\left(\mathbf{F}\mathbf{R}^{-1/2}\left(\widetilde{\boldsymbol{\theta}}_s\right)\right)^2\right]}{\mathrm{Tr}\left[\mathbf{F}^2\right]} \\ &\le \eta + \frac{\mathrm{Tr}\left[\mathbf{F}^2\right]\mathrm{Tr}\left[\mathbf{R}^{-1}\left(\widetilde{\boldsymbol{\theta}}_s\right)\right]}{\mathrm{Tr}\left[\mathbf{F}^2\right]} \\ &= \eta + \frac{\sum_{j\in J_0}\boldsymbol{\psi}_j^*\boldsymbol{\psi}_j}{\widetilde{\theta}_j} \\ &\le \eta + \frac{L\sum_{j\in J_0}\boldsymbol{\psi}_j^*\boldsymbol{\psi}_j}{\alpha_{L+1}^w\left(\mathbf{R}; J_0\right)}.\end{aligned} \tag{B.3}$$

Thus, an upper bound on the left-band side of (B.1) is given by

$$\max_{j\in J_0}\frac{\mathrm{Tr}\left[\left(\mathbf{F}\mathbf{R}^{-1/2}\left(\widehat{\boldsymbol{\theta}}\right)\right)^2\right]}{\mathrm{Tr}\left[\mathbf{F}^2\right]} \le \eta + \frac{L\sum_{j\in J_0}\boldsymbol{\psi}_j^*\boldsymbol{\psi}_j}{\alpha_{L+1}^w\left(\mathbf{R}; J_0\right)}. \tag{B.4}$$

Similarly, we can obtain

$$\begin{aligned}\frac{\mathrm{Tr}\left[\left(\mathbf{F}\mathbf{R}^{-1/2}\left(\widehat{\boldsymbol{\theta}}\right)\right)^2\right]}{\mathrm{Tr}\left[\mathbf{F}^2\right]} &\ge \frac{\mathrm{Tr}\left[\mathbf{F}^2\mathbf{R}^{-1}\left(\widetilde{\boldsymbol{\theta}}_s\right)\right]}{\mathrm{Tr}\left[\mathbf{F}^2\right]} - \eta \\ &= \frac{\sum_{j\in J\setminus J_0}\boldsymbol{\psi}_j^*\mathbf{F}^2\boldsymbol{\psi}_j/\widetilde{\theta}_j}{\mathrm{Tr}\left[\mathbf{F}^2\right]} - \eta \\ &\ge \frac{1}{\|\mathbf{F}\|_2^2} - \eta\end{aligned} \tag{B.5}$$

for all $j \in J \setminus J_0$ and hence describe a lower bound on the right-hand side of (B.1) as

$$\min_{j\in J\setminus J_0}\frac{\mathrm{Tr}\left[\left(\mathbf{F}\mathbf{R}^{-1/2}\left(\widehat{\boldsymbol{\theta}}\right)\right)^2\right]}{\mathrm{Tr}\left[\mathbf{F}^2\right]} \ge \frac{1}{\|\mathbf{F}\|_2^2} - \eta. \tag{B.6}$$

Combining (B.4) with (B.6), the support J_0 can be identified if $\mathbf{R}$ satisfies

$$\alpha_{L+1}^w\left(\mathbf{R}; J_0\right) > \kappa \tag{B.7}$$

for

$$\kappa \ge \frac{L\|\mathbf{F}\|_2^2\sum_{j\in J_0}\boldsymbol{\psi}_j^*\boldsymbol{\psi}_j}{1 - 2\eta\|\mathbf{F}\|_2^2}. \tag{B.8}$$

Conflict of Interests

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

Acknowledgment

This work was supported in part by the National Science Foundation of China under Grant 61201410 and in part by Fundamental Research Funds for the Central Universities under Grand no. HEUCF130804.

References

[1] J. D. Dai, W. C. Xu, and D. Zhao, "Real-valued DOA estimation for uniform linear array with unknown mutual coupling," *Signal Processing*, vol. 92, no. 9, pp. 2056–2065, 2012.

[2] L. Bai, C.-Y. Peng, and S. Biswas, "Association of DOA estimation from two ULAs," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 6, pp. 1094–1101, 2008.

[3] J. Capon, "High-resolution frequency-wavenumber spectrum analysis," *Proceedings of IEEE*, vol. 57, pp. 1408–1418, 1969.

[4] J. P. Burg, "Maximum entropy spectral analysis," in *Proceedings of the 37th Annual International SEG meeting Oklahoma City*, 1967.

[5] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[6] R. Roy and T. Kailath, "ESPRIT—estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984–995, 1989.

[7] B. D. Rao and K. V. S. Hari, "Performance analysis of root-music," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 12, pp. 1939–1949, 1989.

[8] R. Bachl, "Forward-backward averaging technique applied to TLS-ESPRIT processing," *IEEE Transactions on Signal Processing*, vol. 43, no. 11, pp. 2691–2699, 1995.

[9] Z. Sun and Z. Yang, "Study of nonlinear parameter identification using UKF and Maximum Likelihood method," in *Proceedings of the IEEE International Conference on Control Applications (CCA '10)*, pp. 671–676, September 2010.

[10] M. Viberg, B. Ottersten, and T. Kailath, "Detection and estimation in sensor arrays using weighted subspace fitting," *IEEE Transactions on Signal Processing*, vol. 39, no. 11, pp. 2436–2449, 1991.

[11] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[12] E. T. Northardt, I. Bilik, and Y. I. Abramovich, "Spatial compressive sensing for direction-of-arrival estimation with bias mitigation via expected likelihood," *IEEE Transactions on Signal Processing* , vol. 61, no. 5, pp. 1183–1195, 2013.

[13] M. Carlin, P. Rocca, G. Oliveri, F. Viani, and A. Massa, "Direciton-ofarrival estimation through bayesian compressive sensing strategies," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 7, pp. 3828–3838, 2013.

[14] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[15] A. C. Gürbüz, J. H. McClellan, and V. Cevher, "A compressive beamforming method," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '08)*, pp. 2617–2620, April 2008.

[16] Y. Wang, G. Leus, and A. Pandharipande, "Direction estimation using compressive sampling array processing," in *Proceedings of the IEEE/SP 15th Workshop on Statistical Signal Processing (SSP '09)*, pp. 626–629, September 2009.

[17] P. Stoica, P. Babu, and J. Li, "New method of sparse parameter estimation in separable models and its use for spectral analysis of irregularly sampled data," *IEEE Transactions on Signal Processing*, vol. 58, no. 1, pp. 35–47, 2010.

[18] P. Stoica and P. Babu, "SPICE and LIKES: two hyperparameter-free methods for sparse-parameter estimation," *Signal Processing*, vol. 92, no. 7, pp. 1580–1590, 2012.

[19] D. Malioutov, M. Çetin, and A. S. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 3010–3022, 2005.

[20] J. M. Kim, O. K. Lee, and J. C. Ye, "Compressive MUSIC: revisiting the link between compressive sensing and array signal processing," *IEEE Transactions on Information Theory*, vol. 58, no. 1, pp. 278–301, 2012.

[21] K. Lee, Y. Bresler, and M. Junge, "Subspace methods for Joint sparse recovery," *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3613–3640, 2012.

[22] M. A. Davenport and M. B. Wakin, "Analysis of orthogonal matching pursuit using the restricted isometry property," *IEEE Transactions on Information Theory*, vol. 56, no. 9, pp. 4395–4401, 2010.

[23] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Algorithms for simultaneous sparse approximation. Part I: greedy pursuit," *Signal Processing*, vol. 86, no. 3, pp. 572–588, 2006.

[24] J. A. Tropp, "Algorithms for simultaneous sparse approximation. Part II: convex relaxation," *Signal Processing*, vol. 86, no. 3, pp. 589–602, 2006.

[25] P. Feng and Y. Bresler, "Spectrum-blind minimum-rate sampling and reconstruction of multiband signals," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '96)*, vol. 3, pp. 1688–1691, May 1996.

[26] P. Stoica, P. Babu, and J. Li, "SPICE: a sparse covariance-based estimation method for array processing," *IEEE Transactions on Signal Processing*, vol. 59, no. 2, pp. 629–638, 2011.

[27] S. Yin, H. Luo, and S. Ding, "Real-time implementation of fault-tolerant control systems with performance optimization," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 2402–2411, 2014.

[28] S. Yin, G. Wang, and H. Karimi, "Data-driven design of robust fault detection system for wind turbines," *Mechatronics*, 2013.

[29] S. Yin, S. Ding, A. Haghani, and H. Hao, "Data-driven monitoring for stochastic systems and its application on batch process," *International Journal of Systems Science*, vol. 44, no. 7, pp. 1366–1376, 2013.

[30] S. Yin, S. Ding, A. Haghani, H. Hao, and P. Zhang, "A comparison study of basic datadriven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process," *Journal of Process Control*, vol. 22, no. 9, pp. 1567–1581, 2012.

[31] B. Ottersten, P. Stoica, and R. Roy, "Covariance matching estimation techniques for array signal processing applications," *Digital Signal Processing*, vol. 8, no. 3, pp. 185–210, 1998.

[32] J. Yin and T. Chen, "Direction-of-arrival estimation using a sparse representation of array covariance vectors," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4489–4493, 2011.

[33] D. I. Svergun, "Determination of the regularization parameter in indirect-transform methods using perceptual criteria," *Journal of Applied Crystallography*, vol. 25, no. 4, pp. 495–503, 1992.

[34] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, Society for Industrial and Applied Mathematics, 1994.

[35] A. Leshem and A.-J. D. van Veen, "Direction-of-arrival estimation for constant modulus signals," *IEEE Transactions on Signal Processing*, vol. 47, no. 11, pp. 3125–3129, 1999.

[36] P. Stoica, E. G. Larsson, and A. B. Gershman, "The stochastic CRB for array processing: a textbook derivation," *IEEE Signal Processing Letters*, vol. 8, no. 5, pp. 148–150, 2001.

[37] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and Cramer-Rao bound," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 5, pp. 720–741, 1989.

[38] E. J. Candès and Y. Plan, "A probabilistic and RIPless theory of compressed sensing," *IEEE Transactions on Information Theory*, vol. 57, no. 11, pp. 7235–7254, 2011.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 589675, 10 pages
http://dx.doi.org/10.1155/2013/589675

Research Article

Beamforming of Joint Polarization-Space Matched Filtering for Conformal Array

Lutao Liu,[1] Yilin Jiang,[1] Liangtian Wan,[1] and Zuoxi Tian[2]

[1] *Department of Information and Communication Engineering, University of Harbin Engineering, Harbin 150001, China*
[2] *Science and Technology on Underwater Test and Control Laboratory, Dalian 116013, China*

Correspondence should be addressed to Yilin Jiang; jiangyilin80@163.com

Received 16 September 2013; Accepted 17 November 2013

Academic Editors: H. R. Karimi, X. Yang, Z. Yu, and W. Zhang

Due to the polarization mismatch of the antenna, the received signal suffers from energy loss. The conventional beamforming algorithms could not be applied to the conformal array because of the varying curvature. In order to overcome the energy loss of the received signal, a novel joint polarization-space matched filtering algorithm for cylindrical conformal array is proposed. First, the snapshot data model of the conformal polarization sensitive array is analyzed. Second, the analytical expression of polarization sensitive array beamforming is derived. Linearly constrained minimum variance (LCMV) beamforming technique is facilitated for the cylindrical conformal array. Third, the idea of joint polarization-space matched filtering is presented, and the principle of joint polarization-space matched filtering is discussed in detail. Theoretical analysis and computer simulation results verify that the conformal polarization sensitive array is more robust than the ordinary conformal array. The proposed algorithm can improve the performance when signal and interference are too close. It can enhance the signal-to-noise ratio (SNR) by adjusting the polarization of the elements of the conformal array, which matches the polarization of the incident signal.

1. Introduction

Conformal array is defined as a set of sensors or elements mounted on the curved surface [1]. Comparing with the ordinary array, it has obvious advantages, for example, a large coverage of angle with good radiation pattern characteristics, reduction of aerodynamic drag, space savings, potential increase in available aperture, and reduction or elimination of random included preset errors [2]. It can be used in a variety of areas such as radar, sonar, biomedical imaging, and wireless communications [3].

The analysis of conformal array is difficult because the structure of the array is too complicated. Generally, the element pattern is defined in the local coordinate. Unlike the ordinary arrays containing elements of the same pattern, a conformal array can have elements of different patterns. The Euler rotation transformation [4–6] is adopted to transform the parameter from the global coordinate to local coordinate, which is very effective for the pattern transformation.

The multiple parameter estimation technique for array antenna has been widely applied in many fields. The estimation of signal parameters via rotational invariance techniques (ESPRIT) was used for two-dimensional angle and polarization estimation [7]. A novel algorithm to estimate the direction of arrival (DOA) and the polarization of a completely polarized polynomial-phase signal of an arbitrary degree with a single polarized vector-sensor was proposed in [8]. The algorithm can estimate DOA and polarization parameter without a priori knowledge of the polynomial-phase signal's coefficients and the signal's frequency spectrum. An algorithm for joint high-resolution DOA and polarization estimation using real-world arrays was proposed, which was suitable for correlated and coherent signals [9]. Cramer-Rao lower bound (CRB) for DOA and polarization estimation with arbitrary multiport antennas was derived in [10]. It can be used to compare different algorithms. The maximum likelihood (ML) and minimum-variance distortionless response (MVDR) estimators for DOA and polarization parameters were derived in a multipath environment [11]. The polarization sensitive array usually consists of multicomponent polarization sensitive antenna, which is arranged in the space under certain order. The advantage of the polarization sensitive array is that when the multiple polarization signals

cannot be distinguished in the space domain, they can be distinguished in the polarization domain by adjusting the polarization information of the different signals.

At present, there are not many parameter estimation algorithms for conformal array. The multiple signal classification (MUSIC) algorithm and subarray technique were adopted in [12, 13], but the interpolation error always exists and the computational complexity is tremendous. A blind parameter estimation algorithm using ESPRIT was proposed for DOA estimation of conformal array [14]. The ESPRIT algorithm based on polarization sensitive array to estimate DOA and polarization parameter was proposed for conformal array [15]. The MVDR beamforming for cylindrical conformal array was investigated in [2]. A general and systematic method for pattern analysis of conformal arrays was presented using geometric algebra [16]. A beamforming approach for conformal array based on geometric algebra was proposed in [17]. However, the algorithms in [2, 16, 17] are only limited to space filtering. The joint polarization-space matched filtering for conformal array has not been researched extensively thus far.

The tradition matched filtering algorithm is based on spatial filtering, but the polarization information is not utilized. The matched filtering performance could be improved by exploiting the polarization information. The contribution of this paper is that a novel joint polarization-space matched filtering for cylindrical conformal array is proposed. The principle of the adaptive filtering is illustrated. The LCMV beamforming for conformal polarization sensitive array is analyzed. The signal-to-noise ratio (SNR) would increase by adjusting the polarization of the elements of the conformal array, which matches the polarization of the incident signal. In the future work, we will focus on the application of the proposed algorithm [18–21].

The organization of this paper is structured as follows. Section 2 introduces the mathematical data model of the conformal array. Section 3 contains the core contribution of this paper, which is the principle of joint polarization-space matched filtering based on LCMV beamforming. The analytical expression of polarization sensitive array beamforming is derived. Section 4 presents the simulation result. Section 5 summarizes our conclusions.

2. Mathematical Data Model

2.1. The Array Configuration. Unlike ordinary circular arrays, conformal array has the shadow effect attribute to the metal cylinder, which means that for an incident signal from a special angle, not all of the antenna elements can receive this signal. The sub-array divided technique proposed in [13] is adopted in this paper. Four subarrays are divided evenly, and each sub-array is in charge of azimuth coverage of 90°. The parameters estimation results of four sub-arrays are synthesized in order to estimate the incident signals parameters for the scope of the entire space.

Consider a simple $D \times E$ cylindrical conformal array which is shown in Figure 1. There are E uniformly spaced rings on the surface of the cylinder perpendicular to the axis of symmetry. On each ring we have D orthogonal electric dipole pairs uniformly spaced. The radius of the cylinder is R. The distance between rings is L and the global coordinate is placed at the axis of the midpoint of the first ring. The angle between two consecutive antennas on a given ring is α. Where $\mathbf{P}_{mn}$ is the position vector of the (m, n)th element.

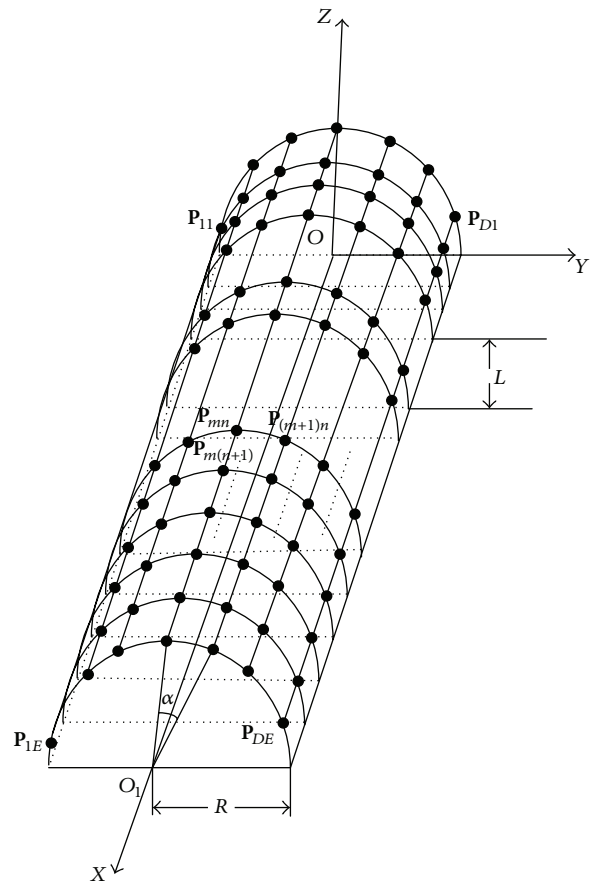


Figure 1: $D \times E$ cylindrical conformal array.

As shown in Figure 2, the (m, n)th element pattern is designed in element local Cartesian coordinate (X_{mn}, Y_{mn}, Z_{mn}), and (X, Y, Z) represents the global Cartesian coordinate. The X_{mn}-axis of element local Cartesian coordinate and the X-axis of the global Cartesian coordinate are parallel. μ is the angle between the Z_{mn}-axis and Y-axis. The Cartesian coordinate of the unit vector with elevation θ and azimuth φ in the global coordinate can be expressed as

$$\mathbf{x} = \sin\theta\cos\varphi, \qquad \mathbf{y} = \sin\theta\sin\varphi, \qquad \mathbf{z} = \cos\theta. \tag{1}$$

With the array configuration and the coordinate design, the element pattern from the global coordinate to the local coordinate can be transformed as

$$\begin{aligned} \mathbf{x}_{mn} &= \mathbf{x}, \\ \mathbf{y}_{mn} &= \mathbf{y}\sin(\mu) - \mathbf{z}\cos(\mu), \\ \mathbf{z}_{mn} &= \mathbf{y}\cos(\mu) - \mathbf{z}\sin(\mu). \end{aligned} \tag{2}$$

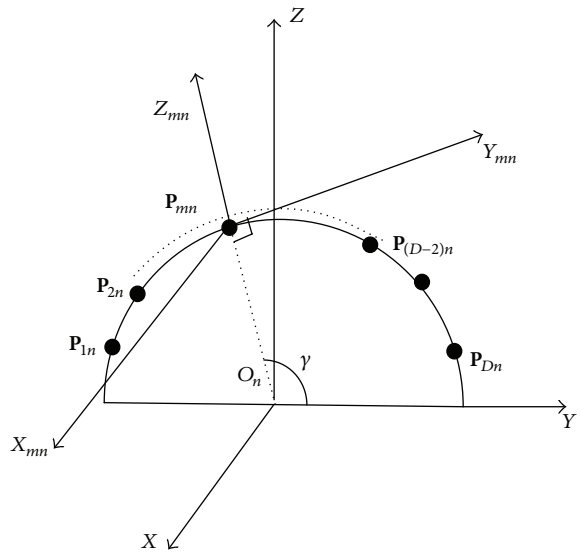


Figure 2: The local coordinate systems on the nth ring.

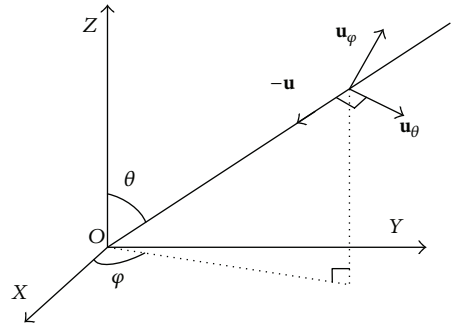


Figure 3: The direction vector $\mathbf{u}$ of the incident signal.

The Cartesian coordinate of the unit vector with elevation θ and azimuth φ in the local coordinates can be expressed as

$$\varphi_{mn} = \begin{cases} \arctan\left(\dfrac{\mathbf{y}_{mn}}{\mathbf{x}_{mn}}\right), & \mathbf{x}_{mn} > 0, \\ \pi + \arctan\left(\dfrac{\mathbf{y}_{mn}}{\mathbf{x}_{mn}}\right), & \mathbf{x}_{mn} < 0, \end{cases} \tag{3}$$

$$\theta_{mn} = \arccos\left(\mathbf{z}_{mn}\right).$$

In order to avoid the condition shown in (3), the global coordinate is placed at the axis of the midpoint of the first ring. The transformation is simplified because $\mathbf{x}_{mn} = \mathbf{x} > 0$.

2.2. The Data Model Based on Polarization Sensitive Array. As shown in Figure 3, a completely polarized wave $\mathbf{u}$ is a narrowband far field source impinging on the array with elevation θ and azimuth φ. The definition and design of the element pattern are always in local coordinate; thus the patterns of each element in plane array are identical. However, for the conformal antenna array, the pattern of each element has different directions because of the varying curvature of conformal carrier. The modeling approach of steering vector based on plane array can no longer be used in this case. The new modeling approach of steering vector based on conformal antenna array is introduced as follows.

In global coordinate as shown in Figure 1, the steering vector of the cylindrical conformal array can be represented as

$$\mathbf{a}(\theta, \varphi, \gamma, \eta) = \left[\mathbf{r}_1 \exp\left(-j2\pi\mathbf{P}_1 \cdot \frac{\mathbf{u}}{\lambda}\right), \mathbf{r}_2 \exp\left(-j2\pi\mathbf{P}_2 \cdot \frac{\mathbf{u}}{\lambda}\right), \ldots, \mathbf{r}_{DE} \exp\left(-j2\pi\mathbf{P}_{DE} \cdot \frac{\mathbf{u}}{\lambda}\right)\right]^T, \tag{4}$$

$$\mathbf{u} = \sin(\theta)\cos(\varphi)\,\mathbf{x} + \sin(\theta)\sin(\varphi)\,\mathbf{y} + \cos(\theta)\,\mathbf{z}, \tag{5}$$

where $0 \le \gamma < \pi/2$ and $-\pi \le \eta < \pi$ represent the auxiliary polarization angle and the polarization phase difference, respectively. $\mathbf{r}_{mn}$ is the response of the (m, n)th dipole to the unit strength incident signal. $\mathbf{P}_{mn}$ is the distance vector of the (m, n)th dipole which is in the global coordinate. λ is the wavelength of the incident signal.

The electric field vector in the global coordinates can be expressed as

$$\mathbf{E} = E_\theta \mathbf{u}_\theta + E_\varphi \mathbf{u}_\varphi,$$

$$\mathbf{u}_\theta = \cos(\theta)\cos(\varphi)\,\mathbf{x} + \cos(\theta)\sin(\varphi)\,\mathbf{y} - \sin(\theta)\,\mathbf{z}, \tag{6}$$

$$\mathbf{u}_\varphi = -\sin(\varphi)\,\mathbf{x} + \cos(\varphi)\,\mathbf{y},$$

where $\mathbf{u}_\theta$ and $\mathbf{u}_\varphi$ are the orthogonal unit component direction vectors of electric field. $E_\theta = \sin\gamma \exp(j\eta)$ and $E_\varphi = \cos\gamma$ are the component vectors by which the electric field vector is projected onto the $\mathbf{u}_\theta$ and $\mathbf{u}_\varphi$, respectively.

In the local coordinate system as shown in Figure 2, the direction vector of the incident signal $\mathbf{u}'$ and the electric field vector are given by

$$\mathbf{u}' = \sin\left(\theta'_{mn}\right)\cos\left(\varphi'_{mn}\right)\mathbf{x} + \sin\left(\theta'_{mn}\right)\sin\left(\varphi'_{mn}\right)\mathbf{y} + \cos\left(\theta'_{mn}\right)\mathbf{z}, \tag{7}$$

$$\mathbf{E}'_{mn} = E_{\theta'_{mn}}\mathbf{u}_{\theta'_{mn}} + E_{\varphi'_{mn}}\mathbf{u}_{\varphi'_{mn}}, \tag{8}$$

$$\mathbf{u}_\theta = \cos\left(\theta'_{mn}\right)\cos\left(\varphi'_{mn}\right)\mathbf{x} + \cos\left(\theta'_{mn}\right)\sin\left(\varphi'_{mn}\right)\mathbf{y} - \sin\left(\theta'_{mn}\right)\mathbf{z}, \tag{9}$$

$$\mathbf{u}_\varphi = -\sin\left(\varphi'_{mn}\right)\mathbf{x} + \cos\left(\varphi'_{mn}\right)\mathbf{y},$$

where $\mathbf{u}_{\theta'_{mn}}$ and $\mathbf{u}_{\varphi'_{mn}}$ are the orthogonal unit component direction vectors of electric field in local coordinate. $E_{\theta'_{mn}}$ and $E_{\varphi'_{mn}}$ are the component vectors by which the electric field vector is projected onto $\mathbf{u}_{\theta'_{mn}}$ and $\mathbf{u}_{\varphi'_{mn}}$, respectively. θ'_{mn} and

φ'_{mn} are the signal's elevation and azimuth in the (m,n)th dipole's local coordinate, respectively.

Taking (2) into (9), then $E_{\theta'_{mn}}$ and $E_{\varphi'_{mn}}$ are projected onto $\mathbf{u}_\theta$ and $\mathbf{u}_\varphi$, respectively:

$$\begin{aligned}
\mathbf{u}_{\theta\theta'_{mn}} &= \mathbf{u}_\theta \cdot \mathbf{u}_{\theta'_{mn}},\\
\mathbf{u}_{\varphi\theta'_{mn}} &= \mathbf{u}_\varphi \cdot \mathbf{u}_{\theta'_{mn}},\\
\mathbf{u}_{\theta\varphi'_{mn}} &= \mathbf{u}_\theta \cdot \mathbf{u}_{\varphi'_{mn}},\\
\mathbf{u}_{\varphi\varphi'_{mn}} &= \mathbf{u}_\varphi \cdot \mathbf{u}_{\varphi'_{mn}}.
\end{aligned} \tag{10}$$

In local coordinate system, the electric field $\mathbf{E}'_{mn}$ vector is represented as

$$\begin{aligned}
\mathbf{E}'_{mn} &= E_{\theta'_{mn}}\mathbf{u}_{\theta'_{mn}} + E_{\varphi'_{mn}}\mathbf{u}_{\varphi'_{mn}}\\
&= \left(E_{\theta\theta'_{mn}} + E_{\varphi\theta'_{mn}}\right)\mathbf{u}_{\theta'_{mn}} + \left(E_{\theta\varphi'_{mn}} + E_{\varphi\varphi'_{mn}}\right)\mathbf{u}_{\varphi'_{mn}}\\
&= \left(E_\theta \mathbf{u}_{\theta\theta'_{mn}} + E_\varphi \mathbf{u}_{\varphi\theta'_{mn}}\right)\mathbf{u}_{\theta'_{mn}}\\
&\quad + \left(E_\theta \mathbf{u}_{\theta\varphi'_{mn}} + E_\varphi \mathbf{u}_{\varphi\varphi'_{mn}}\right)\mathbf{u}_{\varphi'_{mn}}.
\end{aligned} \tag{11}$$

Two orthogonal electric field vectors $E_{\theta'_{mn}}$ and $E_{\varphi'_{mn}}$ in the local coordinate can be expressed as follows using E_θ and E_φ which are defined in the global coordinate:

$$\begin{aligned}
E_{\theta'_{mn}} &= E_\theta u_{\theta\theta'_{mn}} + E_\varphi u_{\varphi\theta'_{mn}},\\
E_{\varphi'_{mn}} &= E_\theta u_{\theta\varphi'_{mn}} + E_\varphi u_{\varphi\varphi'_{mn}}.
\end{aligned} \tag{12}$$

Assume the direction pattern of the (m,n)th dipole is $\mathbf{g}_{mn}(\theta'_{mn},\varphi'_{mn})$. $\mathbf{g}_{\theta_{mn}}(\theta'_{mn},\varphi'_{mn})$ and $\mathbf{g}_{\varphi_{mn}}(\theta'_{mn},\varphi'_{mn})$ are two component vectors by which $\mathbf{g}_{mn}(\theta'_{mn},\varphi'_{mn})$ is projected onto the $\mathbf{u}_{\theta'_{mn}}$ and $\mathbf{u}_{\varphi'_{mn}}$, respectively. Then the dipole's response $\mathbf{p}_{mn}$ to unit electric vector is represented as

$$\begin{aligned}
\mathbf{p}_{mn} &= \mathbf{g}_{mn}\left(\theta'_{mn},\varphi'_{mn}\right)\mathbf{E}'_{mn}\\
&= \left[\mathbf{g}_{\theta_{mn}}\left(\theta'_{mn},\varphi'_{mn}\right)\mathbf{u}_{\theta'_{mn}}\right.\\
&\quad \left. + \mathbf{g}_{\varphi_{mn}}\left(\theta'_{mn},\varphi'_{mn}\right)\mathbf{u}_{\varphi'_{mn}}\right]\\
&\quad \times \left(E_{\theta'_{mn}}\mathbf{u}_{\theta'_{mn}} + E_{\varphi'_{mn}}\mathbf{u}_{\varphi'_{mn}}\right)\\
&= \mathbf{g}_{\theta_{mn}}\left(\theta'_{mn},\varphi'_{mn}\right)E_{\theta'_{mn}}\mathbf{u}_{\theta'_{mn}}\\
&\quad + \mathbf{g}_{\varphi_{mn}}\left(\theta'_{mn},\varphi'_{mn}\right)E_{\varphi'_{mn}}\mathbf{u}_{\varphi'_{mn}}.
\end{aligned} \tag{13}$$

The polarized vector received by the dipole is represented as

$$\begin{aligned}
\mathbf{r}_{mn} &= \begin{bmatrix}\mathbf{u}_{x'_{mn}}\\ \mathbf{u}_{y'_{mn}}\end{bmatrix}\\
&= \begin{bmatrix}\mathbf{g}_{\theta_{mn}}\left(\theta'_{mn},\varphi'_{mn}\right)\cos\left(\theta'_{mn}\right)\cos\left(\varphi'_{mn}\right)E_{\theta'_{mn}} - \mathbf{g}_{\varphi_{mn}}\left(\theta'_{mn},\varphi'_{mn}\right)\sin\left(\varphi'_{mn}\right)E_{\varphi'_{mn}}\\ \mathbf{g}_{\theta_{mn}}\left(\theta'_{mn},\varphi'_{mn}\right)\cos\left(\theta'_{mn}\right)\sin\left(\varphi'_{mn}\right)E_{\theta'_{mn}} + \mathbf{g}_{\varphi_{mn}}\left(\theta'_{mn},\varphi'_{mn}\right)\cos\left(\varphi'_{mn}\right)E_{\varphi'_{mn}}\end{bmatrix}\\
&= \begin{bmatrix}\cos\left(\theta'_{mn}\right)\cos\left(\varphi'_{mn}\right) & -\sin\left(\varphi'_{mn}\right)\\ \cos\left(\theta'_{mn}\right)\sin\left(\varphi'_{mn}\right) & \cos\left(\varphi'_{mn}\right)\end{bmatrix} \times \left(\begin{bmatrix}\mathbf{g}_{\theta_{mn}}\left(\theta'_{mn},\varphi'_{mn}\right)\\ \mathbf{g}_{\theta_{mn}}\left(\theta'_{mn},\varphi'_{mn}\right)\end{bmatrix} \otimes \begin{bmatrix}E_{\theta'_{mn}}\\ E_{\varphi'_{mn}}\end{bmatrix}\right),
\end{aligned} \tag{14}$$

$$\mathbf{E}'_{mn} = \begin{bmatrix}E_{\theta'_{mn}}\\ E_{\varphi'_{mn}}\end{bmatrix} = \begin{bmatrix}u_{\theta\theta'_{mn}} & u_{\varphi\theta'_{mn}}\\ u_{\theta\varphi'_{mn}} & u_{\varphi\varphi'_{mn}}\end{bmatrix}\begin{bmatrix}E_\theta\\ E_\varphi\end{bmatrix} = \mathbf{U}_{mn}\begin{bmatrix}\sin\gamma\exp\left(j\eta\right)\\ \cos\gamma\end{bmatrix}, \tag{15}$$

where $\otimes$ represents the Kronecker product.

Definition 1. For two matrices $\mathbf{B} \in \mathscr{C}^{B\times L}$ and $\mathbf{D} \in \mathscr{C}^{D\times L}$, which possess the identical number of columns, the KR product of the two matrices is given by

$$\mathbf{B}\odot\mathbf{D} = \left[b_1\otimes d_1, b_2\otimes d_2, \ldots, b_L\otimes d_L\right] \in \mathscr{C}^{BD\times L}, \tag{16}$$

where $\otimes$ stands for the Kronecker product.

Definition 2. For two vectors $\mathbf{b} \in \mathscr{C}^B$ and $\mathbf{d} \in \mathscr{C}^D$, the Kronecker product of these two matrices is given by

$$\mathbf{b}\otimes\mathbf{d} = \begin{bmatrix}b_1\mathbf{d}\\ b_2\mathbf{d}\\ \vdots\\ b_B\mathbf{d}\end{bmatrix} = \mathrm{vec}\left(\mathbf{d}\mathbf{b}^T\right). \tag{17}$$

Under the condition of Q signals impinging on the array, the manifold matrix is expressed as

$$\mathbf{A} = \left[\mathbf{a}_1\left(\theta_1,\varphi_1,\gamma_1,\eta_1\right)\ \ \mathbf{a}_2\left(\theta_2,\varphi_2,\gamma_2,\eta_2\right)\ \ \cdots\ \ \mathbf{a}_Q\left(\theta_Q,\varphi_Q,\gamma_Q,\eta_Q\right)\right]. \tag{18}$$

The data model is established as

$$\mathbf{X}(t) = \mathbf{A}\mathbf{S}(t) + \mathbf{N}(t), \tag{19}$$

$$\mathbf{S}(t) = \left[s_1(t) \;\; s_2(t) \;\; \cdots \;\; s_Q(t)\right]^T, \tag{20}$$

where $\mathbf{N}(t)$ is the noise matrix which is assumed to be a Gaussian process with zero mean and σ^2 variance and $\mathbf{S}(t)$ is the $Q \times 1$ signal matrix.

The covariance matrix of $\mathbf{X}(t)$ is represented as

$$\mathbf{R}_{\mathbf{X}} = E\left[\mathbf{X}(t)\,\mathbf{X}^H(t)\right] = \mathbf{A}E\left[\mathbf{S}(t)\,\mathbf{S}(t)^H\right]\mathbf{A}^H + \sigma_n^2\mathbf{I}, \tag{21}$$

where σ^2 denotes the noise power and $E[\mathbf{S}(t)\mathbf{S}(t)^H]$ is the signal power.

In order to illuminate the pattern transformation, the cylinder is adopted as an example. The corresponding Euler rotation angles are

$$D = -\Theta, \qquad E = -\frac{\pi}{2}, \qquad F = 0, \tag{22}$$

respectively. D represents the rotation angle based on the right-hand rule in the first rotation and the Z-axis is the rotation axis; E represents the rotation angle in the second rotation and the Y-axis is the rotation axis; F represents the rotation angle in the third rotation and the X-axis is the rotation axis. Θ represents the angle between the position vector of the element and the positive direction of X-axis.

The unit vector of the qth incident signal (θ_q, φ_q) in the global coordinate can be expressed as

$$x = \sin\left(\theta_q\right)\cos\left(\varphi_q\right), \qquad y = \sin\left(\theta_q\right)\sin\left(\varphi_q\right), \qquad z = \cos\left(\theta_q\right). \tag{23}$$

Based on the Euler rotation matrix, the unit vector in global coordinate can be transformed into the local coordinate of the mth element:

$$\left[x_m \;\; y_m \;\; z_m\right]^T = R\left(D_m, E_m, F_m\right)\left[x \;\; y \;\; z\right]^T, \tag{24}$$

where

$$R\left(D_m, E_m, F_m\right) = \begin{bmatrix} \cos F & \sin F & 0 \\ -\sin F & \cos F & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} \cos E & 0 & -\sin E \\ 0 & 1 & 0 \\ \sin E & 0 & \cos E \end{bmatrix} \times \begin{bmatrix} \cos D & \sin D & 0 \\ -\sin D & \cos D & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{25}$$

Then the elevation θ and azimuth φ of the kth incident signal in the local coordinate of the mth element could be represented as

$$\varphi_{mq} = \arctan\left(\frac{y_m}{x_m}\right), \qquad \theta_{mq} = \arccos\left(z_m\right). \tag{26}$$

The element pattern transformation from the global coordinate to the local coordinate is completed at this time.

3. The Principle of Joint Beamforming

The polarized and spaced information of the target is assumed to be known. In order to maximize the power of the received signal, the weight vector should match the received signal on both polarized domain and spaced domain. By adjusting the weight vector value, it could match the received signal completely on both polarized domain and spaced domain, and the power of the received signal is maximized, which is called polarization-space beamforming (PSB).

Substituting (14) into (4), the steering vector is represented as

$$\begin{aligned} \mathbf{w}_{\mathrm{PSB}} &= \mathbf{a}(\theta, \varphi, \gamma, \eta) = \mathbf{r}\exp\left(-j2\pi\mathbf{P}\cdot\frac{\mathbf{u}}{\lambda}\right) \\ &= \mathbf{a}_p(\theta, \varphi, \gamma, \eta) \otimes \mathbf{a}_s(\theta, \varphi) \\ &= \mathbf{a}_{ps}(\theta, \varphi)\,\mathbf{a}_{pp}(\gamma, \eta) \otimes \mathbf{a}_s(\theta, \varphi). \end{aligned} \tag{27}$$

It can be seen from (20) that the polarization weight vector includes the polarization vector and spaced vector. However, the ordinary weight vector merely contains the spaced vector. Consider the condition that the polarization of the received signal is orthogonal to the polarization of the antenna; the polarization array has much advantage over ordinary array, because the polarization statue of the ordinary array is stationary, but the polarization statue of the polarization array can be changed flexibly. When the polarization of the polarization array and the received signal are identical, the polarization array can still possess a good performance. Based on the analysis above, it can be known that the polarization array has more robustness than the ordinary array.

In order to simplify the analysis, (17) is written in another form which contains desired signal, interference, and noise components, respectively:

$$\begin{aligned} \mathbf{X}(t) &= \mathbf{a}_0 s_0(t) + \sum_{i=1}^{J}\mathbf{a}_i s_i(t) + \mathbf{N}(t) \\ &= \mathbf{a}_0 s_0(t) + \mathbf{a}_J\mathbf{S}_J(t) + \mathbf{N}(t), \end{aligned} \tag{28}$$

where $s_0(t)$ is the desired signal and $s_1(t), s_2(t) \ldots, s_J(t)$ are interferences which are independent of both desired signal and noise.

Taking (21) into (19), the covariance matrix $\mathbf{R}_{\mathbf{X}}$ can be written in another form:

$$\begin{aligned} \mathbf{R}_{\mathbf{X}} &= E\left[\mathbf{X}(t)\,\mathbf{X}^H(t)\right] \\ &= P_S\mathbf{a}_0\mathbf{a}_0^H + \sum_{i=1}^{J}P_i\mathbf{a}_i\mathbf{a}_i^H + \sigma_n^2\mathbf{I}, \end{aligned} \tag{29}$$

where P_S is the signal power and P_i is the ith interference noise. Let $\mathbf{P}_{in} = \sum_{i=1}^{J}P_i\mathbf{a}_i\mathbf{a}_i^H + \sigma_n^2\mathbf{I}$; (23) can be obtained as:

$$\begin{aligned} E\left[y(t)\,y^H(t)\right] &= \mathbf{w}_{\mathrm{PSB}}^H\mathbf{R}_{\mathbf{X}}\mathbf{w} \\ &= P_S\mathbf{w}_{\mathrm{PSB}}^H\mathbf{a}_0\mathbf{a}_0^H\mathbf{w} + \mathbf{w}_{\mathrm{PSB}}^H\mathbf{P}_{in}\mathbf{w}. \end{aligned} \tag{30}$$

The signal-to-interference-plus-noise ratio (SINR) can be expressed as (24) using (23):

$$\mathrm{SINR}_{\mathrm{PSB}} = \frac{P_S \mathbf{w}_{\mathrm{PSB}}^H \mathbf{a}_0 \mathbf{a}_0^H \mathbf{w}}{\mathbf{w}_{\mathrm{PSB}}^H \mathbf{P}_{in} \mathbf{w}}. \tag{31}$$

3.1. No Interferences. The output of the polarization-spaced matched filtering under no interferences is expressed as

$$y(t) = \mathbf{w}_{\mathrm{PSB}}^H \mathbf{X}(t) = \mathbf{a}_0^H \mathbf{a}_0 s_0(t) + \mathbf{a}_0^H \mathbf{N}(t). \tag{32}$$

The signal power that the antenna received is denoted by σ_s^2, so the desired signal power based on the array output is

$$\begin{aligned}\mathbf{P}_S &= E\left[\left|\mathbf{a}_0^H \mathbf{a}_0 s_0(t)\right|^2\right] \\ &= (DE)^2 \sigma_s^2 \mathbf{a}_p^H(\theta,\varphi,\gamma,\eta)\,\mathbf{a}_p(\theta,\varphi,\gamma,\eta) \\ &= (DE)^2 \sigma_s^2 \left(\mathbf{a}_{ps}(\theta,\varphi)\,\mathbf{a}_{pp}(\gamma,\eta)\right)^H \mathbf{a}_{ps}(\theta,\varphi)\,\mathbf{a}_{pp}(\gamma,\eta).\end{aligned} \tag{33}$$

The noise power based on the array output is

$$\mathbf{P}_n = \mathbf{a}_0^H \mathbf{N}(t)\,\mathbf{N}^H(t)\,\mathbf{a}_0 = DE\sigma_n^2. \tag{34}$$

The signal-to-noise ratio (SNR) after the joint matched filtering is

$$\begin{aligned}\mathrm{SNR}_{\mathrm{PSB}} &= DE\frac{\sigma_s^2}{\sigma_n^2}\left(\mathbf{a}_{ps}(\theta,\varphi)\,\mathbf{a}_{pp}(\gamma,\eta)\right)^H \mathbf{a}_{ps}(\theta,\varphi)\,\mathbf{a}_{pp}(\gamma,\eta) \\ &= DE \cdot \mathrm{SNR}_{\mathrm{diple}},\end{aligned} \tag{35}$$

where $\mathrm{SNR}_{\mathrm{diple}}$ is the single dipole's SNR, which is determined by two parts: one is its own SNR of the signal σ_s^2/σ_n^2 and the other is the polarization vector $\mathbf{a}_p(\theta,\varphi,\gamma,\eta)$.

3.2. One Interference. Under the condition that one interference exists, linear constraints have been widely used in the adaptive beamforming. The weight vector of the LCMV beamforming is solved by minimizing the output power under a series of linear constraints $\mathbf{C}^H\mathbf{w} = \mathbf{f}$, where $\mathbf{C}$ is a $N \times m$ constraint matrix, $\mathbf{f}$ is $m \times 1$ constraint value vector, and N is the number of elements. The optimization problem of LCMV beamforming is expressed as follows:

$$\begin{aligned}&\min_{\mathbf{w}} \quad \mathbf{w}^H \mathbf{R}_{\mathbf{X}} \mathbf{w} \\ &\text{s.t.} \quad \mathbf{C}^H \mathbf{w} = \mathbf{f}.\end{aligned} \tag{36}$$

This optimization problem can be solved by the Lagrange multiplier method. The optimal weight vector is represented as follows:

$$\mathbf{w}_c = \mathbf{R}_{\mathbf{X}}^{-1}\mathbf{C}\left(\mathbf{C}^H\mathbf{R}_{\mathbf{X}}^{-1}\mathbf{C}\right)^{-1}\mathbf{f}, \tag{37}$$

where $\mathbf{C}$ and $\mathbf{f}$ are the constraint matrix and response vector, respectively. In this paper, the unit gain constraint $\mathbf{a}_e^H\mathbf{w}_c = 1$ is adopted, and $\mathbf{a}_e$ is the explored signal vector. Under ideal condition, $\mathbf{a}_e = \mathbf{a}_0$. The polarization information and spaced information of the desired signal is already known, in which the unit gain constraint is to be formed. Then (29) can be transformed as

$$\mathbf{w}_c = \mathbf{R}_{\mathbf{X}}^{-1}\mathbf{a}_e\left(\mathbf{a}_e^H\mathbf{R}_{\mathbf{X}}^{-1}\mathbf{a}_e\right)^{-1}\mathbf{f}. \tag{38}$$

Let $\eta = \mathbf{a}_e(\mathbf{a}_e^H\mathbf{R}_X^{-1}\mathbf{a}_e)^{-1}$; (31) is written as

$$\mathbf{w}_c = \eta\mathbf{R}_{\mathbf{X}}^{-1}\mathbf{a}_e. \tag{39}$$

Based on (16), $\mathbf{R}_{\mathbf{X}}^{-1}$ can be obtained by using matrix inversion theorem:

$$\mathbf{R}_{\mathbf{X}}^{-1} = \left(\mathbf{P}_{in} + P_S\mathbf{a}_0\mathbf{a}_0^H\right)^{-1} = \mathbf{P}_{in}^{-1} - \frac{P_S\mathbf{P}_{in}^{-1}\mathbf{a}_0\mathbf{a}_0^H\mathbf{P}_{in}^{-1}}{1 + P_S\mathbf{a}_0^H\mathbf{P}_{in}^{-1}\mathbf{a}_0}. \tag{40}$$

Taking (33) into (32), then let $\mathbf{a}_e = \mathbf{a}_0$; $\mathbf{w}_c$ is represented in another simple form as

$$\mathbf{w}_c = \frac{\eta\mathbf{P}_{in}^{-1}\mathbf{a}_0}{1 + P_S\mathbf{a}_0^H\mathbf{P}_{in}^{-1}\mathbf{a}_0}. \tag{41}$$

The SINR of the polarization-spaced LCMV beamforming is expressed as follows using (34):

$$\mathrm{SINR} = \frac{P_S\mathbf{w}_c^H\mathbf{a}_0\mathbf{a}_0^H\mathbf{w}_c}{\mathbf{w}_c^H\mathbf{P}_{in}\mathbf{w}_c} = P_S\mathbf{a}_0^H\mathbf{P}_{in}^{-1}\mathbf{a}_0. \tag{42}$$

When one interference merely exists, $\mathbf{P}_{in} = P_1\mathbf{a}_1\mathbf{a}_1^H + \sigma_n^2\mathbf{I}$,

$$\mathbf{P}_{in}^{-1} = \frac{1}{\sigma_n^2}\left(\mathbf{I} - \frac{P_1\mathbf{a}_1\mathbf{a}_1^H}{\sigma_n^2 + P_1\left\|\mathbf{a}_1\right\|^2}\right). \tag{43}$$

Taking (36) into (35), SINR of one interference is acquired.

$$\mathrm{SINR}_{\mathrm{one}} = \frac{P_S\left\|\mathbf{a}_0\right\|^2}{\sigma_n^2 + P_1\left\|\mathbf{a}_1\right\|^2} + \frac{P_SP_1\left\|\mathbf{a}_0\right\|^2\left\|\mathbf{a}_1\right\|^2}{\sigma_n^2\left(\sigma_n^2 + P_1\left\|\mathbf{a}_1\right\|^2\right)}\left(1 - |\tau|^2\right), \tag{44}$$

$$|\tau|^2 = \frac{\mathbf{a}_0^H\mathbf{a}_1\mathbf{a}_1^H\mathbf{a}_0}{\left\|\mathbf{a}_0\right\|^2\left\|\mathbf{a}_1\right\|^2}. \tag{45}$$

The correlation between desired signal and interference is reflected by $|\tau|$:

$$\begin{aligned}\mathbf{a}_0^H\mathbf{a}_1 &= \left(\mathbf{a}_{p0}\otimes\mathbf{a}_{s0}\right)^H\left(\mathbf{a}_{p1}\otimes\mathbf{a}_{s1}\right) = \left(\mathbf{a}_{p0}^H\mathbf{a}_{p1}\right)\left(\mathbf{a}_{s0}^H\mathbf{a}_{s1}\right), \\ \mathbf{a}_1^H\mathbf{a}_0 &= \left(\mathbf{a}_{p1}\otimes\mathbf{a}_{s1}\right)^H\left(\mathbf{a}_{p0}\otimes\mathbf{a}_{s0}\right) = \left(\mathbf{a}_{p1}^H\mathbf{a}_{p0}\right)\left(\mathbf{a}_{s1}^H\mathbf{a}_{s0}\right), \\ \left\|\mathbf{a}_0\right\|^2 &= \left\|\mathbf{a}_{p0}\right\|^2\cdot\left\|\mathbf{a}_{s0}\right\|^2, \\ \left\|\mathbf{a}_1\right\|^2 &= \left\|\mathbf{a}_{p1}\right\|^2\cdot\left\|\mathbf{a}_{s1}\right\|^2.\end{aligned} \tag{46}$$

$|\tau|$ could be written in another form as

$$|\tau| = \frac{\left(\mathbf{a}_{p0}^H\mathbf{a}_{p1}\right)\left(\mathbf{a}_{p1}^H\mathbf{a}_{p0}\right)}{\left\|\mathbf{a}_{p0}\right\|^2\left\|\mathbf{a}_{p1}\right\|^2}\frac{\left(\mathbf{a}_{s0}^H\mathbf{a}_{s1}\right)\left(\mathbf{a}_{s1}^H\mathbf{a}_{s0}\right)}{\left\|\mathbf{a}_{s0}\right\|^2\left\|\mathbf{a}_{s1}\right\|^2} = M_pM_s, \tag{47}$$

where M_p is polarization match coefficient and M_s is spaced match coefficient. Obviously, $0 < M_p < 1$, $0 < M_s < 1$.

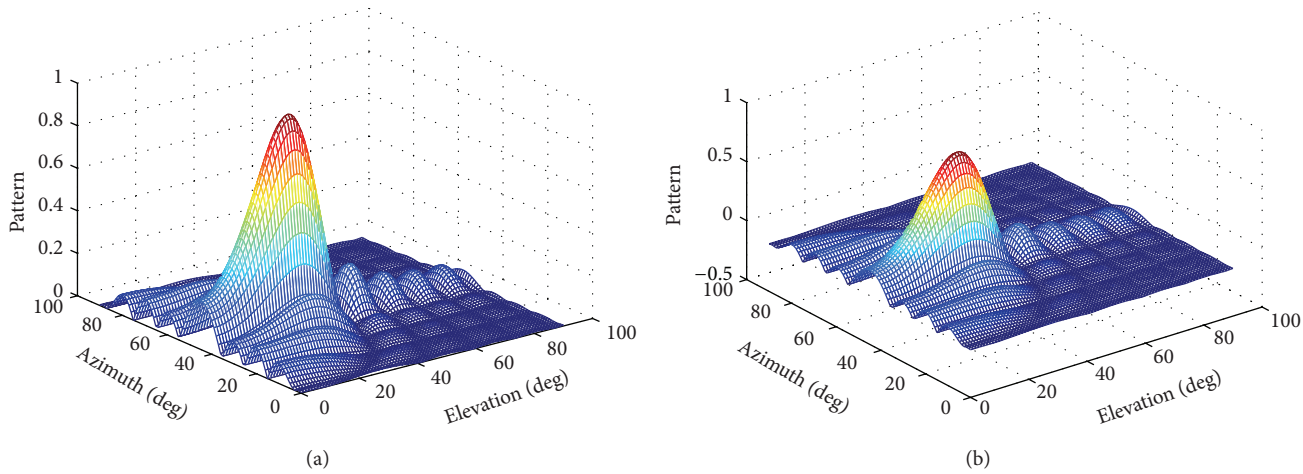


Figure 4: The array pattern. (a) The pattern of polarization array. (b) The pattern of ordinary array.

3.3. Multiinterferences. Consider multiinterferences impinging on the array; it is too complicated to calculate the inversion of the covariance matrix $\mathbf{P}_{in}$. Instead, an iterative method is used here:

$$\left(\mathbf{P}_{in}^{(k)}\right)^{-1} = \left(\mathbf{P}_{in}^{(k-1)} + \mathbf{P}_i^{(k)}\mathbf{a}_k\mathbf{a}_k^H\right)^{-1} = \left(\mathbf{P}_{in}^{(k-1)}\right)^{-1} - \frac{\mathbf{P}_i^{(k)}\left(\mathbf{P}_{in}^{(k-1)}\right)^{-1}\mathbf{a}_k\mathbf{a}_k^H\left(\mathbf{P}_{in}^{(k-1)}\right)^{-1}}{1 + \mathbf{P}_i^{(k)}\mathbf{a}_k^H\left(\mathbf{P}_{in}^{(k-1)}\right)^{-1}\mathbf{a}_k}. \tag{48}$$

According to the recursive relation of $\mathbf{P}_{in}$ as shown in (43), the recursive relation of SINR is represented as

$$\begin{aligned}\mathrm{SINR}^{(k)} &= P_S\mathbf{a}_0^H\mathbf{P}_{in}^{-1}\mathbf{a}_0 \\ &= P_S\mathbf{a}_0^H\left(\left(\mathbf{P}_{in}^{(k-1)}\right)^{-1} - \frac{\mathbf{P}_i^{(k)}\left(\mathbf{P}_{in}^{(k-1)}\right)^{-1}\mathbf{a}_k\mathbf{a}_k^H\left(\mathbf{P}_{in}^{(k-1)}\right)^{-1}}{1 + \mathbf{P}_i^{(k)}\mathbf{a}_k^H\left(\mathbf{P}_{in}^{(k-1)}\right)^{-1}\mathbf{a}_k}\right)\mathbf{a}_0 \\ &= \mathrm{SINR}^{(k-1)} - \frac{\mathbf{P}_S\mathbf{P}_i^{(k)}\left|\mathbf{a}_0^H\left(\mathbf{P}_{in}^{(k-1)}\right)^{-1}\mathbf{a}_k\right|^2}{1 + \mathbf{P}_i^{(k)}\mathbf{a}_k^H\left(\mathbf{P}_{in}^{(k-1)}\right)^{-1}\mathbf{a}_k}.\end{aligned} \tag{49}$$

It can be seen from (44) that SINR decreases as the number of interferences increases. When multiinterferences exist, SINR not only depends on the correlation between desired signal and interference, but also depends on the correlation among the interferences.

4. Simulation Results

Three simulation texts were performed corresponding to the conditions that no interferences exist, one interference exists, or multiinterferences exist, respectively. The element pattern used in simulation is the lowest order circular patch model:

$$g_\theta(\theta,\varphi) = \left\{J_2\left(\frac{\pi d}{\lambda}\sin\theta\right) - J_0\left(\frac{\pi d}{\lambda}\sin\theta\right)\right\} \times (\cos\varphi - j\sin\varphi), \quad 0 \le \theta \le \frac{\pi}{2}, \tag{50}$$

$$g_\varphi(\theta,\varphi) = \left\{J_2\left(\frac{\pi d}{\lambda}\sin\theta\right) + J_0\left(\frac{\pi d}{\lambda}\sin\theta\right)\right\} \times \cos\theta(\sin\varphi - j\cos\varphi), \quad 0 \le \theta \le \frac{\pi}{2}, \tag{51}$$

$$g_\theta(\theta,\varphi) = g_\varphi(\theta,\varphi) = 0, \quad \text{otherwise}, \tag{52}$$

where J_0 and J_2 are the zeroth- and second-order Bessel functions of the first kind. The element pattern transformation is completed with the method proposed above.

Simulation 1. To demonstrate the effectiveness of the proposed joint polarization-space matched filtering algorithm, a 5×5 cylindrical conformal array is used in the simulation. The normalized patterns of polarization array and ordinary array mounted on conformal carrier are depicted in Figures 4(a) and 4(b), respectively. The frequency of the incident signal is $f = 2$ GHz, $\lambda = c/f$, $c = 3 \times 10^8$ m/s is light velocity; $\alpha = 5^\circ$, and the distance between rings is $L = \lambda/2$. The radius of the cylinder is $R = L/\alpha$. The polarization parameter and DOA are $(\gamma_0, \eta_0) = (45^\circ, 270^\circ)$ and $(\theta_0, \varphi_0) = (30^\circ, 45^\circ)$, respectively. Figure 4(a) shows the pattern of the polarization array; the maximum gain happens at the direction of the incident signal. Figure 4(b) shows the pattern of ordinary array when the polarization parameter is $(\gamma_0, \eta_0) = (45^\circ, 315^\circ)$. The polarization weights could not be adjusted in ordinary array. The polarization mismatch exists between the element and incident signal, and the pattern gain is lower than that of the polarization array. However, the shapes of the polarization array and ordinary array are similar, which is consistent with the theoretical analysis.

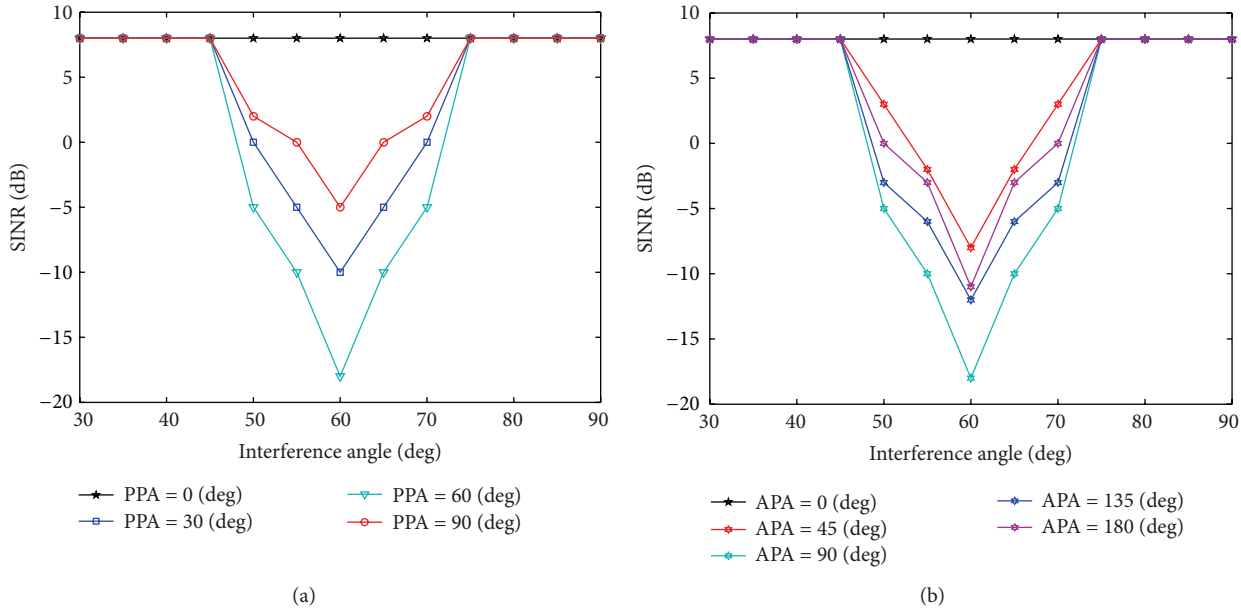


FIGURE 5: The SINR under different polarization parameters. (a) The SINR under different polarization phase angles. (b) The SINR under different auxiliary polarization angles.

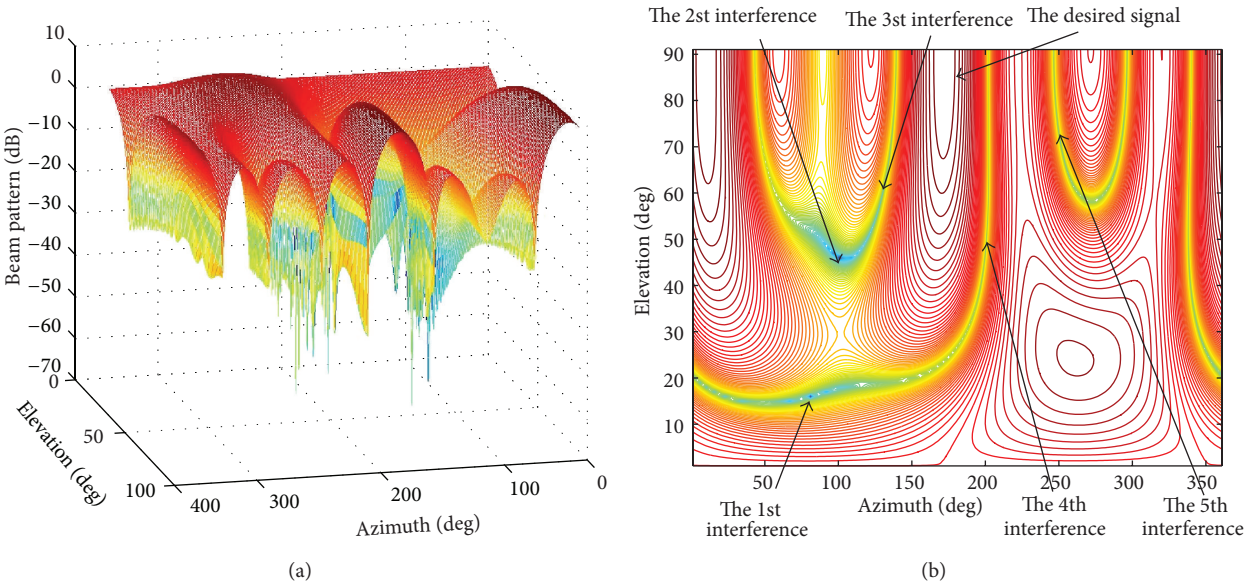


FIGURE 6: The LCMV beamforming. (a) Beamforming 3D pattern. (b) Contour plot.

Simulation 2. Only the uniform linear array (ULA) which is perpendicular to the "ring" is considered for array matched filtering. The elevation and azimuth of the desired signal is $(\theta_1, \varphi_1) = (60^\circ, 0^\circ)$, and the polarization parameter is $(\gamma_1, \eta_1) = (90^\circ, 60^\circ)$, SNR = 0 dB, and INR = 20 dB. The azimuth range of interference signal is zero. The elevation range of interference signal is 30°–90°. Other simulation conditions are the same as those in Simulation 1. As shown in Figure 5(a), the polarization phase angle (it is short for PPA in Figure 5(a)) of the interference signal is $\eta_2 = 60^\circ$, and the elevation of the interference signal is $\gamma_2 \in (0^\circ, 90^\circ)$; the step-length is 30°. As shown in Figure 5(b), the auxiliary polarization angle (it is short for APA in Figure 5(b)) of the interference signal is $\gamma_2 = 90^\circ$, and the elevation of the interference signal is $\eta_2 \in (0^\circ, 180^\circ)$; the step-length is 45°. In Figure 5, we can see that when the desired signal and interference signal are close ($|\theta_2 - \theta_1| < 10^\circ$), the polarization information plays an important role in improving the performance of filtering. The larger the difference between desired signal and interference signal, the better performance of filtering. When the desired signal and interference signal are relatively far apart ($|\theta_2 - \theta_1| > 15^\circ$), the space information

plays an important role in improving the performance of filtering; the function of polarization information is not obvious.

Simulation 3. The following LCMV beamforming experiment demonstrates that the joint polarization-space matched filtering technique can be applied to beamforming for conformal array effectively. The whole space is all included. The noise variance is assumed to be $\sigma^2 = 1$. The SNR = −10 dB, and INR = 20 dB. There are six incident signals impinging on the conformal array:

(1) the desired signal from $(\varphi_s, \theta_s) = (180^\circ, 85^\circ)$,

(2) the first interference signal from $(\varphi_{i1}, \theta_{i1}) = (80^\circ, 15^\circ)$,

(3) the second interference signal from $(\varphi_{i2}, \theta_{i2}) = (100^\circ, 45^\circ)$,

(4) the third interference signal from $(\varphi_{i3}, \theta_{i3}) = (130^\circ, 60^\circ)$,

(5) the fourth interference signal from $(\varphi_{i4}, \theta_{i4}) = (200^\circ, 50^\circ)$,

(6) the fifth interference signal from $(\varphi_{i5}, \theta_{i5}) = (250^\circ, 70^\circ)$.

Other simulation conditions are the same as the ones in Simulation 1. As shown in Figure 6, the LCMV beamforming approach sets five null points on the five interferences. The main beam is pointed at the desired signal. The proposed algorithm is very effective for beamforming of conformal array.

5. Conclusion

In this paper, a novel joint polarization-space matched filtering approach for conformal array is proposed. The analytical expression of polarization sensitive array beamforming is derived, which is equivalent to the matched coefficient multiplying the beamforming of the ordinary array. The polarization sensitive array has a higher gain by adjusting the polarization of the elements. Simulation results demonstrate that the joint polarization-space matched filtering can improve the filtering performance of the conformal array effectively.

Acknowledgments

This work was supported in part by the National Science Foundation of China under Grant 61201410 and in part by Fundamental Research Focused on Special Fund Project of the Central Universities (Program no. HEUCF130804).

References

[1] L. Josefsson and P . Persson, *Conformal Array Antenna Theory and Design*, IEEE Press, Piscataway, NJ, USA, 2006.

[2] L. Zou, J. Laseby, and Z. He, "Beamformer for cylindrical conformal array of non-isotropic antennas," *Advances in Electrical and Computer Engineering*, vol. 11, no. 1, pp. 39–42, 2011.

[3] W. T. Li, X. W. Shi, Y. Q. Hei, S. F. Liu, and J. Zhu, "A hybrid optimization algorithm and its application for conformal array pattern synthesis," *IEEE Transactions on Antennas and Propagation*, vol. 58, no. 10, pp. 3401–3406, 2010.

[4] H. A. Burger, "Use of Euler-rotation angles for generating antenna patterns," *IEEE Antennas Propagation Magazine*, vol. 37, no. 2, pp. 56–63, 1995.

[5] T. Milligan, "More applications of Euler rotation angles," *IEEE Antennas Propagation Magazine*, vol. 41, no. 4, pp. 78–83, 1999.

[6] B.-H. Wang, Y. Guo, Y.-L. Wang, and Y.-Z. Lin, "Frequency-invariant pattern synthesis of conformal array antenna with low cross-polarisation," *IET Microwaves, Antennas and Propagation*, vol. 2, no. 5, pp. 442–450, 2008.

[7] J. Li and R. T. Compton Jr., "Two-dimensional angle and polarization estimation using the ESPRIT algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 40, no. 5, pp. 550–555, 1992.

[8] X. Yuan, "Estimating the DOA and the polarization of a polynomial-phase signal using a single polarized vector-sensor," *IEEE Transactions on Signal Processing*, vol. 60, no. 3, pp. 1270–1282, 2012.

[9] M. Costa, A. Richter, and V. Koivunen, "DoA and polarization estimation for arbitrary array configurations," *IEEE Transactions on Signal Processing*, vol. 60, no. 5, pp. 2330–2343, 2012.

[10] S. Nordebo, M. Gustafsson, and J. Lundbäck, "Fundamental limitations for DOA and polarization estimation with applications in array signal processing," *IEEE Transactions on Signal Processing*, vol. 54, no. 10, pp. 4055–4061, 2006.

[11] D. Rahamim, J. Tabrikian, and R. Shavit, "Source localization using vector sensor array in a multipath environment," *IEEE Transactions on Signal Processing*, vol. 52, no. 11, pp. 3096–3103, 2004.

[12] P. Yang, F. Yang, and Z. Nie, "DOA estimation using MUSIC algorithm on a cylinderical conformal array," in *Proceedings of the IEEE Antennas and Propagation Society International Symposium*, pp. 5299–5302, June 2007.

[13] P. Yang, F. Yang, and Z. P. Nie, "DOA estimation with sub-array divided technique and interporlated esprit algorithm on a cylindrical conformal array antenna," *Progress in Electromagnetics Research*, vol. 103, pp. 201–216, 2010.

[14] Z.-S. Qi, Y. Guo, and B.-H. Wang, "Blind direction-of-arrival estimation algorithm for conformal array antenna with respect to polarisation diversity," *IET Microwaves, Antennas and Propagation*, vol. 5, no. 4, pp. 433–442, 2011.

[15] L. Zou, J. Lasenby, and Z. He, "Direction and polarization estimation using polarized cylindrical conformal arrays," *IET Signal Processing*, vol. 6, no. 5, pp. 395–403, 2012.

[16] L. Zou, J. Lasenby, and Z. He, "Pattern analysis of conformal array based on geometric algebra," *IET Microwaves, Antennas and Propagation*, vol. 5, no. 10, pp. 1210–1218, 2011.

[17] L. Zou, J. Lasenby, and Z. He, "Beamforming with distortionless co-polarisation for conformal arrays based on geometric algebra," *IET Radar, Sonar and Navigation*, vol. 5, no. 8, pp. 842–853, 2011.

[18] S. Yin, S. Ding, and H. Luo, "Real-time implementation of fault tolerant control system with performance optimization," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 5, pp. 2402–2411, 2014.

[19] S. Yin, S. Ding, A. Haghani, H. Hao, and P. Zhang, "A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process," *Journal of Process Control*, vol. 22, no. 9, pp. 1567–1581, 2012.

[20] S. Yin, S. Ding, A. Haghani, and H. Hao, "Data-driven monitoring for stochastic systems and its application on batch process," *International Journal of Systems Science*, vol. 44, no. 7, pp. 1366–1376, 2013.

[21] S. Yin, X. Yang, and H. Karimi, "Data-driven adaptive observer for fault diagnosis," *Mathematical Problems in Engineering*, vol. 2012, Article ID 832836, 21 pages, 2012.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 391782, 11 pages
http://dx.doi.org/10.1155/2013/391782

Research Article

Reconfiguration and Search of Social Networks

Lianming Zhang,[1] Aoyuan Peng,[1,2] and Jianping Yu[3]

[1] *College of Physics and Information Science, Hunan Normal University, Changsha 410081, China*
[2] *Department of Computer Science and Technology, Hunan University of Humanities, Science and Technology, Loudi 417000, China*
[3] *College of Mathematics and Computer Science, Hunan Normal University, Changsha 410081, China*

Correspondence should be addressed to Lianming Zhang; lianmingzhang@gmail.com

Received 6 October 2013; Accepted 17 November 2013

Academic Editors: H. R. Karimi and W. Zhang

Social networks tend to exhibit some topological characteristics different from regular networks and random networks, such as shorter average path length and higher clustering coefficient, and the node degree of the majority of social networks obeys exponential distribution. Based on the topological characteristics of the real social networks, a new network model which suits to portray the structure of social networks was proposed, and the characteristic parameters of the model were calculated. To find out the relationship between two people in the social network, and using the local information of the social network and the parallel mechanism, a hybrid search strategy based on k-walker random and a high degree was proposed. Simulation results show that the strategy can significantly reduce the average number of search steps, so as to effectively improve the search speed and efficiency.

1. Introduction

The social network is a system which consists of the interpersonal or intergroup relationships. In the system, an individual is abstracted as a node, interpersonal or intergroup social relationships act as the edge between nodes, and they get together to form a social network. Social relations can be many and varied, such as friend relationships between individuals, working relationships between colleagues, marriage relationships between families, and business relationships between companies. The sociologists wish the structural properties of the network can provide a systematic explanation for social phenomena; for example, the average distance of the social network can reflect the speed of information transmission in society. The clustering coefficient reflects the transitivity of social relations, that is, the occurrence possibility of social ties between an individual and his friends' friends. The node degree reflects the frequent interaction among social structures. The node degree distribution reflects the social stratification. Milgram arrived at some inferences as follows through the survey of social relations [1]: The average distance between any two people on the earth is 6, which to some extent reflects the small-world characteristic of interpersonal relationships, but the completion rate of Milgram's experiment is too low. Subsequently, Bacon game about the collaboration network of film actors and Erdös number about the collaboration network of mathematicians confirmed the small-world phenomenon, but they are still too small in size and the statistical properties of the relationship network have low credibility. However, for a fairly long time thereafter, random graph remains the basic theory and the analysis method for the complex structure of the network. But by the end of the 20th century, Watts and Strogatz revealed the small-world characteristic of the complex network [2], that is, large clustering coefficient and short average path length, and Barabási and Albert revealed the scale-free nature of the complex network [3] to establish appropriate models to illustrate the mechanism of these characteristics. Meanwhile, the research team of Watts has done some online experiments on the small-world property of social networks [4]. From then on, people began to consider the overall characteristics of real networks which have large number of nodes and complex structure, such as social networks, information networks, technological networks, and biological networks.

Clearly, to understand the correlation between network structure and network behavior, and to improve the behavior of the network, we need to have a good understanding of the structural features of the real network and on this basis to establish the appropriate network model. Milgram's small-world experiment not only reveals the small-world property of social networks, but also the searchability of social networks. Keinberg first had shown that in theory [5], a quick search can achieve in the complex networks with small-world characteristic. Since then, Watts et al. made a further study of this problem for social networks [6]. Adamic and Adar had verified some conclusions of the Watts model of social networks based on the small-world experiment of the e-mail network [7]. WS small-world network model explains the small-world property of social networks to a certain extent. However, the small-world property of the network does not necessarily mean that the network can quickly search. Whether a node in the network can find a shorter, or the shortest path between any of other nodes, depends on the network structure information of the node and the search strategy used by the node and the actual structure of the entire network. Therefore, the establishment of the network model and the implementation of efficient search strategy are in favor of finding the shortest chain between two people in the social network, and it has become an important topic in social network analysis.

In this paper, we formulate a deterministic small-world social network model and then put forward a hybrid search strategy based on k-walker random and a high degree (KRDS).

2. Related Works

2.1. Network Model. Nearest-neighbor coupled network (NNCN) model is a sparse regular network model in which each node only connects with its neighbor nodes. Suppose that a NNCN consists of N nodes and these N nodes are connected into a ring in turn, each node is connected with K neighbor nodes and these K neighbors are symmetrical to the node as the center. The average path length of the NNCN is $L \approx N/(2K)$, and the clustering coefficient is $C = 3(K-2)/(4(K-1)) \approx 3/4$. Obviously, the clustering property of the NNCN is too high, and the average path length is too long, so it does not have the small-world characteristic.

Contrary to the regular network, ER random graph (ERRG) model is a typical representative of random networks [8]. Assume that the total number of the nodes of the random network is N, any node connects to other nodes with the same probability p. The average path length of the ERRG is $L \propto \ln(N)/\ln(\langle k\rangle)$, where $\langle k\rangle = p(N-1) \approx pN$, and clustering coefficient of the ERRG is $C = p = \langle k\rangle/N \ll 1$. Obviously, the ERRG has a smaller average path length, and the clustering coefficient is very small, so it is not a small-world network.

The NNCN and the ERRG cannot reproduce some important features of real networks. After all, most of the real networks are neither completely regular nor completely random. As transition from a completely regular network to a completely random graph, Watts and Strogatz introduced a small-world network (WSSWN) model in 1998 [2]. The WSSWN sets the NNCN as the original model. To an identified edge in the original model, one of its endpoints is fixed, and another endpoint randomly selects a node to reconnect in addition to itself. When the reconnection probability is $p = 0$, the WSSWN is a NNCN. When $p = 1$, the WSSWN is a completely random network. Therefore, the transition from a completely regular network to a completely random network can be controlled by regulating the value of p. The average path length of the WSSWN is $L(p) = 2N/Kf(NKp/2)$ [9], where $f(u)$ is a universal scaling function, and if $u \ll 1$, $f(u)$ = constant and if $u \gg 1$, $f(u) = \ln(u)/u$. The clustering coefficient is $C(p) = 3(K-2)(1-p)^3/(4(K-1))$ [10]. When p is small (e.g., $0 < p \ll 1$), the local properties of the reconnected network and the original regular network are not very different; thus the clustering coefficient of the network will not change much ($C(p) \propto C(0)$), but its average path length has declined rapidly ($L(p) \ll L(0)$). The network that has a shorter average path length and a higher clustering coefficient is a small-world network.

A common feature of the ERRG and the WSSWN is able to represent the network's connectivity distribution using the Poisson distribution. The distribution has a peak at the average degree $\langle k\rangle$, and it exponentially decays. This means that when $k \gg \langle k\rangle$, the k-degree nodes are almost nonexistent. Therefore, these networks are also called uniform networks or exponential networks. Barabási and Albert proposed a scale-free network (BASFN) model in 1999 [3]. The node degree of the BASFN has no obvious characteristics, and it is used to analyze the real networks with power-law distribution. The BASFN starts from a network which has n_0 original nodes and then adds a new node each time, where the probability that the new node connects to an old node is $\prod_i$, and the degree of the node i is k_i and the degree of the node j is k_j. We have the following expression: $\prod_i = k_i/\sum_j k_j$. After t steps, the number of the nodes of the result graph is $N = t + n_0$. The average path length of the BASFN is $L \propto \log N/\log\log N$ [11], and the clustering coefficient is $C \propto (\ln(N))^2/N$ [12]. Clearly, the BASFN has small average path length, but when N is relatively large, the BASFN does not have the clustering characteristic.

In the past 10 years, many varieties of WSSWN and BASFN have been put forward. There are some common models, such as NW small-world model [9], deterministic small-world model [13], multidimensional growth of deterministic small-world network (DMG) model [14], local-world evolving network model [15], and hierarchical network model [16].

2.2. Search Strategies. When the source node s applies breadth first search (BFS) strategy to search for the target node t, s first judges whether t exists in the neighbor nodes; if any, it is returned to the source node, otherwise, the neighbor nodes pass information to their respective neighbor nodes. Repeat this process until it finds any neighbor node connected with t.

When the source node s applies depth first search (DFS) strategy to search the target node t, s first inquires whether t

exists along a possible branch path, if any, the search is complete, otherwise, it explores as far as possible along each branch before backtracking, and each node can only visit once. Repeat this process until it finds the target node t.

When the source node s applies random walk search (RWS) strategy to search for the target node t, s first judges whether its own neighbor nodes have t. If any, it stops searching, otherwise, it inspects all the neighboring nodes. Repeat this process until the target node is found [17]. The RWS can be further divided into unrestricted random walk (URW), no-retracing random walk search (NRRW), and self-avoiding random walk search (SARW).

When the source node s applies k-walker random walk search (k-RWS) strategy to search for the target node t, s will pass k copies of queried information to k randomly selected neighbors; thus there are k visitors to find t forward. Each queried information takes its own random walk. Each walker periodically talks with the querying node to decide whether that walker should terminate [18].

When the source node s uses high degree search (DS) strategy to search for the target node t, s searches for the nodes with a maximum degree in its neighbors, if any, it stops searching, otherwise, it keeps searching for the maximum degree one in its all the neighboring nodes [19].

Overall, the BFS strategy can find the shortest path between any two nodes. The shortest path length of the small-world networks is $L \propto \log N$, so the BFS strategy in finding the path between the nodes is very effective, and the query is parallel, so the search range will increase exponentially; usually just the inquiries of a few steps will be throughout the network and the search speed is quite fast. However, with the expansion of network size, this approach will produce a large number of querying message flows in the network, resulting in a sharp increase in network traffic and network congestion. The DFS strategy may make the search process along the futile path, resulting in a too long path. It is resolved using the boundary of the depth, and then the path may be the shortest. The number of search steps of the RWS strategy is larger than that of the BFS strategy, and the traffic reduces, but the same time, search speed is greatly reduced. The k-RWS strategy makes the search time greatly reduced, and the message traffic generated is much lower than that of the BFS strategy and search performance is better than that of the DFS strategy. The average path length obtained from the DS strategy gets on the same order of magnitude with the average path length obtained from the BFS strategy, and the search speed is fast, but the success rate of search is less than 100%.

3. Reconfiguration Method

Real social networks have small-world characteristic, that is, large clustering coefficient and short average path length. Regular network models and random graph models cannot depict the small-world characteristic, and scale-free network models do not have obvious clustering effect. Therefore, the main use of small-world models is to study the structural properties of social networks and to illustrate the generation mechanism of these features. According to the construction methods of the models, small-world networks can be divided into two categories: statistical small-world network models and the deterministic small-world network models. Connection between two nodes in the statistical models is random, and they are not suitable to describe the deterministic relations between the two nodes in social networks. The typical clustering coefficient of the existing deterministic small-world models is about 0.693 [20], and it is less than that of the real social networks, such as the clustering coefficient of the network of film actors which is about 0.78 and that of the networks of company directors which is about 0.88. In this paper, we propose a new social network reconfiguration method and the specific process is as follows.

(1) Construct a sphere.

(2) For $t = 0$, construct a triangular pyramid, denoted by CUB(0), whose four vertexes are placed in the spherical surface, and each vertex is connected with one of the other three vertexes.

(3) For $t = 1$, add a new node on the corresponding sphere of each outside of the triangular pyramid, and add three new edges between the new node and each of three vertexes constructing a corresponding side of the triangular pyramid. It is denoted by CUB(1).

(4) After t times, create a new graph, denoted by CUB(t) in turn.

Through analysis and calculation, we find that, as $t \to \infty$, the node degree distribution function of the model is an exponential distribution, that is, $p \propto 3^{-\log_2^{k-1}} - 3^{-\log_2^{k}}$, and the clustering coefficient of the model is $C \approx 0.828$ and the average path length of the model is $L \approx \ln(n)$. This shows that the model has small-world property.

When $t = 8$, the network size N has reached 13124. Obviously, the construction algorithm of the model has fast convergence and easily constructs large-scale social networks.

4. KRDS

In this section, we propose a hybrid search strategy based on k-walker random and a high degree (KRDS). The basic ideas of the strategy are as follows: assume that each node in the network knows the information of its neighbor nodes, the source node s randomly selects k neighbor nodes to transmit information. If there is the target node t in these k nodes, it will return information. Otherwise, the k nodes will, respectively, choose their neighbors with the high degree to pass the information, if the next step still does not find the target node, then use the k-walker random to inquire again. Repeat this process until it finds the target node. The pseudocode of the KRDS algorithm as in Algorithm 1.

In Algorithm 1, k has a constraint that it will less than or equal to the degree of the current node.

5. Simulation and Analysis Results

5.1. Search Efficiency. The efficiency of search strategies can be described by the query message traffic in the network and

```
KRDS(i, j, k)
if (i ≠ j)
        //Random walk search
        then go random(i, j, k)
else return routes
    if (i ≠ j)
          go max(i, j) //high degree search
until i = j
return routes
```

Algorithm 1

the average number of search steps. The latter reflects the search speed of the strategies, and the former can be measured by the number of traverse nodes from the search process, mainly affecting the network congestion. In this paper, we mainly use the latter to measure the efficiency of the search strategies; that is, the smaller the average number of search steps, the higher the search efficiency. When we put some kind of search strategies for a network with N nodes, repeat randomly to select n different source nodes. For each selected source node i, the sum of the number of the search steps from the source node to all other $N-1$ nodes based on this search strategy is written by $T_i = \sum_{j=1, j\neq i}^{N} t_{ij}$. Consequently, the average number of search steps between any two nodes is expressed as $T = \sum_i^N T_i/[N(N-1)]$.

5.2. Simulation Results. In this section, the simulation process is as follows. First, we generate the six network models with different types and sizes: NNCN, ERRG, WSSWN, BASFN, DMG and CUB. Then we assume that each node of the network models only knows its own neighbors and when the source node in the networks finds the target node, and for each model, we apply seven kinds of search strategies: BFS, DFS, DS, URW, NRRW, SARW, and KRDS. Finally, we calculate the average number of search steps of each strategy for each network model.

5.2.1. NNCN. In the simulation, we generate seven NNCN models with $K = 4$; that is, each node is connected with $K/2 = 2$ neighbors from right and left. The degree of each node is 4. The network size N is 20, 40, 50, 60, 80, 100, and 150, respectively. The BFS, the DFS, the DS, the URW, the NRRW, the SARW, and the KRDS ($k = 2$ and $k = 4$) were applied to each network. Figure 1 shows the impact of the network size N on the average number of search steps between any two nodes of the NNCN in the log-log scale.

The average number of search steps from the BFS, the DFS, and the DS strategies is less than that of the NRRW and the SARW strategies, that from the SARW is less than that of the above five strategies, and that from the KRDS strategy is relatively the smallest. For the KRDS strategy, when $k = 4$, the average number of search steps is less than that when $k = 2$. Thus, selecting an appropriate value of k, we can greatly reduce the average number of search steps. Pandit and Amritkar derived the theoretical value of the average number

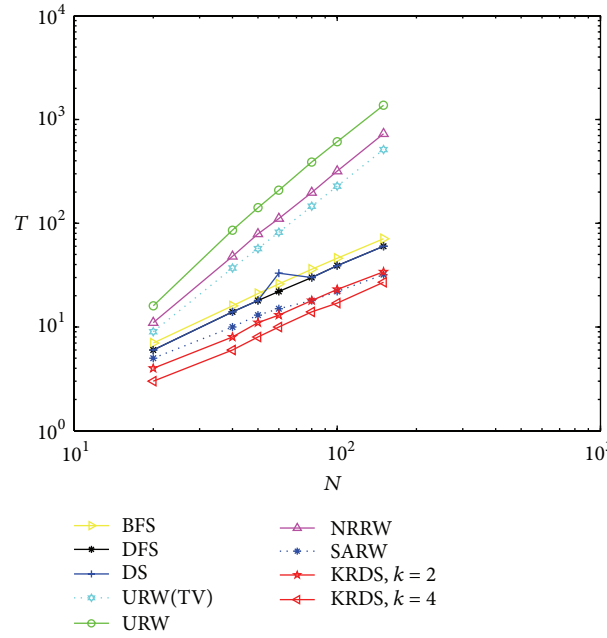


Figure 1: Plot of the average number T of the steps versus the network size N of the NNCN in the log-log scale.

of search steps in the application of the URW strategy for searching the NNCN model; that is, $T_{\mathrm{TV}} \approx (0.17 + 0.14\ln(K))(N/K)^2$ [17]. It can be seen that the simulation curves of the average number of search steps of the URW and the NRRW strategies is approximately paralleled with the theoretical curve of the average number of search steps of the URW strategy (see the URW (TV) curve in Figure 1); that is, they approximately obey a power-law relationship: $T \propto N^2$. The simulation curves of the average number of search steps from the BFS, the DFS, the DS, the SARW, and the KRDS strategies are almost parallel, but their slope is slightly less than the theoretical value of the URW strategy. Therefore, they approximately obey power-law relationship: $T \propto N^{\beta}$, where $1 < \beta < 2$.

5.2.2. ERRG. In the simulation, we generate seven ERRG models with node connection probability $p = 0.15$ and 0.03. The network size N of the ERRG is 20, 40, 50, 60, 80, 100, and 150, respectively. The simulation process is similar to the NNCN model. Figures 2 and 3 show the impact of the network size N on the average number T of search steps of the ERRG with connection probability $p = 0.15$ and $p = 0.03$ in the log-log scale.

Figure 2 shows that the average number of search steps from the URW and the NRRW strategies is the biggest, that of the BFS, the DFS, the DS, and the SARW is less than above, and that of the KRDS is relatively the smallest. The theoretical value of the average number of search steps from the ERRG model based on the NRRW strategy is $T_{\mathrm{TV}} \approx N/(p(N-1))$. It can be seen when the network size is big, the simulation curves of the average number of search steps obtained by the BFS, the DFS, the URW, the NRRW, and the SARW strategies are

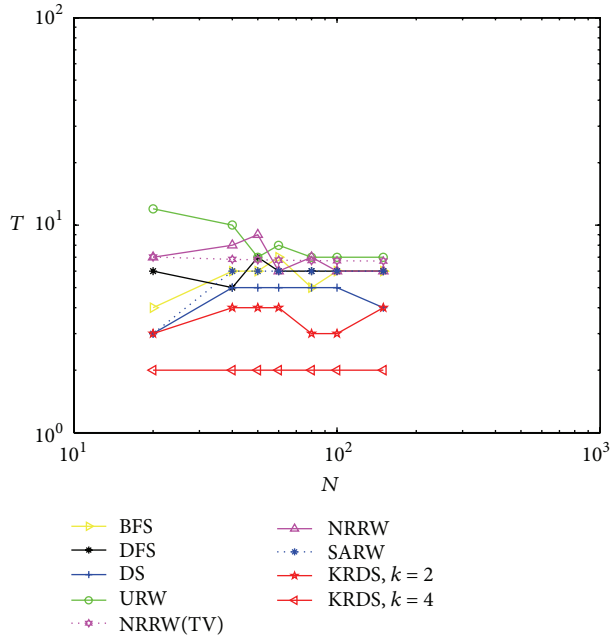


FIGURE 2: Plot of the average number T of the search steps versus the network size N of the ERRG with connection probability $p = 0.15$ in the log-log scale.

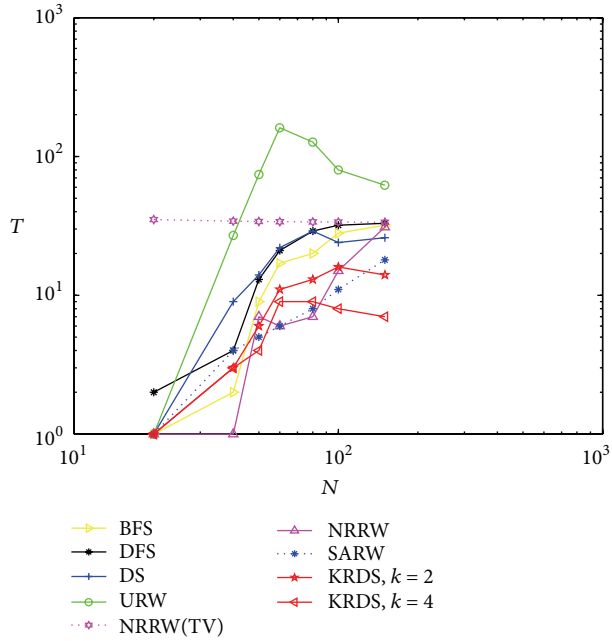


FIGURE 3: Plot of the average number T of the search steps versus the network size N of the ERRG with connection probability $p = 0.03$ in the log-log scale.

approximate equal to the theoretical curve of the average number of search steps from the NRRW strategy (see NRRW (TV) curve in Figure 2). When the network size is small, the average number of search steps obtained by the BFS, the DFS, the URW, the NRRW, and the SARW strategies will be so very different depending on the network size. When the network size reaches a certain value, the average number of search steps has nothing to do with the network size, and that from the DS strategy will appear with the process of the increase, the stability, and the decline with the increase of network size and that from the KRDS ($k = 2$) strategy will appear as fluctuated phenomenon with the increase in the network size. When we select the most appropriate value of k (e.g., $k = 4$), the average number of search steps from the KRDS strategy has nothing to do with the network size; it is in good agreement with the theoretical value of the NRRW strategy.

Figure 3 shows that the average number of search steps from the BFS, the DFS, the DS, the NRRW, and the SARW strategies increases with the increase of network size and that from the URW and the KRDS strategies increases first with the increase of network size and then decreases. When the network size reaches a certain value, the simulation value of the average number of search steps from the BFS, the DFS, and the NRRW strategies and the theoretical value of that from the NRRW strategy are in good agreement. When the network size is large, the average number of search steps from the URW strategy is the biggest and that from the KRDS strategy is relatively the smallest. Overall, for different ERRG models for the connection probability $p = 0.15$, the average number of search steps obtained from various strategies is about 5, far from the average number (e.g., 20) of search steps for the connection probability $p = 0.03$. We can know from the critical value of the ERRG model which is $p_c \propto \ln(N)/N$, and the seven critical values of different networks above is 0.1498, 0.0922, 0.0782, 0.0682, 0.0548, 0.0461 and 0.0034, respectively. Obviously, the connection probability $p = 0.15$ is greater than the maximum critical value, so the random graph is connected graph, the connection probability $p = 0.03$ is less than the minimum critical value, the random graph is nonconnected graph, and the simulation results are far away from the theoretical values.

In the following section, we consider the average number of search steps for a different node connection probability. In the simulation, we generate twelve ERRG models with the node connection probability $p = 0.001, 0.005, 0.01, 0.02, 0.04, 0.06, 0.1, 0.2, 0.4, 0.6, 0.8$, and 1, respectively, and the network size is all $N = 100$. For each connection probability, using a similar simulation process, the impact of the node connection probability p on the average number T of search steps of the ERRG models is shown in the log-log scale in Figure 4.

The curves of the average number of search steps from different search strategies have a peak value, and the average number of search steps from the BFS, the DFS, the DS, the URW, and the KRDS strategies to reach the peak corresponding to the connection probability $p = 0.04$ is in good agreement with the theoretical threshold of the ERRG connectivity $p_c \propto \ln(N)/N = 0.0461$, while the connection probability of the NRRW and SARW strategies reaching the peak is about 0.06 and there are some deviations with the theoretical value. Obviously, when the ERRG model is the nonconnected graph, that is, the connection probability is less than the critical value, the average number of search steps from each search strategy will increase with the increase of the connection probability. When the ERRG shows the connected graph, the connection probability is greater than the critical value,

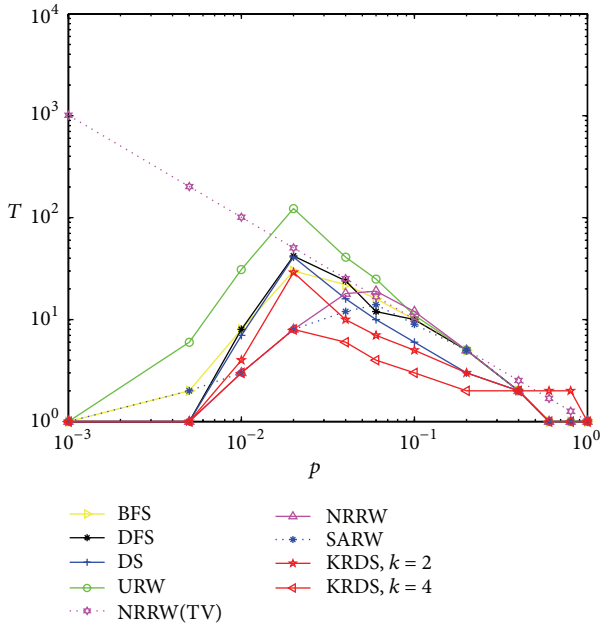


Figure 4: Plot of the average number T of search steps versus the connection probability p of ERRG in the log-log scale.

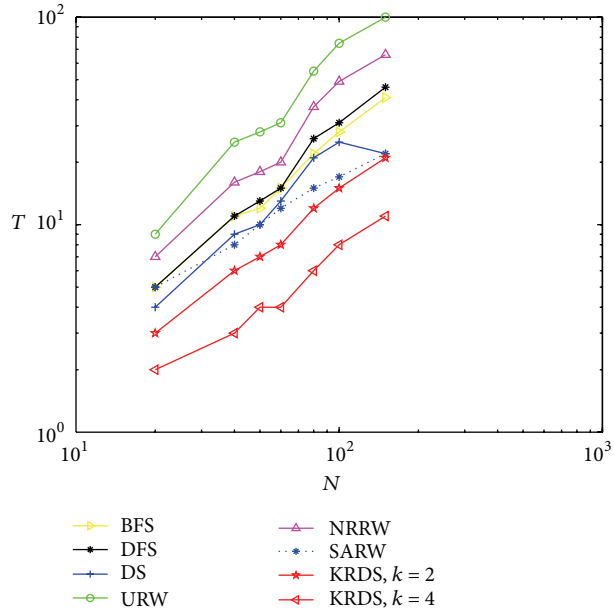


Figure 5: Plot of the average number T of search steps versus the network size N of the WSSWN with connection probability $p = 0.3$ in the log-log scale.

the average number of search steps from each search strategy will increase with the decrease of connection probability and obey approximate power-law relationship: $T \propto p^{-\alpha}$, where $\alpha > 0$. Simulation results of the average number of search steps from the BFS, the DFS, the NRRW, and the SARW strategies and the theoretical value of the NRRW strategy are in good agreement. Overall, the average number of search steps from the URW strategy is the biggest and that from the KRDS ($k = 4$) strategy is relatively the smallest.

5.2.3. WSSWN. We generate seven WSSWNs in the simulation, and the network size is $N = 20$, 40, 50, 60, 80, 100, and 150, respectively; the coefficients of all the NNCNs are $K = 4$ and the reconnection probability of the NNCNs is $p = 0.3$ and 1, respectively. The simulation process is similar to the NNCN and the ERRG. Figures 5 and 6 show the impact of the network size N on the average number T of search steps of the WSSWN in the log-log scale.

Overall, the relationship between the average number of search steps from a variety of search strategies and the network size of the WSSWN is similar to that between the average number of search steps and the network size of the NNCN. The average number of search steps obtained by each search strategy increases with the increase of network size, and it is similar to the NNCN and approximately obeys the power-law relationship: $T \propto N^{\beta}$. Those obtained by the URW and the NRRW strategies is the biggest and those obtained by the KRDS strategy is relatively the smallest. A WSSWN is the ERRG when the reconnection probability is $p = 1$, so we can apply the NRRW strategy to get the theoretical value of the average number of search steps. Figure 6 shows that the simulation results of the average number of search steps based on the BFS, the DFS, the URW, the NRRW,

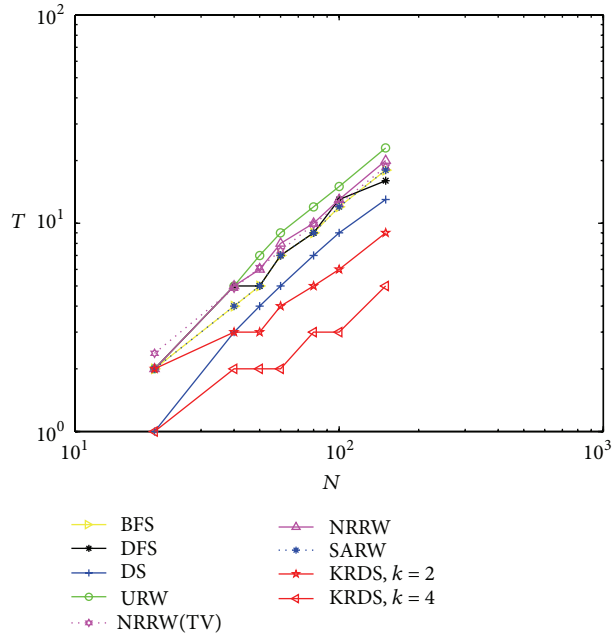


Figure 6: Plot of the average number T of search steps versus the network size N of the WSSWN with connection probability $p = 1$ in the log-log scale.

and the SARW strategies is closed with the theoretical value of the NRRW strategy.

In the following section, we would consider the average number of search steps obtained by a variety of search strategies from different connection probabilities.

We generate eight WSSWN models with the reconnection probability $p = 0.001$, 0.002, 0.005, 0.01, 0.06, 0.2, 0.5, and 1,

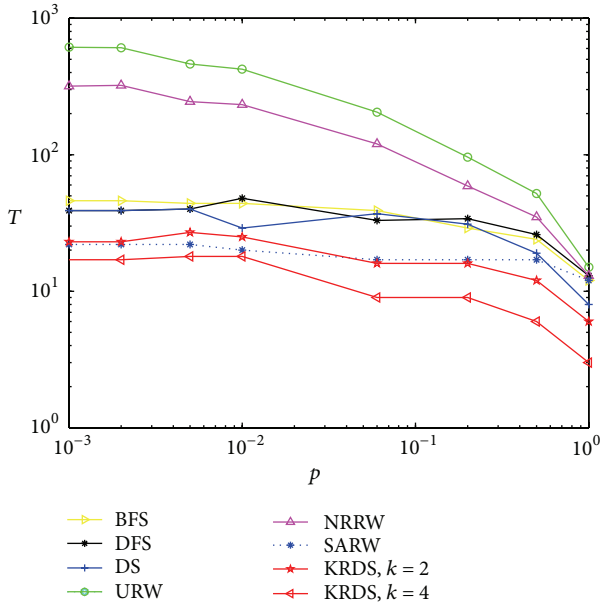


FIGURE 7: Plot of the average number T of search steps versus the connection probability p of the WSSWN in the log-log scale.

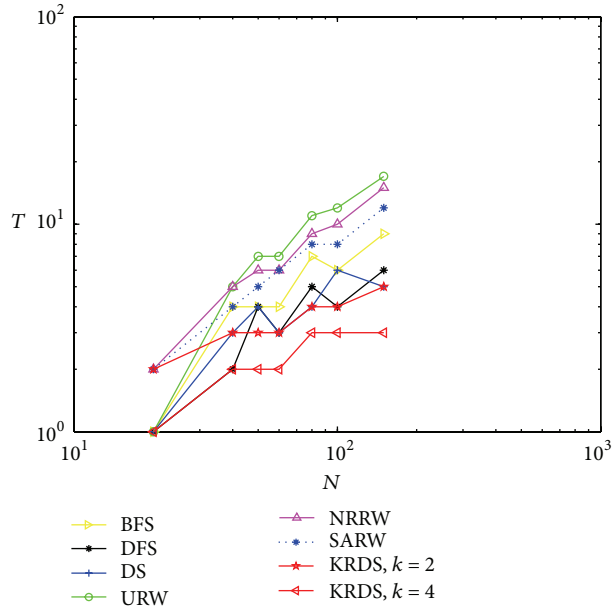


FIGURE 8: Plot of the average number T of search steps versus the network size N of the BASFN in the log-log scale.

respectively. The network size is $N = 100$, and the coefficients of the all NNCNs are $K = 4$. Figure 7 shows the impact of the reconnection probability p on the average number T of search steps of the WSSWN in the log-log scale.

The average number of search steps obtained from various strategies decreases with the increase of p, and those from the URW and the NRRW strategies decline the fastest, those from the DS and the KRDS strategies are behind the former, and those from the BFS, the DFS, and the SARW strategies are relatively the smallest. Those from the URW and the NRRW strategies are the biggest, but those from the KRDS strategy are the smallest. When the reconnection probability satisfies $0.01 < p < 0.5$, the average number T of search steps obtained from the URW, the NRRW, and the KRDS strategies and the reconnection probability p approximately obey the power-law relationship: $T \propto p^{-\alpha}$, where $a > 0$. For the WSSWN model, if $p = 0$, it is a NNCN and if $p = 1$, it is an ERRG. Therefore, if the reconnection probability p changes from 0 to 1, the WS small-world network model transfers from a NNCN to an ERRG. The simulation results also reflect such a process. The network is the NNCN with reconnection probability $p \to 0$, and the gap between the simulation results of the average number of search steps obtained from the URW, the NRRW, and the SARW strategies and the theoretical value (i.e., 228) of the average number of search steps obtained from the URW strategy is relative small; The network is an ERRG with reconnection probability $p = 1$, and the simulation results of the average number of search steps obtained from the BFS, the DFS, the NRRW and the SARW strategies are closed with the theoretical value (this is 12) of the average number of search steps obtained from the NRRW strategy.

5.2.4. BASFN. We generate seven BASFN models in which the number of the original nodes is $m_0 = 4$, and one new node is introduced and connected to the five nodes each time, and the network size is $N = 20$, 40, 50, 60, 80, 100, and 150, respectively. The simulation process is similar to the former network model. Figure 8 shows the impact of the network size N on the average number T of search steps of the BASFN in the log-log scale.

The average number of search steps from three random walk strategies and the KRDS strategy would increase with the increase of network size, but those from the BFS, the DFS, and the DS strategies would increase, reduce and then increase with the increase of network size in the BASFN. When network size is large, the curves obtained from three random walk strategies and the BFS strategy are more or less parallel. The curves from the KRDS strategy using different parameter k are roughly parallel for any network size. Thus, they approximately obey power-law relationship: $T \propto N^{\beta}$. Overall, the average number of search steps from three random walk strategies rises the fastest, and the increase rate of the KRDS strategy is relatively the slowest. The average number of search steps from the random walk strategy is the largest and that from the KRDS strategy is relatively the smallest.

5.2.5. DMG. We generate five DMG models, and the network size is $N = 10$, 22, 46, 94, and 190, respectively. Because the relationship of network size and the number of iterations satisfies $N = 6 \times 2^t - 2$, we have $t = 1$, 2, 3, 4, and 5, respectively. Simulation process is similar to the previous four network models. Figure 9 shows the impact of the network size N on the average number T of search steps in the DMG model, and Figure 10 shows the impact of the number t of iterations on the average number T of search steps in the DMG model in the semilogarithmic scale.

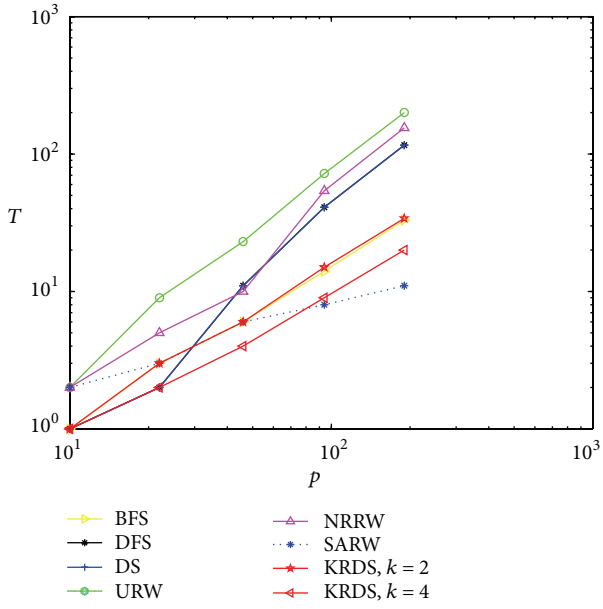


Figure 9: Plot of the average number T of search steps versus the network size N of the DMG in the log-log scale.

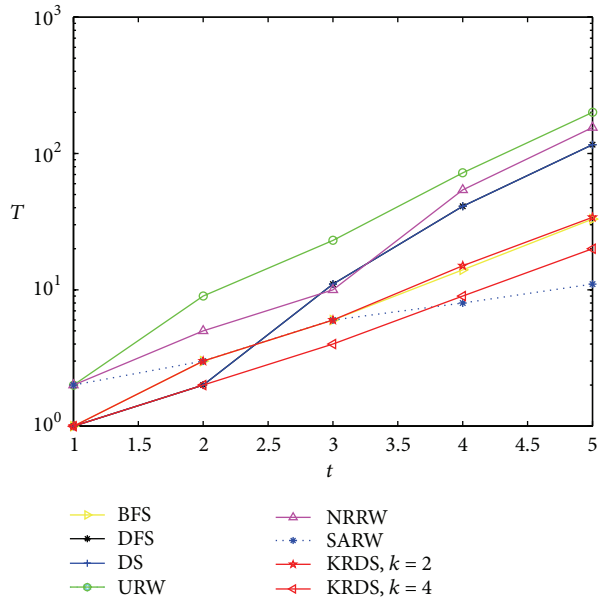


Figure 10: Plot of the average number T of search steps versus the number t of iterations of the DMG in the semilog scale.

The average number of search steps obtained from various strategies would increase with the increase of the number of iterations and the network size and that obtained from the URW strategy is the largest. When the network size is smaller, that obtained from the KRDS ($K = 4$) strategy is relatively the smallest. When the network size is larger, that obtained from the SARW strategy is relatively the smallest. Those obtained from the DFS strategy and the DS strategy are equal. When the network is large, the curves obtained from the DFS, URW, DS, and KRDS strategies are parallel, and they approximately

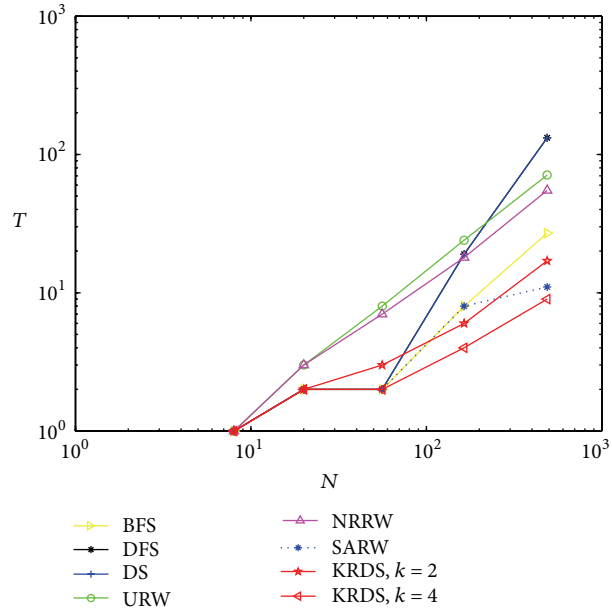


Figure 11: Plot of the average number T of search steps versus the network size N of the CUB in the log-log scale.

obey power-law relationship: $T \propto N^{\beta}$. In addition, that from the KRDS strategy would decrease with the increase of the parameter k, and we can adjust the parameter k values to get a smaller average search steps.

5.2.6. CUB. We generate five deterministic small-world network models with spatial structure (CUB), and the network size is $N = 8, 20, 56, 164$, and 488, respectively, and the corresponding number of iterations $t = 1, 2, 3, 4$, and 5. The network size N and the number t of iterations satisfy the following relationship: $N = 2 \times 3^t + 2$. Simulation process is similar to the previous five network models. Figure 11 shows the impact of the network size N on the average number T of search steps in the CUB in the log-log scale, and Figure 12 shows the impact of the number t of iterations on the average number T of search steps in the CUB in the semilogarithmic scale.

The curve of the average number of search steps obtained from the CUB model is similar to that obtained from the DMG model based on various strategies. However, the former is far less than the latter. When the network size is smaller, the average number of search steps from the URW and NRRW strategies is the largest, but when the network size is large, that from the DS strategy is the largest. That from the KRDS ($K = 4$) strategy is relatively the smallest. When the network size is large, the curves of those from the BFS, the URW, the NRRW, and the KRDS strategies are approximately parallel, and they approximately obey the power-law relationship: $T \propto N^{\beta}$.

5.3. Discussion. We have known that the values of the average number of search steps obtained from the same kind of search strategies to different network models are different. Those obtained from different search strategies to the network models with the same type and size are not the same too. In order

Table 1: The number of search steps.

	N	BFS	DFS	DS	URW	NRRW	SARW	KRDS ($k = 2$)	KRDS ($k = 4$)
NNCN ($K = 4$)	164	74	56	56	537	410	32	31	21
	488	236	177	177	5075	3838	86	62	64
ERRG ($p = 0.04$)	164	23	23	19	35	29	21	11	6
	488	24	25	17	27	25	25	12	6
WSSWN ($K = 4$, $p = 0.3$)	164	25	30	18	40	32	25	12	6
	488	78	100	54	121	100	71	35	18
BASFN	164	12	8	6	19	17	12	6	3
	488	32	20	16	56	50	35	14	7
DMG	164	21	69	69	119	93	9	23	13
	488	91	425	425	815	646	15	40	41
CUB	164	8	19	19	23	18	8	6	4
	488	27	132	132	71	55	11	17	9

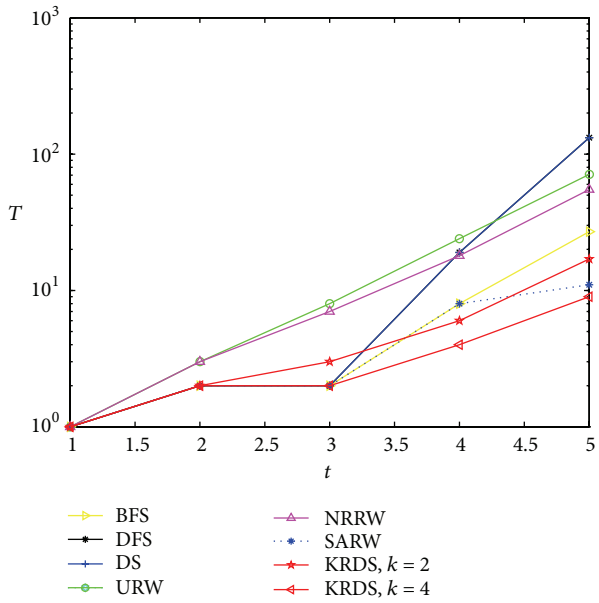


Figure 12: Plot of the average number T of search steps versus the number t of iterations of the CUB in the semilog scale.

to understand search capabilities of the different types of networks, such as the social network, and to find the best search strategy of various networks, we have adopted a similar simulation process.

First, we generate the NNCN models, the ERRG models, the WSSWN models, the BASFN models, the DMG models, and the CUB models in succession, and the network size of them is $N = 164$ and 488. The parameters in those models are set as follows: the parameter of the NNCN model is $K = 4$; the connection probability of the ERRG model is $p = 0.04$; the parameters of the WSSWN model are $K = 4$, and $p = 0.3$. The number of the original nodes in the BASFN model is $m_0 = 4$, and the degree of a new node is $m = 5$. When the network size is 164, the number of iteration steps of the DMG model is set to $t = 4$ and 5 and that of the CUB model is $t = 4$. When the network size is 488, the number of iteration steps of the DMG model is set to $t = 6$ and 7 and that of the CUB model is $t = 5$.

Then, we apply the BFS, DFS, DS, URW, NRRW, SARW, and KRDS ($k = 2$ and 4) strategies to six kinds of the network models to get the average number of search steps. Among them, for the DMG model, we first use iteration steps $t = 4$ and 5 to get the corresponding network size $N = 94$ and 190, respectively. And then, we apply a variety of search strategies to the two networks and can obtain two sets of the average number of search steps. Finally, we estimate the average number of search steps $T_{N=164}$. The estimation method is as follows: the logarithms of the average number T of search steps from various strategies and the network size N are approximately linear relationship; that is, $\log(T_{N=164}) \approx (\log(T_{N=190}) + \log(T_{N=94}))/2$, so the value of $T_{N=164}$ can be calculated. We can calculate the value of $T_{N=488}$ in the same way. Table 1 shows that the average number T of search steps obtained from a variety of search strategies used in different types of network in which the network size $N = 164$ and 488, respectively.

If the topology structure of the network is different, the average number of search steps is very different; namely, whether the network has the nature of quick search is relative with the network structure. In the above network, the search capability of the NNCN model is the worst, that of the BASFN model is the strongest, and that of the ERRG model is slightly weaker than that of the BASFN model, the search capabilities of small-world networks are stronger than that of the ERRG, but far less than that of the NNCN. In comparison, the BASFN has a bit of high degree nodes, and then there is a bit of long-range connections in the WSSWN, including the ERRG with reconnection probability of 1, but there is a bit of high degree nodes and long-range connections in the CUB, but high degree nodes and long-range connections lead directly to fast search capabilities of the networks. Of course, the proportion of high degree nodes and long-range connections in the network is also very critical, even all are small-world networks and search capabilities are very different. In three small-world networks, the search probability of the CUB is the strongest, that of the DMG is the worst, and that of the WSSWN is between these two. Thus, the small-world characteristic of the network not necessarily means that the network can quickly search.

In addition, for various search strategies, the values of the average number of search steps from the same network are different. That from the URW strategy is the largest, that from the NRRW strategy is a little smaller than that from the URW strategy, and that in the SARW strategy greatly reduces. The values of the average number of search steps from the BFS, DFS, and DS strategies are all a little larger than that of the SARW strategy but are far less than that of the NRRW strategy. In three strategies, that from the DFS strategy is the largest, that from the BFS strategy is the smallest, and that of the DS strategy is between these two. The average number of search steps from the KRDS strategy is the smallest in all strategies. Thus, the search strategy has a significant impact on search efficiency. In comparison, the NRRW strategy avoids the cycle query information between two neighbors in the URW strategy; therefore, the average number of search step has declined; the SARW strategy avoids repeatedly querying each node, and the average number of search steps is relatively small. The BFS strategy is parallel, and the average number of search steps is relatively small, the DFS and the DS strategies need to query individual nodes, so the average number of search steps is slightly larger than that of the BFS strategy. The KRDS strategy is more flexible, and it can get the minimum average number of search steps through adjusting parameter k according to the actual network. The KRDS strategy applies the method of k-degree parallel query, in every step of search process, and you can obtain k times of the node information. At the same time, it avoids querying each node repeatedly. In addition, the average number of search steps of the DMG model from the SARW strategy is the smallest, and that of other various networks from the KRDS strategy is relatively the smallest and the search efficient is the highest.

6. Conclusion

In this paper, we proposed a deterministic small-world network model of spatial structure (CUB). The node degree of the network has small-world properties and follows the power-law distribution. Its main characteristics are the ability to portray the social network with a strong clustering coefficient. We proposed the KRDS strategy to find the shortest chain between two people in the social network. The following discoveries were obtained from various search strategies in various network models. (1) The simulation results of the average number of search steps based on the URW strategy of the NNCN model and that based on the NRRW strategy of the ERRG and the WSSWN models are in good agreement with existing theoretical values. (2) The average number T of search steps and the network size N obtained from various search strategies based on various models obey approximate power-law relationship: $T \propto N^{\beta}$. (3) The average number T of search steps and the connection probability p obtained from various search strategies based on the ERRG and the WSSWN models obey approximate power-law relationship: $T \propto p^{-\alpha}$. (4) The search capability of various networks is different and for small-word network models, the quick search capability of the CUB model proposed in this paper is relatively strongest. (5) The search efficiency of various search strategies is also different, the KRDS strategy proposed in this paper is suitable for most types of network models, and the average number of search steps is the smallest, and in addition, the KRDS strategy only needs to use the local information and would not produce a large amount of network traffic which can quickly get access to a remote connection and effectively improve the efficiency of network search.

Conflict of Interests

The authors declare that they have no conflict of interests.

Acknowledgments

This study was supported in part by the National Natural Science Foundation of China (Grant nos. 60973129, 60903168), the China Postdoctoral Science Foundation funded project (Grant no. 200902324), the Guangdong Provincial Natural Science Foundation of China (Grant no. S2011010000812), the Hunan Provincial Natural Science Foundation of China (Grant no. 12JJ6063), and the Program for Excellent Talents in Hunan Normal University (Grant no. ET51102).

References

[1] S. Milgram, "The small world problem," *Psychology Today*, vol. 1, pp. 60–67, 1967.

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

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

[4] P. S. Dodds, R. Muhamad, and D. J. Watts, "An experimental study of search in global social networks," *Science*, vol. 301, no. 5634, pp. 827–829, 2003.

[5] J. M. Kleinberg, "Navigation in a small world," *Nature*, vol. 406, no. 6798, p. 845, 2000.

[6] D. J. Watts, P. S. Dodds, and M. E. J. Newman, "Identity and search in social networks," *Science*, vol. 296, no. 5571, pp. 1302–1305, 2002.

[7] L. Adamic and E. Adar, "How to search a social network," *Social Networks*, vol. 27, no. 3, pp. 187–203, 2005.

[8] P. Fronczak, A. Fronczak, and M. Bujok, "Exponential random graph models for networks with community structure," *Physical Review E*, vol. 88, no. 3, Article ID 032810, 2013.

[9] M. E. J. Newman and D. J. Watts, "Renormalization group analysis of the small-world network model," *Physics Letters A*, vol. 263, no. 4–6, pp. 341–346, 1999.

[10] A. Barrat and M. Weigt, "On the properties of small-world network models," *European Physical Journal B*, vol. 13, no. 3, pp. 547–560, 2000.

[11] R. Cohen and S. Havlin, "Scale-free networks are ultrasmall," *Physical Review Letters*, vol. 90, no. 5, Article ID 058701, 2003.

[12] A. Fronczak, P. Fronczak, and J. A. Holyst, "Mean-field theory for clustering coefficients in Barabasi-Albert networks," *Physical Review E*, vol. 68, no. 4, Article ID 046126, 2003.

[13] F. Comellas, J. Ozón, and J. G. Peters, "Deterministic small-world communication networks," *Information Processing Letters*, vol. 76, no. 1-2, pp. 83–90, 2000.

[14] A. Peng and L. Zhang, "Deterministic multidimensional growth model for small-world networks," http://arxiv.org/abs/1108.5450.

[15] P. Li, Q. Zhao, and H. Wang, "A weighted local-world evolving network model based on the edge weights preferential selection," *International Journal of Modern Physics B*, vol. 27, no. 12, Article ID 1350039, 2013.

[16] T. M. Sweet, A. C. Thomas, and B. W. Junker, "Hierarchical network models for education research," *Journal of Educational and Behavioral Statistics*, vol. 38, no. 3, pp. 295–318, 2013.

[17] S. A. Pandit and R. E. Amritkar, "Random spread on the family of small-world networks," *Physical Review E*, vol. 63, no. 4, Article ID 041104, 2001.

[18] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker, "Search and replication in unstructured peer-to-peer networks," in *Proceedings of the 16th International Conference on Supercomputing*, pp. 84–95, June 2002.

[19] L. A. Adamic, R. M. Lukose, A. R. Puniyani, and B. A. Huberman, "Search in power-law networks," *Physical Review E*, vol. 64, no. 4, Article ID 046135, 2001.

[20] Z. Zhang, L. Rong, and C. Guo, "A deterministic small-world network created by edge iterations," *Physica A*, vol. 363, no. 2, pp. 567–572, 2006.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 463828, 10 pages
http://dx.doi.org/10.1155/2013/463828

Research Article

Joint Estimation of 2D-DOA and Frequency Based on Space-Time Matrix and Conformal Array

Liang-Tian Wan,[1] Lu-Tao Liu,[1] Wei-Jian Si,[1] and Zuo-Xi Tian[2]

[1] *Department of Information and Communication Engineering, University of Harbin Engineering, Harbin 150001, China*
[2] *Science and Technology on Underwater Test and Control Laboratory, Dalian 116013, China*

Correspondence should be addressed to Lu-Tao Liu; liulutao@msn.com

Received 16 September 2013; Accepted 18 November 2013

Academic Editors: H. R. Karimi, X. Yang, Z. Yu, and W. Zhang

Each element in the conformal array has a different pattern, which leads to the performance deterioration of the conventional high resolution direction-of-arrival (DOA) algorithms. In this paper, a joint frequency and two-dimension DOA (2D-DOA) estimation algorithm for conformal array are proposed. The delay correlation function is used to suppress noise. Both spatial and time sampling are utilized to construct the spatial-time matrix. The frequency and 2D-DOA estimation are accomplished based on parallel factor (PARAFAC) analysis without spectral peak searching and parameter pairing. The proposed algorithm needs only four guiding elements with precise positions to estimate frequency and 2D-DOA. Other instrumental elements can be arranged flexibly on the surface of the carrier. Simulation results demonstrate the effectiveness of the proposed algorithm.

1. Introduction

Conformal array is the array mounted on the surface of the conformal carrier [1]. The advantage of the conformal array includes the alleviation of array weight, reduction of aerodynamic drag, reduction of radar cross-section (RCS), and the cover of the whole range of azimuth [2]. Thus the conformal array is of great interest in a variety of applications such as radar, sonar, wireless communication, and seismology.

Conventional algorithms for direction-of-arrival (DOA) estimation, such as multiple signal classification (MUSIC) [3] and estimation of signal parameters via rotational invariance techniques (ESPRIT) [4], are not suitable for conformal array because of the varying curvature of the conformal carrier. Based on array interpolation [5, 6], the manifold separation technique was applied to arbitrary array structure [7]. The above algorithm possesses good performance when the pattern of element is isotropic directional. However, due to the effect of the curvature of conformal carrier, each element of the conformal array has different patterns [8]. Thus, the algorithm proposed in [7] could not be used for conformal array. Not all elements can receive signals because of the "shadow effect" caused by metallic carrier. The subarray divided technique and interpolation transformation were adopted for DOA estimation of conformal array. But the interpolation accuracy of virtual manifold transformation is not high [9, 10]. Recently, DOA estimation algorithms for conformal array have been proposed. Based on ESPRIT and specially designed array structure, a blind DOA estimation algorithm was presented [11]. The iterative ESPRIT algorithm was proposed for joint DOA and polarization parameter estimation [12]. Nevertheless, the two algorithms need parameter pairing and special array design.

By combining the techniques of temporal filtering and spatial beam forming, the joint DOA and delay were estimated [13]. Similarly, the proposed approach in [14] was a novel twist of parameter estimation and filtering processes, in which two 1D frequencies and one 1D spatial MUSIC were employed to estimate the DOAs and frequencies, respectively. Two approximate maximum likelihood (ML) algorithms were proposed in the spatially correlated noise scenario for joint frequency and DOA estimation. However, MUSIC and ML algorithms used in [13–15] suffer from tremendous computational complexity and are only suitable for linear array or planar array, which could not be used for conformal array. The research of joint frequency and DOA estimation is rare for the conformal array.

As a useful analysis tool of data arrays, parallel factor (PARAFAC) analysis model [16, 17] is a generalization of low-rank matrix decomposition to three-way arrays (TWAs) or multiway arrays (MWAs), which was introduced in array signal processing to estimate DOA [18]. PARAFAC has been first introduced as a data analysis tool in psychometrics, where it has been used in arithmetic complexity, exploratory data analysis, and other fields. The PARAFAC model was developed by Sidiropoulos et al. [19]. Most parameter estimation algorithms for conformal array focus on DOA estimation. The frequency estimation is not researched as far as we known. For 2D-DOA estimation of conformal array, the algorithms based on ESPRIT suffer from parameter pairing problem. The algorithms based on MUSIC suffer from tremendous computational complexity. In this paper, a novel joint frequency and 2D-DOA estimation algorithm for conformal array is proposed. The spatial domain sampling and time domain sampling are utilized to construct the spatial-time matrix. The delay correlation function is used to suppress noise. Based on PARAFAC, the joint frequency and 2D-DOA estimation are accomplished without parameter pairing. Unlike the special array design in [9–12], the proposed algorithm needs only four guiding elements, other instrumental elements of the array can be arranged flexibly.

The organization of this paper is structured as follows. Section 2 introduces the structure of the conformal array and describes the snapshot data model. Section 3 contains the core contributions of this paper, including the construction of spatial-time matrix and the PARAFAC model used for high accuracy frequency and 2D-DOA estimation. Section 4 presents the simulation result. Section 5 summarizes our conclusions.

2. The Structure of the Conformal Array and the Snapshot Data Model

A curve array antenna mounted on an arbitrary shape carrier is shown in Figure 1. Assuming that there are M elements on the surface of the carrier, the coordinate of each element can be represented as (x_m, y_m, z_m), $m = 1, 2, \ldots, M$. Four guiding elements $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$, and $\mathbf{p}_4$ with precise positions are chosen, and $\mathbf{p}_1$ is assumed to be the reference element. $\mathbf{p}_m$ represents the position vector of the mth element on the surface of the carrier. $\vec{\mathbf{e}}_x$, $\vec{\mathbf{e}}_y$, and $\vec{\mathbf{e}}_z$ represent the unit vector of X-axis, Y-axis, and Z-axis, respectively. $\mathbf{p}_m$ can be expressed as

$$\mathbf{p}_m = x_m \vec{\mathbf{e}}_x + y_m \vec{\mathbf{e}}_y + z_m \vec{\mathbf{e}}_z. \tag{1}$$

Consider K narrow far field incident signals impinging on the conformal array, which is shown in Figure 1. The elevation and azimuth of kth incident signal are denoted as (θ_k, φ_k), $k = 1, 2, \ldots, K$. Thus, the general form of the array output is represented as

$$x_m(t) = \sum_{k=1}^{K} s_k(t)\, g_m \exp\left(-j\frac{2\pi f_k}{c}\mathbf{p}_m \bullet \mathbf{u}_k\right), \tag{2}$$

where $s_k(t)$ stands for the kth incident signal and g_m is the element pattern which is defined in the mth local coordinate.

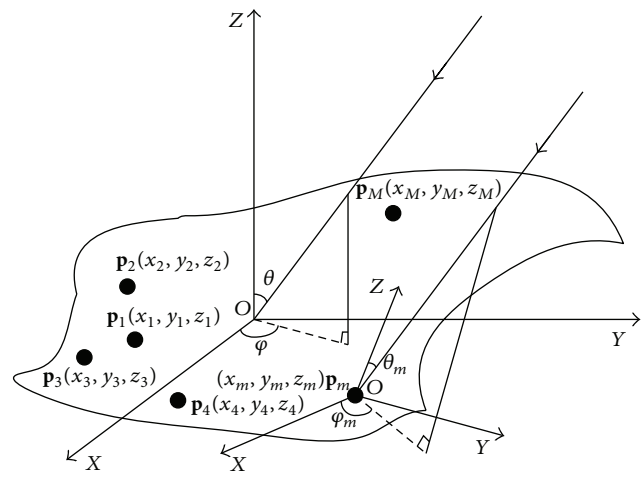


Figure 1: The structure of the conformal array.

f_k is the frequency of the kth incident signal, u_k is the direction vector of the incident signal $s_k(t)$.

Consider

$$\mathbf{u}_k = \sin(\theta_k)\cos(\varphi_k)\,\vec{\mathbf{e}}_x + \sin(\theta_k)\sin(\varphi_k)\,\vec{\mathbf{e}}_y + \cos(\theta_k)\,\vec{\mathbf{e}}_z. \tag{3}$$

The time domain sampling is done for the signal of the array output. The number of the tapped delay line is L. Then the lth tapped delay of the signal output of the mth element can be represented as

$$x_m(t - l\tau) = \sum_{k=1}^{K} s_k(t)\, g_m \exp\left[-j\omega_k(\tau_{mk} + l\tau)\right] + n_m(t - l\tau), \quad 0 \le l \le L - 1, \tag{4}$$

where τ_{mk} represents the time delay between the mth element $\mathbf{p}_m$ and reference element $\mathbf{p}_1$ when the mth element receives the kth incident signal, τ is the time interval of sampling, n_m is the additional white Gaussian noise (AWGN) with zero mean and variance σ^2, $\omega_k = 2\pi f_k$, and $\tau_{mk} = (\mathbf{p}_m \bullet \mathbf{u}_k)/c$.

The condition that the same incident signal impinges on the global coordinate and local coordinate is also shown in Figure 1. The coordinate which is constructed by the original point $\mathbf{p}_1$ is the global coordinate. The coordinate which is constructed by the original point $\mathbf{p}_m$ is the local coordinate. As shown in Figure 1, the elevation and azimuth of the incident signal in global coordinate and local coordinate are distinct. The pattern function $g_m(\theta_m, \varphi_m)$ is defined in the local coordinate of each element. Thus, the pattern in the local coordinate of each element is different. Here the parameter component transformation from the global coordinate to the mth local coordinate can be accomplished by using the general Euler rotate method [8]. More details can be found in [20].

Due to the "shadow effect" caused by the metal, not all elements in the conformal array can receive the signal. Thus, the subarray divided technique [9, 10] is adopted and the whole array can be divided into several parts. The same

parameter estimation mechanism is used for each part. For the sake of simplicity, it is assumed that all the elements can receive the signal.

3. Joint Frequency and 2D-DOA Estimation

3.1. The Construction of Spatial-Time Matrix. The delay correlation function between the mth element and nth element of the conformal array is constructed as follows:

$$\begin{aligned} R_{x_m x_n} &= E\left[x_m(t)\, x_n^*(t-\tau)\right] \\ &= \sum_{k=1}^{K} E\left[s_k(t)\, s_k^*(t-\tau)\right] \\ &\quad \times \exp\left[-j\omega_k\left(\tau_{mk}-\tau_{nk}-\tau\right)\right] \\ &\quad + E\left[n_m(t)\, n_n^*(t-\tau)\right] \\ &= \sum_{k=1}^{K} R_{s_k s_k}(\tau)\, g_m g_n \exp\left[-j\omega_k\left(\tau_{mk}-\tau_{nk}\right)\right] \\ &\quad \times \exp\left(j\omega_k\tau\right) + R_{n_m n_n}(\tau), \quad \tau > 0, \end{aligned} \tag{5}$$

where $\mathbf{R}_{s_k s_k}(\tau) = E[s_m(t)s_n^*(t-\tau)]$ represents the delay correlation function of $s_k(t)$, and $\mathbf{R}_{n_m n_n}(\tau) = E[n_m(t)n_n^*(t-\tau)]$ represents the noise cross-correlation function between the mth element and nth element.

On condition that the incident signal is coming from narrow far field and the bandwidth of the incident signal is $\mathbf{B}$, the delay correlation processing of the data that the elements receive takes a long period of time. Hence, $\mathbf{R}_{s_k s_k}(\tau) = E[s_m(t)s_n^*(t-\tau)] \neq 0$. In the more general case, when $\tau \ll 1/B$, the Gaussian white noise is not correlated during the time interval τ. At the same time, the variety of the signal envelope can be neglected. Then we can suppress the noise without destroying the signal.

Consider

$$\begin{aligned} \mathbf{R}_{n_m n_n}(\tau) &= E\left[n_m(t)\, n_n^*(t-\tau)\right] \\ &= \sigma^2 \delta(\tau)\, \delta(m-n) = 0, \end{aligned} \tag{6}$$

where the delay correlation function is constructed between $x_1(t-\tau)$ and the data that each element receives. Based on (5), we can obtain

$$\begin{aligned} \mathbf{R}_{x_m x_1}(\tau) &= E\left[x_m(t)\, x_1^*(t-\tau)\right] \\ &= \sum_{k=1}^{K} R_{s_k s_k}(\tau)\, g_m g_1 \exp\left[-j\omega_k\left(\tau_{mk}-\tau_{1k}\right)\right] \\ &\quad \times \exp\left(j\omega_k\tau\right), \quad 1 \le m \le M. \end{aligned} \tag{7}$$

Similarly, the delay correlation functions are constructed between $x_2(t-\tau)$, $x_3(t-\tau)$, $x_4(t-\tau)$, and $x_1(t)$ and the data that each element receives, respectively.

Consider

$$\begin{aligned} \mathbf{R}_{x_m x_2}(\tau) &= E\left[x_m(t)\, x_2^*(t-\tau)\right] \\ &= \sum_{k=1}^{K} \mathbf{R}_{s_k s_k}(\tau)\, g_m g_1 \exp\left[-j\omega_k\left(\tau_{mk}-\tau_{2k}\right)\right] \\ &\quad \times \frac{g_2}{g_1} \exp\left(j\omega_k\tau\right), \quad 1 \le m \le M, \\ \mathbf{R}_{x_m x_3}(\tau) &= E\left[x_m(t)\, x_3^*(t-\tau)\right] \\ &= \sum_{k=1}^{K} \mathbf{R}_{s_k s_k}(\tau)\, g_m g_1 \exp\left[-j\omega_k\left(\tau_{mk}-\tau_{3k}\right)\right] \\ &\quad \times \frac{g_3}{g_1} \exp\left(j\omega_k\tau\right), \quad 1 \le m \le M, \\ \mathbf{R}_{x_m x_4}(\tau) &= E\left[x_m(t)\, x_4^*(t-\tau)\right] \\ &= \sum_{k=1}^{K} \mathbf{R}_{s_k s_k}(\tau)\, g_m g_1 \exp\left[-j\omega_k\left(\tau_{mk}-\tau_{4k}\right)\right] \\ &\quad \times \frac{g_4}{g_1} \exp\left(j\omega_k\tau\right), \quad 1 \le m \le M, \\ \mathbf{R}_{x_m x_1}(0) &= E\left[x_m(t)\, x_1^*(t)\right] \\ &= \sum_{k=1}^{K} \mathbf{R}_{s_k s_k}(0)\, g_m g_1 \exp\left[-j\omega_k\left(\tau_{mk}-\tau_{1k}\right)\right] \\ &\quad + \sigma^2 \mathbf{I}, \quad 1 \le m \le M. \end{aligned} \tag{8}$$

The spatial-time matrix is constructed as follows:

$$\mathbf{R}_S(\tau) = \left[\mathbf{R}_{s_1 s_1}(\tau), \mathbf{R}_{s_2 s_2}(\tau), \ldots, \mathbf{R}_{s_K s_K}(\tau)\right]^T, \tag{9}$$

$$\mathbf{R}_f(\tau) = \left[\mathbf{R}_{x_1 x_1}(\tau), \mathbf{R}_{x_2 x_1}(\tau), \ldots, \mathbf{R}_{x_m x_1}(\tau)\right]^T, \tag{10}$$

$$\mathbf{R}_1(\tau) = \left[\mathbf{R}_{x_1 x_2}(\tau), \mathbf{R}_{x_2 x_2}(\tau), \ldots, \mathbf{R}_{x_m x_2}(\tau)\right]^T, \tag{11}$$

$$\mathbf{R}_2(\tau) = \left[\mathbf{R}_{x_1 x_3}(\tau), \mathbf{R}_{x_2 x_3}(\tau), \ldots, \mathbf{R}_{x_m x_3}(\tau)\right]^T, \tag{12}$$

$$\mathbf{R}_3(\tau) = \left[\mathbf{R}_{x_1 x_4}(\tau), \mathbf{R}_{x_2 x_4}(\tau), \ldots, \mathbf{R}_{x_m x_4}(\tau)\right]^T, \tag{13}$$

$$\mathbf{R}(0) = \left[\mathbf{R}_{x_1 x_1}(0), \mathbf{R}_{x_2 x_1}(0), \ldots, \mathbf{R}_{x_m x_1}(0)\right]^T, \tag{14}$$

$$\mathbf{A} = \left[\mathbf{a}_1(\omega_1), \mathbf{a}_2(\omega_2), \ldots, \mathbf{a}_K(\omega_K)\right], \tag{15}$$

$$\begin{aligned} \mathbf{a}_k(\omega_k) = \Big[& g_1 e^{(-j(\omega_k/c)\mathbf{p}_1 \bullet \mathbf{u}_k)}, g_2 e^{(-j(\omega_k/c)\mathbf{p}_2 \bullet \mathbf{u}_k)} \\ & , \ldots, g_M e^{(-j(\omega_k/c)\mathbf{p}_M \bullet \mathbf{u}_k)} \Big]^T. \end{aligned} \tag{16}$$

The matrix $\mathbf{A}$ is the manifold matrix and $\mathbf{a}$ is the steering matrix. Then (9)–(14) can be written as

$$\mathbf{R}_f(\tau) = \mathbf{A}\boldsymbol{\Phi}_f\mathbf{R}_S(\tau), \tag{17}$$

$$\mathbf{R}_1(\tau) = \mathbf{A}\boldsymbol{\Phi}_1\boldsymbol{\Phi}_f\mathbf{R}_S(\tau), \tag{18}$$

$$\mathbf{R}_2(\tau) = \mathbf{A}\boldsymbol{\Phi}_2\boldsymbol{\Phi}_f\mathbf{R}_S(\tau), \tag{19}$$

$$\mathbf{R}_3(\tau) = \mathbf{A}\boldsymbol{\Phi}_3\boldsymbol{\Phi}_f\mathbf{R}_S(\tau), \tag{20}$$

$$\mathbf{R}(0) = \mathbf{A}\mathbf{R}_S(0), \tag{21}$$

where

$$\boldsymbol{\Phi}_f = \operatorname{diag}\left[e^{(j\omega_1\tau)}, e^{(j\omega_2\tau)}, \ldots, e^{(j\omega_K\tau)}\right]^T, \tag{22}$$

$$\boldsymbol{\Phi}_1 = \operatorname{diag}\left[\frac{g_2}{g_1}e^{(j(\omega_1/c)(\mathbf{p}_2-\mathbf{p}_1)\bullet\mathbf{u}_1)}, \frac{g_2}{g_1}e^{(j(\omega_2/c)(\mathbf{p}_2-\mathbf{p}_1)\bullet\mathbf{u}_2)} \ldots, \frac{g_2}{g_1}e^{(j(\omega_K/c)(\mathbf{p}_2-\mathbf{p}_1)\bullet\mathbf{u}_K)}\right]^T, \tag{23}$$

$$\boldsymbol{\Phi}_2 = \operatorname{diag}\left[\frac{g_3}{g_1}e^{(j(\omega_1/c)(\mathbf{p}_3-\mathbf{p}_1)\bullet\mathbf{u}_1)}, \frac{g_3}{g_1}e^{(j(\omega_2/c)(\mathbf{p}_3-\mathbf{p}_1)\bullet\mathbf{u}_2)} \ldots, \frac{g_3}{g_1}e^{(j(\omega_K/c)(\mathbf{p}_3-\mathbf{p}_1)\bullet\mathbf{u}_K)}\right]^T, \tag{24}$$

$$\boldsymbol{\Phi}_3 = \operatorname{diag}\left[\frac{g_4}{g_1}e^{(j(\omega_1/c)(\mathbf{p}_4-\mathbf{p}_1)\bullet\mathbf{u}_1)}, \frac{g_4}{g_1}e^{(j(\omega_2/c)(\mathbf{p}_4-\mathbf{p}_1)\bullet\mathbf{u}_2)} \ldots, \frac{g_4}{g_1}e^{(j(\omega_K/c)(\mathbf{p}_4-\mathbf{p}_1)\bullet\mathbf{u}_K)}\right]^T, \tag{25}$$

$$\eta_{fk} = \omega_k\tau = 2\pi f_k\tau, \tag{26}$$

$$\eta_{1k} = \frac{\omega_k}{c}(\mathbf{p}_2 - \mathbf{p}_1)\bullet u_i, \tag{27}$$

$$\eta_{2k} = \frac{\omega_k}{c}(\mathbf{p}_3 - \mathbf{p}_1)\bullet u_i, \tag{28}$$

$$\eta_{3k} = \frac{\omega_k}{c}(\mathbf{p}_4 - \mathbf{p}_1)\bullet u_i. \tag{29}$$

The incident signal is narrowband; thus, $\mathbf{R}_{s_k s_k}(n\tau) = \mathbf{R}_{s_k s_k}((n-1)\tau)$. Equation (21) can be written in another form as

$$\mathbf{R}(\tau) = \mathbf{A}\mathbf{R}_S(\tau). \tag{30}$$

Equations (17)–(21) possess the analogous form. $\mathbf{R}_f(\tau)$, $\mathbf{R}_1(\tau)$, $\mathbf{R}_2(\tau)$, $\mathbf{R}_3(\tau)$, and $\mathbf{R}(0)$ are sampled at a time delay τ_s for N ($N \geq K$) times, $\tau_s = T_s, 2T_s, \ldots, NT_s$, and T_s is the time interval of sampling which should be less than the reciprocal of the highest frequency. Thus, the aliasing effect would not happen. The pseudosnapshot data matrix can be constructed by using (17)–(21) as follows:

$$\begin{aligned}
\mathbf{R}_f &= \left[\mathbf{R}(T_s), \mathbf{R}(2T_s), \ldots, \mathbf{R}(NT_s)\right],\\
\mathbf{R}_1 &= \left[\mathbf{R}_1(T_s), \mathbf{R}_1(2T_s), \ldots, \mathbf{R}_1(NT_s)\right],\\
\mathbf{R}_2 &= \left[\mathbf{R}_2(T_s), \mathbf{R}_2(2T_s), \ldots, \mathbf{R}_2(NT_s)\right],\\
\mathbf{R}_3 &= \left[\mathbf{R}_3(T_s), \mathbf{R}_3(2T_s), \ldots, \mathbf{R}_3(NT_s)\right],\\
\mathbf{R} &= \left[\mathbf{R}_f(T_s), \mathbf{R}_f(2T_s), \ldots, \mathbf{R}_f(NT_s)\right].
\end{aligned} \tag{31}$$

Equation (31) can be written as

$$\begin{aligned}
\mathbf{R}_f &= \mathbf{A}\boldsymbol{\Phi}_f\mathbf{R}_S,\\
\mathbf{R}_1 &= \mathbf{A}\boldsymbol{\Phi}_1\boldsymbol{\Phi}_f\mathbf{R}_S,\\
\mathbf{R}_2 &= \mathbf{A}\boldsymbol{\Phi}_2\boldsymbol{\Phi}_f\mathbf{R}_S,\\
\mathbf{R}_3 &= \mathbf{A}\boldsymbol{\Phi}_3\boldsymbol{\Phi}_f\mathbf{R}_S,\\
\mathbf{R} &= \mathbf{A}\mathbf{R}_S,
\end{aligned} \tag{32}$$

where

$$\mathbf{R}_S = \left[\mathbf{R}_S(T_s), \mathbf{R}_S(2T_s), \ldots, \mathbf{R}_S(NT_s)\right]. \tag{33}$$

3.2. PARAFAC. The element of a tensor $\underline{\mathbf{X}} \in \mathscr{C}^{C\times D\times E}$ is $x_{i,j,k}$ and its P-component PARAFAC decomposition is given by

$$x_{c,d,e} = \sum_{p=1}^{M} \bar{s}_{c,p}\bar{t}_{d,p}\bar{u}_{e,p}, \tag{34}$$

where $c = 1, \ldots, C$, $d = 1, \ldots, D$, and $e = 1, \ldots, E$. $\bar{s}_{c,p}$, $\bar{t}_{d,p}$, and $\bar{u}_{e,p}$ stand for the elements of $\overline{\mathbf{S}} \in \mathscr{C}^{C\times P}$, $\overline{\mathbf{T}} \in \mathscr{C}^{D\times P}$, and $\overline{\mathbf{U}} \in \mathscr{C}^{E\times P}$, respectively. The typical elements of the $\mathbf{X}_c \in \mathscr{C}^{D\times E}$, $\mathbf{X}_d \in \mathscr{C}^{E\times C}$, and $\mathbf{X}_e \in \mathscr{C}^{C\times D}$are $X_c(d,e) := X_d(c,e) := X_e(c,d) := x_{c,d,e}$, respectively. The three way array (TWA) $\underline{\mathbf{X}}$ is "sliced" in three different directions:

$$\begin{aligned}
\mathbf{X}_c &= \overline{\mathbf{T}}\Lambda_c\left(\overline{\mathbf{S}}\right)\overline{\mathbf{U}}^T, \quad c = 1, \ldots, C,\\
\mathbf{X}_d &= \overline{\mathbf{U}}\Lambda_d\left(\overline{\mathbf{T}}\right)\overline{\mathbf{S}}^T, \quad d = 1, \ldots, D,\\
\mathbf{X}_e &= \overline{\mathbf{S}}\Lambda_e\left(\overline{\mathbf{U}}\right)\overline{\mathbf{T}}^T, \quad e = 1, \ldots, E.
\end{aligned} \tag{35}$$

For a given matrix $\overline{\mathbf{S}}$, $\Lambda_c(\overline{\mathbf{S}})$ means a diagonal matrix with the diagonals given by the cth row of the matrix. The symbol "$\odot$" is used to denote the Khatri-Rao (KR) product. As an example, the KR product of two matrices $\overline{\mathbf{S}} \in \mathscr{C}^{C\times P}$ and $\overline{\mathbf{T}} \in \mathscr{C}^{D\times P}$ of an identical number of columns is given by

$$\overline{\mathbf{S}} \odot \overline{\mathbf{T}} = \left[\bar{\mathbf{s}}_1 \otimes \bar{\mathbf{t}}_1, \ldots, \bar{\mathbf{s}}_P \otimes \bar{\mathbf{t}}_P\right] \in \mathscr{C}^{CD\times P}, \tag{36}$$

where "⊗" represents the Kronecker product. Then the definition of Kronecker product of two vectors $\bar{\mathbf{s}} \in \mathscr{C}^C$ and $\bar{\mathbf{t}} \in \mathscr{C}^D$ is given by

$$\bar{\mathbf{s}} \otimes \bar{\mathbf{t}} = \begin{bmatrix} \bar{s}_1 \bar{\mathbf{t}} \\ \bar{s}_2 \bar{\mathbf{t}} \\ \vdots \\ \bar{s}_C \bar{\mathbf{t}} \end{bmatrix}. \tag{37}$$

Based on the definition of KR product, $\mathbf{X}^{CD\times E}$ can be expressed as

$$\mathbf{X}^{CD\times E} = \begin{vmatrix} \mathbf{X}_{c=1} \\ \mathbf{X}_{c=2} \\ \vdots \\ \mathbf{X}_{c=C} \end{vmatrix} = \begin{vmatrix} \overline{\mathbf{T}} \Lambda_1 \left(\overline{\mathbf{S}}\right) \\ \overline{\mathbf{T}} \Lambda_2 \left(\overline{\mathbf{S}}\right) \\ \vdots \\ \overline{\mathbf{T}} \Lambda_C \left(\overline{\mathbf{S}}\right) \end{vmatrix} \overline{\mathbf{U}}^T = \left(\overline{\mathbf{S}} \odot \overline{\mathbf{T}}\right) \overline{\mathbf{U}}^T. \tag{38}$$

Similarly, the matrix $\mathbf{X}^{DE\times C}$ and $\mathbf{X}^{EC\times D}$ can be expressed as

$$\begin{aligned} \mathbf{X}^{DE\times C} &= \left(\overline{\mathbf{T}} \odot \overline{\mathbf{U}}\right) \overline{\mathbf{S}}^T, \\ \mathbf{X}^{EC\times D} &= \left(\overline{\mathbf{U}} \odot \overline{\mathbf{S}}\right) \overline{\mathbf{T}}^T. \end{aligned} \tag{39}$$

Definition 1 (Kruskal rank [18]). The Kruskal rank, or k-rank of a given matrix $\overline{\mathbf{S}} \in \mathscr{C}^{C\times P}$ denoted by krank $(\overline{\mathbf{S}})$ is said to be equal to r when every collection of r columns of $(\overline{\mathbf{S}})$ is linearly independent but there exists a collection of $r + 1$ linearly dependent columns. However, the rank of $\overline{\mathbf{S}}$ is said to be the maximal number of linearly independent columns. The definition of rank$(\overline{\mathbf{S}})$ is more relaxed than that of krank$(\overline{\mathbf{S}})$; that is, krank$(\overline{\mathbf{S}}) \le$ rank$(\overline{\mathbf{S}})$.

Theorem 2 (Kruskal theorem [19]). *Given a PARAFAC model $\underline{\overline{\mathbf{X}}} \in \mathscr{C}^{C\times D\times E}$; then the decomposition of this PARAFAC model is unique to permutation and scaling when*

$$k_{\overline{\mathbf{S}}} + k_{\overline{\mathbf{T}}} + k_{\overline{\mathbf{U}}} \ge 2P + 2. \tag{40}$$

The matrix $\mathbf{X}$ *is constructed by* $\widehat{\mathbf{S}}$, $\widehat{\mathbf{T}}$, *and* $\widehat{\mathbf{U}}$ *as*

$$\widehat{\mathbf{S}} = \overline{\mathbf{S}}\mathbf{\Pi}\mathbf{\Delta}_1, \qquad \widehat{\mathbf{T}} = \overline{\mathbf{T}}\mathbf{\Pi}\mathbf{\Delta}_2, \qquad \widehat{\mathbf{U}} = \overline{\mathbf{U}}\mathbf{\Pi}\mathbf{\Delta}_3, \tag{41}$$

where $\mathbf{\Pi}$ *stands for the permutation matrix.* $\mathbf{\Delta}_1$, $\mathbf{\Delta}_2$, *and* $\mathbf{\Delta}_3$ *are diagonal scaling matrices satisfying*

$$\mathbf{\Delta}_1\mathbf{\Delta}_2\mathbf{\Delta}_3 = \mathbf{I}, \tag{42}$$

where $\mathbf{I}$ *is a unit matrix.*

The $M\times M\times 5$ TWA of the conformal array is constructed based on PARAFAC model.

$$\widehat{\mathbf{R}} = \begin{vmatrix} \mathbf{R}(:,:,1) \\ \mathbf{R}(:,:,2) \\ \mathbf{R}(:,:,3) \\ \mathbf{R}(:,:,4) \\ \mathbf{R}(:,:,5) \end{vmatrix} = \begin{vmatrix} \mathbf{R}_f \\ \mathbf{R}_1 \\ \mathbf{R}_2 \\ \mathbf{R}_3 \\ \mathbf{R} \end{vmatrix} = \begin{vmatrix} \mathbf{A}\mathbf{\Phi}_f\mathbf{R}_S \\ \mathbf{A}\mathbf{\Phi}_1\mathbf{\Phi}_f\mathbf{R}_S \\ \mathbf{A}\mathbf{\Phi}_2\mathbf{\Phi}_f\mathbf{R}_S \\ \mathbf{A}\mathbf{\Phi}_3\mathbf{\Phi}_f\mathbf{R}_S \\ \mathbf{A}\mathbf{R}_S \end{vmatrix} + \widehat{\mathbf{Q}}, \tag{43}$$

where $\widehat{\mathbf{Q}}$ represents the observed noise in practice. In order to make the decomposition of the PARAFAC unique, the following assumption must be satisfied:

$$\mathbf{a}_i\left(\omega_i\right) \ne \mathbf{a}_j\left(\omega_j\right). \tag{44}$$

In other words, for two incident signals $s_i(t)$ and $s_j(t)$ with different DOAs, the following condition must be satisfied

$$g_{oe}\mathrm{e}^{-j2\pi f_i((\mathbf{p}_{oe}\bullet\mathbf{u}_i)/c)} \ne g_{oe}\mathrm{e}^{-j2\pi f_j((\mathbf{p}_{oe}\bullet\mathbf{u}_j)/c)}. \tag{45}$$

The k-rank of matrix $\mathbf{D}$, $\mathbf{A}$, and $\mathbf{R}_S$ are $k_D = \min(5, K)$, $k_A = K$, and $k_{R_S} = K$, respectively. Thus, when $K \ge 2$, the k-rank decomposition of the PARAFAC is unique.

Based on the definition of KR product, (43) can be written in three forms as follows:

$$\mathbf{R} = (\mathbf{D} \odot \mathbf{A})\,\mathbf{R}_S + \widehat{\mathbf{Q}}, \tag{46}$$

$$\mathbf{R}_X = \left(\mathbf{R}_S^T \odot \mathbf{D}\right)\mathbf{A}^T + \widehat{\mathbf{Q}}_X, \tag{47}$$

$$\mathbf{R}_Y = \left(\mathbf{A} \odot \mathbf{R}_S^T\right)\mathbf{D}^T + \widehat{\mathbf{Q}}_Y, \tag{48}$$

where

$$\mathbf{D} = \begin{vmatrix} \Lambda^{-1}\left(\mathbf{\Phi}_f\right) \\ \Lambda^{-1}\left(\mathbf{\Phi}_1\mathbf{\Phi}_f\right) \\ \Lambda^{-1}\left(\mathbf{\Phi}_2\mathbf{\Phi}_f\right) \\ \Lambda^{-1}\left(\mathbf{\Phi}_3\mathbf{\Phi}_f\right) \\ \Lambda^{-1}(1) \end{vmatrix}, \tag{49}$$

where $\Lambda^{-1}(\mathbf{\Phi}_f)$ means the row vector constructed by the diagonal elements of the diagonal matrix $\mathbf{\Phi}_f$.

The matrices $\mathbf{D}$, $\mathbf{A}$, and $\mathbf{R}_S$ can be solved by the trilinear alternating least squares (TALS) regression algorithm. Due to the existence of the observed noise, (46) could be transformed as a least square problem as follows:

$$\min_{\mathbf{D},\mathbf{A},\mathbf{R}_S} \left\|\widehat{\mathbf{R}} - (\mathbf{D} \odot \mathbf{A})\,\mathbf{R}_S\right\|_F^2. \tag{50}$$

Then the least square estimator of $\mathbf{R}_S$ can be expressed as

$$\mathbf{R}_S = \arg\min_{\mathbf{R}_S} \left\|\widehat{\mathbf{R}} - (\mathbf{D} \odot \mathbf{A})\,\mathbf{R}_S\right\|_F^2. \tag{51}$$

Similarly, the solutions of $\mathbf{D}$ and $\mathbf{A}$ can be expressed as

$$\begin{aligned} \mathbf{A}^T &= \left(\mathbf{R}_S^T \odot \mathbf{D}\right)^\dagger \mathbf{R}_X \\ \mathbf{D}^T &= \left(\mathbf{A} \odot \mathbf{R}_S^T\right)^\dagger \mathbf{R}_Y. \end{aligned} \tag{52}$$

We can update the matrices $\mathbf{D}$, $\mathbf{A}$, and $\mathbf{R}_S$ alternately until the algorithm converges; the estimators of the three matrices can be acquired. In the following experiments, the COMFAC algorithm (the TALS regression algorithm is realized in practice) is used to fit the $m \times m \times 5$ TWA. The initialization and fitting of the COMFAC is done in compressed space. The Tucker 3 three-way model is used in data compression [21].

Now the ALS algorithm steps can be summarized as follows.

(1) Initialize $\mathbf{A}^{(0)} \in \mathscr{C}^{M\times K}$ and $\mathbf{D}^{(0)} \in \mathscr{C}^{5\times K}$.

(2) Initialize $\varepsilon > 0$ and $k = 0$.

(3) If $\|\rho^{(k+1)} - \rho^{(k)}\|/\rho^{(k)} > \varepsilon$, calculate matrices $\mathbf{A}$, $\mathbf{R}_S$, and $\mathbf{D}$ by (51)-(52). Update just one matrix at each time; then $k \rightarrow k+1$.

(4) Else $\|\rho^{(k+1)} - \rho^{(k)}\|/\rho^{(k)} < \varepsilon$; the iteration is terminated.

In this paper, ρ stands for $\mathbf{A}$, $\mathbf{R}_S$, and $\mathbf{D}$, respectively.

3.3. The Joint Parameter Estimation. The matrix $\mathbf{D}$ can be estimated by TALS regression algorithm; then the estimator of ω_{fk} is given by

$$\omega_{ki} = \text{angle}\left|\frac{\mathbf{D}_{1k}}{\mathbf{D}_{5k}}\right|. \tag{53}$$

Because r_1 and r_2 are real numbers, $\mathbf{D}_{2k}/\mathbf{D}_{1k}$ is squared to avoid the ambiguity caused by the positive and negative values of r_1 and r_2. Then

$$\begin{aligned}\omega_{1k} &= -\frac{1}{2}\text{angle}\left(\left[\frac{g_2(\theta_k,\varphi_k)}{g_1(\theta_k,\varphi_k)}\exp(-j\omega_{1k})\right]^2\right)\\ &= -\frac{1}{2}\text{angle}\left(\exp(-j2\omega_{1k})\right)\\ &= -\frac{1}{2}\text{angle}\left(\left[\frac{\mathbf{D}_{2k}}{\mathbf{D}_{1k}}\right]^2\right).\end{aligned} \tag{54}$$

Similarly

$$\omega_{2k} = -\frac{1}{2}\text{angle}\left(\left[\frac{\mathbf{D}_{3k}}{\mathbf{D}_{1k}}\right]^2\right) \tag{55}$$

$$\omega_{3k} = -\frac{1}{2}\text{angle}\left(\left[\frac{\mathbf{D}_{4k}}{\mathbf{D}_{1k}}\right]^2\right). \tag{56}$$

Taking (26) into (53), the frequency f_k of the kth incident signal is expressed as

$$f_k = \frac{1}{2\pi\tau}\text{angle}\left|\frac{\mathbf{D}_{1k}}{\mathbf{D}_{5k}}\right|. \tag{57}$$

Assuming that $\gamma_{1k} = \sin(\theta_k)\cos(\varphi_k)$, $\gamma_{2k} = \sin(\theta_k)\sin(\varphi_k)$, and $\gamma_{3k} = \cos(\theta_k)$, and solving (27), (28), (29), (55), (56), and (57) simultaneously, we can obtain the equation as follows:

$$\frac{c}{2\pi f_k}\begin{bmatrix}\frac{\omega_{1k}}{d_1}\\ \frac{\omega_{2k}}{d_2}\\ \frac{\omega_{3k}}{d_3}\end{bmatrix} = \begin{bmatrix}x_2-x_1 & y_2-y_1 & z_2-z_1\\ x_3-x_1 & y_3-y_1 & z_3-z_1\\ x_4-x_1 & y_4-y_1 & y_4-y_1\end{bmatrix}\begin{bmatrix}\gamma_{1k}\\ \gamma_{2k}\\ \gamma_{3k}\end{bmatrix}. \tag{58}$$

The solution of the equation is represented as

$$\begin{bmatrix}\gamma_{1k}\\ \gamma_{2k}\\ \gamma_{3k}\end{bmatrix} = \frac{c}{2\pi f_i}\begin{bmatrix}x_2-x_1 & y_2-y_1 & z_2-z_1\\ x_3-x_1 & y_3-y_1 & z_3-z_1\\ x_4-x_1 & y_4-y_1 & y_4-y_1\end{bmatrix}^{-1}\begin{bmatrix}\frac{\omega_{1k}}{d_1}\\ \frac{\omega_{2k}}{d_2}\\ \frac{\omega_{3k}}{d_3}\end{bmatrix}. \tag{59}$$

There are two approaches to solve θ_k and φ_k. The first approach is

$$\begin{aligned}\theta_k &= \arccos\gamma_{3k},\\ \varphi_k &= \arcsin\left(\frac{\gamma_{1k}}{\cos(\theta_k)}\right) \text{ or } \varphi_k = \arcsin\left(\frac{\gamma_{2k}}{\sin(\theta_k)}\right).\end{aligned} \tag{60}$$

The second approach is

$$\begin{aligned}\varphi_k &= \arctan\left(\frac{\gamma_{2k}}{\gamma_{1k}}\right),\\ \theta_k &= \arccos\left(\frac{\gamma_{1k}}{\cos(\varphi_k)}\right) \text{ or } \theta_k = \arcsin\left(\frac{\gamma_{2k}}{\sin(\varphi_k)}\right).\end{aligned} \tag{61}$$

In this paper the first approach is employed.

In order to ensure the uniqueness of the estimated parameters, the following condition must be satisfied:

$$\begin{aligned}\|\mathbf{p}_2 - \mathbf{p}_1\| &\le \frac{c}{(2\max(f_i))} = \frac{\min(\lambda_k)}{2},\\ \|\mathbf{p}_3 - \mathbf{p}_1\| &\le \frac{c}{(2\max(f_i))} = \frac{\min(\lambda_k)}{2},\\ \|\mathbf{p}_4 - \mathbf{p}_1\| &\le \frac{c}{(2\max(f_i))} = \frac{\min(\lambda_k)}{2}.\end{aligned} \tag{62}$$

According to Theorem 2, the matrices $\mathbf{D}$, $\mathbf{A}$, and $\mathbf{R}_S$ have the same permutation matrix, that is, the kth column of matrix $\mathbf{D}$ corresponds to the kth column of the manifold matrix $\mathbf{A}$.

Based on the PARAFAC model and TALS regression algorithm, the joint frequency and 2D-DOA estimation can be accomplished. The algorithm steps are summarized as follows:

(1) choose four guiding elements and obtain the accurate positions of the elements;

(2) calculate the delay correlation function between the four elements mentioned above and the whole elements of the conformal array;

(3) calculate the matrices $\mathbf{R}_f(\tau)$, $\mathbf{R}_1(\tau)$, $\mathbf{R}_2(\tau)$, $\mathbf{R}_3(\tau)$, and $\mathbf{R}(0)$, respectively according to (5);

(4) calculate the pseudosnapshot data matrix based on (17)–(21). $\mathbf{R}_f(\tau)$, $\mathbf{R}_1(\tau)$, $\mathbf{R}_2(\tau)$, $\mathbf{R}_3(\tau)$, and $\mathbf{R}(0)$ are sampled at time delay τ_s, respectively;

(5) construct the PARAFAC model based on (43);

(6) estimate the matrix $\mathbf{D}$ using TALS regression algorithm;

(7) obtain ω_{fk}, ω_{1k}, ω_{2k}, and ω_{3k} by using the estimator of the matrix $\mathbf{D}$ and calculating (53)–(56);

(8) obtain the joint frequency and 2D-DOA estimation by calculate (57)–(59) or (57), (60), and (61).

4. Simulation Results

This section focuses on several sets of simulation results to demonstrate the performance of the proposed algorithm. In all simulation examples, 200 Monte Carlo trials are considered, and the root mean square error (RMSE) is used as our performance measure, which is defined as

$$\text{RMSE} = \sqrt{\frac{1}{200}\sum_{\eta=1}^{200}\left[\left(\widehat{\theta}_{k,\eta}-\theta\right)^2+\left(\widehat{\varphi}_{k,\eta}-\varphi\right)^2\right]}, \quad (63)$$

where $\widehat{\theta}_{i,t}$ and $\widehat{\varphi}_{i,t}$ are the elevation and azimuth estimators of the kth incident signal in the ηth trial.

The successful rate is defined as the proportion of the number of the successful experiments to the total number of the experiments. For frequency estimation, a successful experiment is defined as the experiment with estimation error of less than 2 MHz; for DOA estimation, a successful experiment is defined as the experiment with estimation error of less than 2 degree.

In order to simplify the transformation from the global coordinate to the local coordinate, only the cylinder with single curvature is considered in the simulations. The array structure is shown in Figure 2 is only a subarray, which only covers 120°. Then three subarrays can estimate the whole space. The highest frequency of the incident signal is $f_{\max}$ = 2 GHz, $\lambda_{\min} = c/f_{\max}$. The radius of the cylinder is $5\lambda_{\min}$, the adjacent element spacing in the same intersecting surface of the cylinder is $\lambda_{\min}/4$, and the adjacent intersecting surface spacing is $\lambda_{\min}/4$, $\lambda_{\min}$ stands for the shortest wavelength of the incident signal. The position of the reference element is $p_1 = (0, 5\lambda, 0)$.

The element pattern is defined in the local coordinate of each element; thus, the transformation from the global coordinate to the local coordinate of elevation θ_k and azimuth φ_k of the incident signal must be accomplished. For the cylindrical conformal array, the corresponding Euler rotation angles are

$$D = -\Theta, \qquad E = -\pi/2, \qquad F = 0, \quad (64)$$

respectively. D represents the rotation angle based on the right-hand rule in the first rotation and the Z-axis is the rotation axis; E represents the rotation angle in the second rotation and the Y-axis is the rotation axis; and F represents the rotation angle in the third rotation and the X-axis is the rotation axis. Θ represents the angle between the position vector of the element and the positive direction of X-axis.

The unit vector of the kth incident signal (θ_k, φ_k) in the global coordinate can be expressed as

$$x = \sin(\theta_k)\cos(\varphi_k), \qquad y = \sin(\theta_k)\sin(\varphi_k), \qquad z = \cos(\theta_k). \quad (65)$$

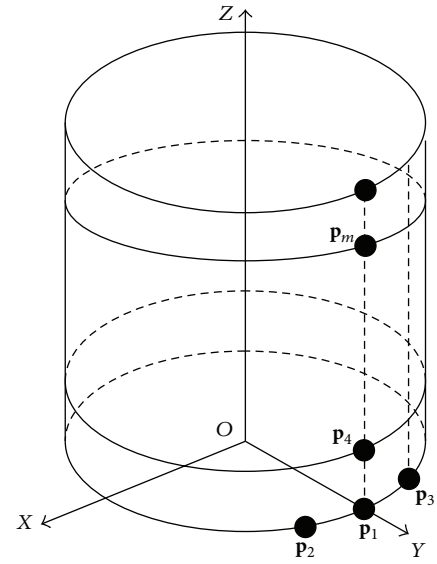


Figure 2: The structure of the cylindrical conformal array.

Based on the Euler rotation matrix, the unit vector in global coordinate can be transformed into the local coordinate of the mth element.

Consider

$$\begin{bmatrix} x_m & y_m & z_m \end{bmatrix}^T = R\left(D_m, E_m, F_m\right)\begin{bmatrix} x & y & z \end{bmatrix}^T, \quad (66)$$

where

$$R\left(D_m, E_m, F_m\right) = \begin{bmatrix} \cos F & \sin F & 0 \\ -\sin F & \cos F & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos E & 0 & -\sin E \\ 0 & 1 & 0 \\ \sin E & 0 & \cos E \end{bmatrix} \times \begin{bmatrix} \cos D & \sin D & 0 \\ -\sin D & \cos D & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (67)$$

Then the elevation θ and azimuth φ of the kth incident signal in the local coordinate of the mth element can be represented as

$$\varphi_{mk} = \arctan\left(y_m/x_m\right), \qquad \theta_{mk} = \arccos\left(z_m\right). \quad (68)$$

Thus, the element pattern transformation from the global coordinate to the local coordinate is completed.

First, the curves of RMSE of frequency and angle and the successful probability of angle in different SNR are plotted in Figure 3 using the proposed algorithm. The number of snapshots is 200, the number of pseudosnapshots is 100. SNR $\in [5, 30]$, and the noise is AWGN which is independent of incident signals. The sampling frequency is $f_s = 1/T_s$ = 5 GHz. The elevation θ, azimuth φ, and frequency of these two narrow incident signals are (100°, 60°, 1 GHz) and (95°, 50°, 2 GHz), respectively, which are independent of each other.

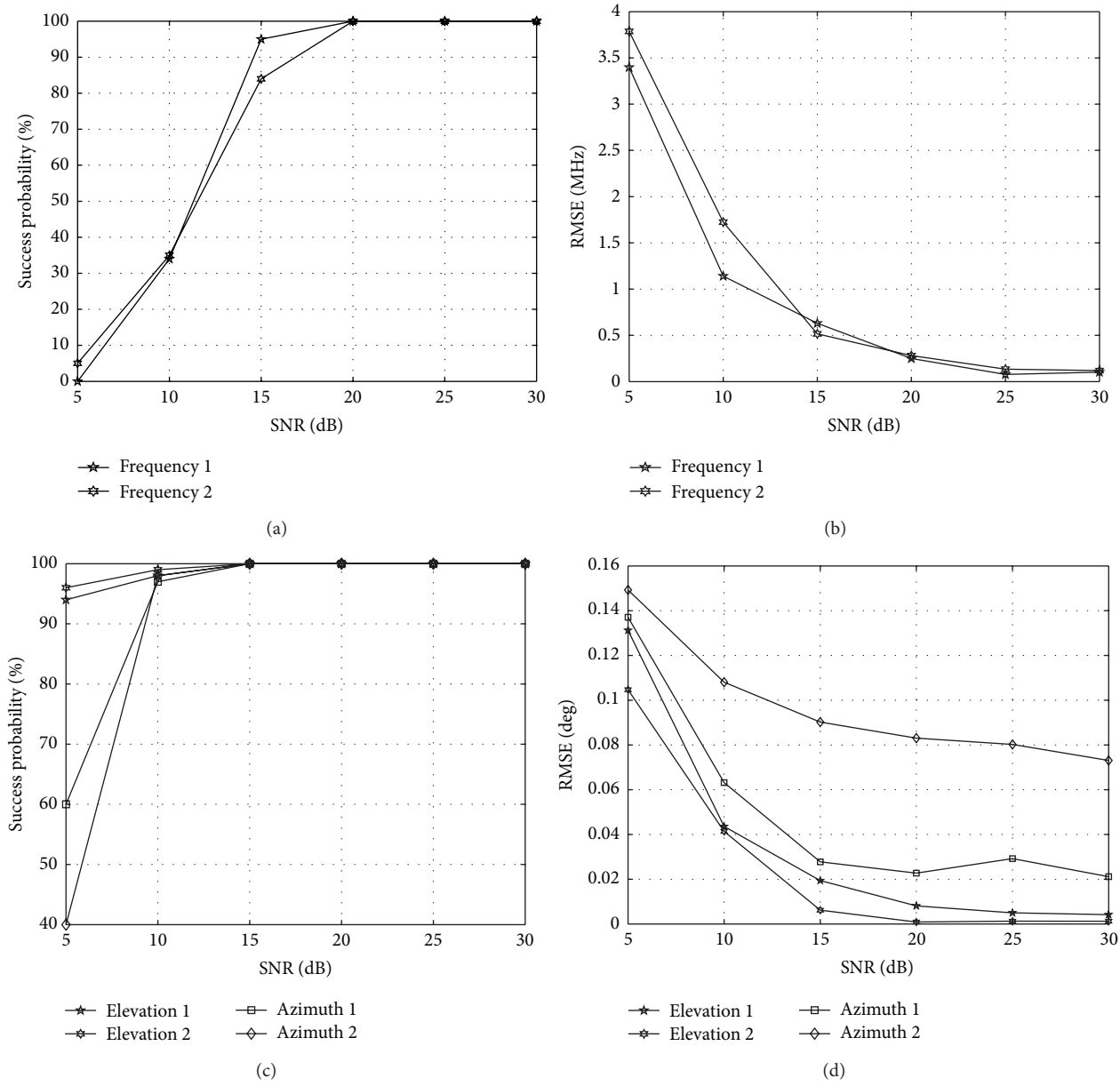


Figure 3: The RMSE and successful probability with different SNR. (a) The successful probability of frequencies, (b) the RMSE of frequency, (c) the successful probability of angles, and (d) the RMSE of angles.

The number of elements are 10, including 4 guiding elements and other 6 instrumental elements. The element pattern used in simulation is the lowest order circular patch model [8, 22]

$$
\begin{aligned}
g_\theta(\theta,\varphi) &= \left\{J_2\left(\frac{\pi d}{\lambda}\sin\theta\right) - J_0\left(\frac{\pi d}{\lambda}\sin\theta\right)\right\} \\
&\quad \times (\cos\varphi - j\sin\varphi), \quad 0 \le \theta \le \frac{\pi}{2}, \\
g_\varphi(\theta,\varphi) &= \left\{J_2\left(\frac{\pi d}{\lambda}\sin\theta\right) + J_0\left(\frac{\pi d}{\lambda}\sin\theta\right)\right\} \\
&\quad \times \cos\theta(\sin\varphi - j\cos\varphi), \quad 0 \le \theta \le \frac{\pi}{2}, \\
g_\theta(\theta,\varphi) &= g_\varphi(\theta,\varphi) = 0, \quad \text{otherwise},
\end{aligned}
\tag{69}
$$

where J_0 and J_2 are the zeroth- and second-order Bessel functions of the first kind. The element pattern transformation is achieved with the method proposed above.

Figure 3 shows that the RMSE of frequency and angle decreases while the SNR increases; meanwhile, the success probability of frequency and angle increase as the SNR increases. The estimated performance of the two frequencies are very close which can be seen in Figures 3(a) and 3(b). As shown in Figure 3(a), the successful probability is larger than 80%, when the SNR is larger than 15 dB. The RMSE of frequency is less than 2 MHz, when SNR is larger than 10 dB as shown in Figure 3(b). In Figure 3(c) the successful probability of azimuths is slightly lower than that of the elevations. The success probability of the angle is almost

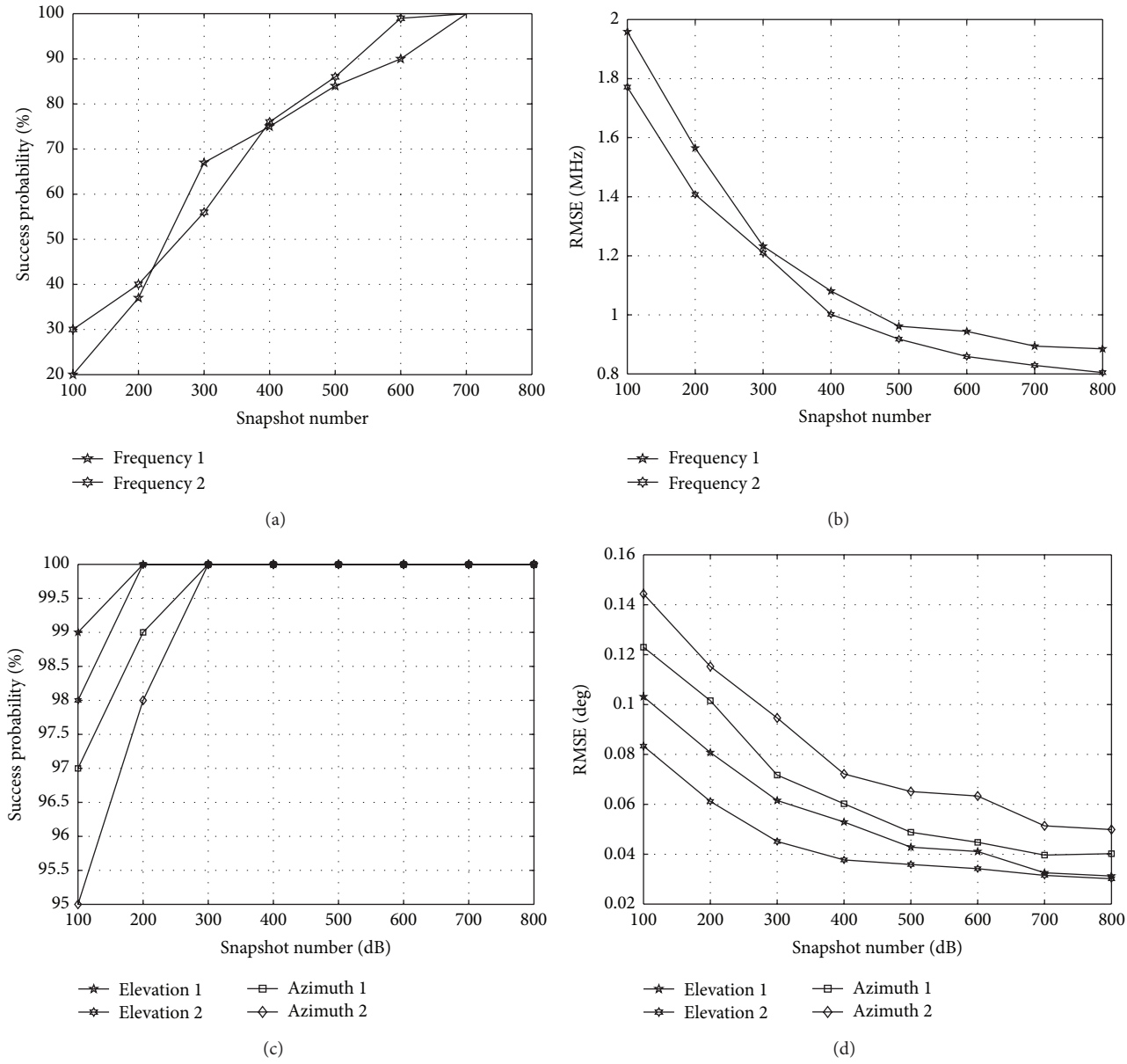


Figure 4: The RMSE and successful probability with different snapshot number. (a) The successful probability of frequencies, (b) the RMSE of frequency, (c) the successful probability of angles, and (d) the RMSE of angles.

100% when the SNR reaches 15 dB. Figure 3(d) shows that the RMSE is smaller than 0.16° when SNR is larger than 5 dB. The RMSE of azimuth is larger than that of elevation. The estimation results of frequency and elevation are applied to estimate azimuth. The phenomenon in Figure 3(d) can be interpreted as the existence of the estimated error of frequency and elevation. Next, the RMSE of frequency and angle as well as the success probability of angle in different snapshot number are depicted in Figure 4. SNR = 10 dB; other simulation conditions are identical with the previous experiment. In Figure 4, the RMSE of frequency and angle decreases as the snapshot number increases, and the success probability of frequency and angle rises as the snapshot number increases. As shown in Figure 4(a), the successful probability is larger than 80%, when the number of snapshot is larger than 500. Figure 4(b) shows that the RMSE of frequency 1 is slightly larger than that of frequency 2. When the snapshot number is larger than 500, the RMSE drops below 1 MHz, meaning that the proposed algorithm achieved three orders of magnitude reduction. Figures 4(c) and 4(d) show that the success probability of azimuth is lower than that of the elevation, and the RMSE of azimuth are larger than that of elevation. The reason why this happens is because the estimation error of frequency and elevation affects the estimation performance of azimuth.

5. Conclusion

Due to the varying curvature of the surface of the conformal carrier, the pattern of each element of the conformal array

is different. Thus, the conventional algorithms could not be used for conformal array. In this paper, a novel joint frequency and 2D-DOA estimation algorithm with high accuracy are proposed. Both spatial and time sampling are utilized to construct the spatial-time matrix. The delay correlation function is used to suppress noise. The PARAFAC model is used for parameter estimation without parameter pairing. Only four elements are needed, and the positions of these elements should be known accurately. Other instrumental elements can be flexibly arranged on the surface of the conformal carrier. The algorithm proposed in this paper can be extended to estimate 2D-DOA straightly with little modification. The simulation results verify the effectiveness of the proposed algorithm. It can be expected that the proposed algorithm would have an application prospect in the parameter estimation of conformal array. In the future work, we will focus on the application of the proposed algorithm [23–26].

Acknowledgments

This work was supported in part by the National Science Foundation of China under Grant 61201410 and in part by Fundamental Research Focused on Special Fund Project of the Central Universities (Program no. HEUCF130804).

References

[1] L. Josefsson and P. Persson, *Conformal Array Antenna Theory and Design*, IEEE Press, Piscataway, NJ, USA, 2006.

[2] W. T. Li, X. W. Shi, Y. Q. Hei, S. F. Liu, and J. Zhu, "A hybrid optimization algorithm and its application for conformal array pattern synthesis," *IEEE Transactions on Antennas and Propagation*, vol. 58, no. 10, pp. 3401–3406, 2010.

[3] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[4] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984–995, 1989.

[5] P. Hyber, M. Jansson, and B. Ottersten, "Array interpolation and bias reduction," *IEEE Transaction on Signal Processing*, vol. 52, no. 10, pp. 2711–2720, 2004.

[6] P. Hyberg, M. Jansson, and B. Ottersten, "Array interpolation and DOA MSE reduction," *IEEE Transactions on Signal Processing*, vol. 53, no. 12, pp. 4464–4471, 2005.

[7] F. Belloni, A. Richter, and V. Koivunen, "DOA estimation via manifold separation for arbitrary array structures," *IEEE Transactions on Signal Processing*, vol. 55, no. 10, pp. 4800–4810, 2007.

[8] B.-H. Wang, Y. Guo, Y.-L. Wang, and Y.-Z. Lin, "Frequency-invariant pattern synthesis of conformal array antenna with low cross-polarisation," *IET Microwaves, Antennas and Propagation*, vol. 2, no. 5, pp. 442–450, 2008.

[9] P. Yang, F. Yang, and Z. Nie, "DOA estimation using MUSIC algorithm on a cylinderical conformal array," in *Proceedings of the IEEE Antennas and Propagation Society International Symposium*, pp. 5299–5302, June 2007.

[10] P. Yang, F. Yang, and Z. P. Nie, "DOA estimation with sub-array divided technique and interporlated esprit algorithm on a cylindrical conformal array antenna," *Progress in Electromagnetics Research*, vol. 103, pp. 201–216, 2010.

[11] Z.-S. Qi, Y. Guo, and B.-H. Wang, "Blind direction-of-arrival estimation algorithm for conformal array antenna with respect to polarisation diversity," *IET Microwaves, Antennas and Propagation*, vol. 5, no. 4, pp. 433–442, 2011.

[12] L. Zou, J. Lasenby, and Z. He, "Direction and polarization estimation using polarized cylindrical conformal arrays," *IET Signal Processing*, vol. 6, no. 5, pp. 395–403, 2012.

[13] Y.-Y. Wang, J.-T. Chen, and W.-H. Fang, "TST-MUSIC for joint DOA-delay estimation," *IEEE Transactions on Signal Processing*, vol. 49, no. 4, pp. 721–729, 2001.

[14] J.-D. Lin, W.-H. Fang, Y.-Y. Wang, and J.-T. Chen, "FSF MUSIC for joint DOA and frequency estimation and its performance analysis," *IEEE Transactions on Signal Processing*, vol. 54, no. 12, pp. 4529–4542, 2006.

[15] M. Djeddou, A. Belouchrani, and S. Aouada, "Maximum likelihood angle-frequency estimation in partially known correlated noise for low-elevation targets," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 3057–3064, 2005.

[16] R. B. Cattell, "'Parallel proportional profiles' and other principles for determining the choice of factors by rotation," *Psychometrika*, vol. 9, no. 4, pp. 267–283, 1944.

[17] J. D. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, 1970.

[18] N. D. Sidiropoulos, G. B. Giannakis, and R. Bro, "Blind PARAFAC receivers for DS-CDMA systems," *IEEE Transactions on Signal Processing*, vol. 48, no. 3, pp. 810–823, 2000.

[19] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis, "Parallel factor analysis in sensor array processing," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2377–2388, 2000.

[20] T. Milligan, "More applications of euler rotation," *IEEE Antennas and Propagation Magazine*, vol. 41, no. 4, pp. 77–83, 1999.

[21] N. D. Sidiropoulos, "COMFAC: Matlab Code for LS Fitting of the Complex PARAFAC Model in 3-D," 1998, http://www.telecom.tuc.gr/~nikos/.

[22] J. R. James, P. S. Hall, and C. Wood, *Microstrip Antenna Theory and Design*, Peter Peregrinus, New York, NY, USA, 1981.

[23] S. Yin, S. Ding, and H. Luo, "Real-time implementation of fault tolerant control system with performance optimization," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 5, pp. 2402–2411, 2014.

[24] S. Yin, S. Ding, A. Haghani, H. Hao, and P. Zhang, "A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process," *Journal of Process Control*, vol. 22, no. 9, pp. 1567–1581, 2012.

[25] S. Yin, S. Ding, A. Haghani, and H. Hao, "Data-driven monitoring for stochastic systems and its application on batch process," *International Journal of Systems Science*, vol. 44, no. 7, pp. 1366–1376, 2013.

[26] S. Yin, X. Yang, and H. Karimi, "Data-driven adaptive observer for fault diagnosis," *Mathematical Problems in Engineering*, vol. 2012, Article ID 832836, 21 pages, 2012.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 185970, 8 pages
http://dx.doi.org/10.1155/2013/185970

Research Article

The Robustness Analysis of Wireless Sensor Networks under Uncertain Interference

Changjian Deng[1,2]

[1] *School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China*
[2] *Department of Control Engineering, Chengdu University of Information Technology, Chengdu 610225, China*

Correspondence should be addressed to Changjian Deng; chengli_dcj@163.com

Received 9 October 2013; Accepted 17 November 2013

Academic Editors: H. R. Karimi and W. Zhang

Based on the complex network theory, robustness analysis of condition monitoring wireless sensor network under uncertain interference is present. In the evolution of the topology of sensor networks, the density weighted algebraic connectivity is taken into account, and the phenomenon of removing and repairing the link and node in the network is discussed. Numerical simulation is conducted to explore algebraic connectivity characteristics and network robustness performance. It is found that nodes density has the effect on algebraic connectivity distribution in the random graph model; high density nodes carry more connections, use more throughputs, and may be more unreliable. Moreover, the results show that, when network should be more error tolerant or robust by repairing nodes or adding new nodes, the network should be better clustered in median and high scale wireless sensor networks and be meshing topology in small scale networks.

1. Introduction

Currently, wireless sensor networks have been deployed for condition monitoring application. In industrial harsh environment, there are many kinds of uncertain interference, for example, energy dependence, dynamic topological update, and varying large number of nodes, and these make WSN a type of complex system.

Under uncertain industrial environment, robustness is an important property. Robustness is often defined as invariance degree of state, behavior, and function or the adaptation/flexibility degree under interference of perturbations. Robust analysis of wireless sensor networks is intractable and challenging.

There are three models of complex network [1–3]. The first model is the Erdős-Rényi model of random graphs, the second model is small-world model, and the third model is scale-free model of the power-law degree distribution.

Papers [4–12] proved that many complex systems display a surprising degree of tolerance for errors. Robustness of wireless complex networks can be enhanced by optimization of networks topology or by repair of its faults.

Papers [13–18] discuss that second smallest eigenvalue of the Laplacian matrix (algebraic connectivity) plays a special role for the robustness of networks using the Erdős-Rényi random graph, as an example for the model of condition monitoring wireless sensor networks.

Papers [19–27] study the ability to control networks. Recent work has extended the concept of pinning control and structural controllability to complex networks and so on.

With the fundamentals of these, the paper focuses on the research of topology choice and repairing control based on density weighted algebraic connectivity; when the vertex and links are not always constant, they can change with time.

The contributions of the paper are (1) weighted and changeable algebraic connectivity analysis in random network and (2) presenting a method to do topology choice and repairing control of different topology of wireless sensors, for example, the star topology, the tree-cluster topology, the mash topology, and so on.

The paper is arranged as follows: in Section 2, the related works are introduced; in Section 3, the research method of the paper is described; in Section 4, the simulation and its discussion are presented; in Section 5, the conclusion is given.

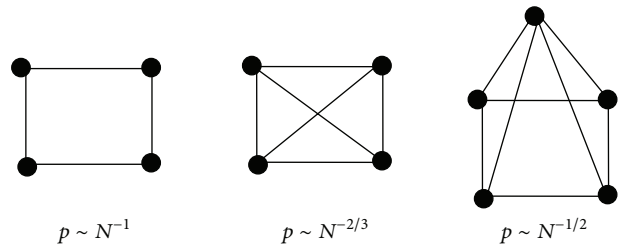


Figure 1: Threshold probabilities at which different subgraphs appear in a random graph.

2. Related Work

2.1. Statistics Results of Erdős and Rényi Model. A network is represented as an undirected graph $G = (N, L)$ consisting of $N = |N|$ nodes and $L = |L|$ links.

Erdős and Rényi define a random graph as N labeled nodes connected by n edges, which are chosen randomly from the $N(N-1)/2$ possible edges. In total there are $C^n_{[N(N-1)/2]}$ graphs with N nodes and n edges. An alternative and equivalent definition of a random graph is the binomial model, and following properties of the random graph can be determined asymptotically [1].

(i) In ER model, there is a critical probability $p_c(N)$. If $p(N)$ grows more slowly than $p_c(N)$ as $N \to \infty$, then almost every graph with connection probability $p(N)$ fails to have property Q. If $p(N)$ grows somewhat faster than $p_c(N)$, then almost every graph has the property Q.

(ii) The critical probability at which almost every graph contains a subgraph with k nodes and edges is $p_c(N) = cN^{-k/l}$. (a) The critical probability of having a tree of order k is $p_c(N) = cN^{-k(k-1)}$; (b) the critical probability of having a cycle of order k is $p_c(N) = cN^{-1}$; (c) the critical probability of having a complete subgraph of order k is $p_c(N) = cN^{-2/(k-1)}$.

(iii) The useful threshold probabilities for WSN at which different subgraphs appear in a random graph are shown in Figure 1. At $p \sim N^{-1}$ trees of all orders are present, and at the same time cycles of all orders appear. The probability $p \sim N^{-2/3}$ marks the appearance of complete subgraphs of order 4; $p \sim N^{-1/2}$ corresponds to complete subgraphs of order 5. As z approaches 0, the graph contains complete subgraphs of increasing order.

(iv) The expectation value of the number of nodes with degree k has a passion distribution.

(v) A general conclusion is that, for most values of p, almost all graphs with the same N and p have precisely the same diameter.

(vi) The clustering coefficient of a random graph is formula

$$C_{\text{rand}} = p = \frac{\langle k \rangle}{N}. \tag{1}$$

2.2. Algebraic Connectivity of Erdős and Rényi Model. The Laplacian matrix of a graph G with N nodes is an $N \times N$ matrix $Q = \Delta - A$, where $\Delta = \text{diag}(D_i)$. D_i denotes the nodal degree of the node $i \in N$ and A is the adjacency matrix of G.

The asymptotic behavior of the algebraic connectivity in the Erdős-Rényi random graph $G_p(N)$ is as follows: for any $\varepsilon > 0$,

$$\lambda_{N-1} = pN + o\left(N^{1/2+\varepsilon}\right), \tag{2}$$

where the algebraic connectivity converges in probability as $N \to \infty$.

Paper [18] defined that the correlation coefficient of the degrees D_i and D_j of two random nodes i and j in $G_p(N)$ for $0 < p < 1$ is

$$\rho\left(D_i, D_j\right) = \frac{\text{Cov}\left[D_i, D_j\right]}{\sqrt{\text{Var}\left[D_i\right]}\sqrt{\text{Var}\left[D_j\right]}} = \frac{1}{N-1}. \tag{3}$$

At large graph size N, the distribution of the algebraic connectivity will rapidly approach the mean value. And the distribution of the algebraic connectivity grows linearly with the minimum nodal degree, $\lambda_{N-1} \approx D_{\min}$.

And the larger the algebraic connectivity is, the better the graph's robustness of node and link failures is.

3. Weighted and Changeable Algebraic Connectivity of WSN

3.1. Weighted Algebraic Connectivity. Figure 2 shows the simplest topology of normal equipment condition monitoring WSN.

In Figure 2, the vertex number of graph is seven. There is a very interesting phenomenon that if we omitted the vertex of sink node, the network should be almost connected with equal probability p (k of N nodes).

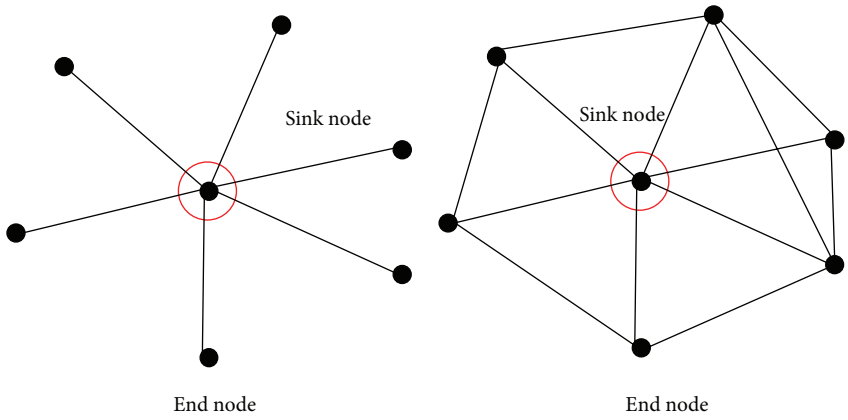
In time domain, this omitted network graph can be looked as a random graph. (For the data transmit, the connected link is at probability p.)

Here the average degree of a vertex in this network is

$$p_k = \binom{N-1}{k} p^k (1-p)^{N-1-k} \cong \frac{z^k}{k!} e^{-z}. \tag{4}$$

If N were large, this distribution should be looked to as the Poisson distribution.

(1) Medium Complexity Topology of Normal Condition Monitoring WSN. As shown in Figure 3, the vertex number of graph is almost 21. In the left and right of Figure 3, the density of a network (number of nodes in an area) is the same, but possibility of links is different. There is also a phenomenon that if the vertex of route node was omitted, the network should be almost connected with equal probability p if they had the same density. And, if the density is large, the vertex degree of graph is large.



(a) Star topology of WSN (b) Single hop mesh topology of WSN

Figure 2: Simplest topology of normal condition monitoring WSN.

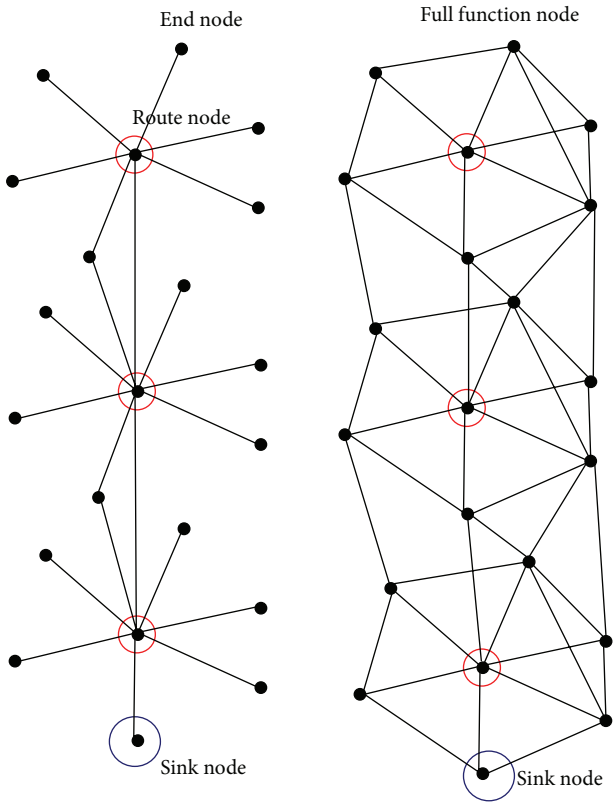


Figure 3: Modern complexity topology of normal condition monitoring WSN.

The omitted network graph can also be looked to as a random graph. And the network made from route nodes and sink nodes has similar property with mesh network.

(2) Large Complexity Topology of Condition Monitoring WSN. When there are hundreds of nodes in a wireless condition monitoring network and the network consists of many similar areas that have independent functions, and then we can define this every area as a cell of network. Then if the cell of network has the same topology as Figure 2, then it has the similar results as (1).

The eigenvalues of Q are called the Laplacian eigenvalues. The Laplacian eigenvalues $\lambda_N = 0 \leqslant \lambda_{N-1} \leqslant \cdots \leqslant \lambda_1$ are all real and nonnegative. The second smallest Laplacian eigenvalue λ_{N-1} is also known as the algebraic connectivity.

Lemma 1. *Algebraic connectivity of wireless star, cluster-tree, or mesh network has below properties:*

For an ideal full-function mesh network, algebraic connectivity is proportional with density of network.

For an ideal star or cluster-tree network, the end nodes have similar algebraic connectivity; algebraic connectivity of the route nodes and AP (or coordinator) is proportional with density of network.

For complex WSN, algebraic connectivity has a weight; the simplest example of the weight is density of network.

3.2. The Influence Function of Weighted Algebraic Connectivity. To discuss the robustness of WSN, here two connectivity metrics of G are introduced: (1) the link (edge) connectivity κ_L is the minimal number of links whose removal would disconnect G; (2) the node (vertex) connectivity κ_N is defined analogously (nodes together with adjacent links are removed).

The robustness of network graph has a relationship with algebraic connectivity; the algebraic connectivity of a graph is increased with the node and the link connectivity. In another way, robustness has a relationship with error tolerance of network. Two types of node removal were considered. The first type was that all the nodes were randomly removed. The second type was that the most highly connected nodes were of more reliability nodes; other was the same as type 1.

Definition 2 (influence function set of network). In graph R of network, if the subgraph R_1 is not connected with Sink (or network access point node) node, then subgraph is R_1 graph, and connected graph is R-R_1 graph.

If R_1 occurred from removal of node k, then ratio of $E_k = R_1/R$ means the destroy effect of node k, called influence function of node k.

The sum of E_i is called one-dimension influence function of network

$$\sum_{i=1}^{n} E_i = \mathrm{IF}_{\mathrm{net}}, \quad i \in R. \tag{5}$$

If k_1, k_2 were removed, $E_{1,2}$ were called influence function of nodes k_1, k_2. If the removed nodes were sets, for example, the removed nodes sets are $\wp K = \{K_1, K_2, \ldots\}$, $K_1 = \{k_1, k_2\}$, $K_2 = \{k_3, k_4\}$, then the $\wp E = \{E_{K_1}, E_{K_2}, \ldots\}$ is the influence function set of network.

Definition 3 (influence function set of weighted algebraic connectivity). Propose the density (weighted) stands for more easy to produce congest, and to be more unreliability. Here density weighted influence function set of network is defined. It has a relationship with throughput, energy cost, and so on.

Lemma 4. *An ideal random network means that there are no determining factors that can infect network communication, for example, (nodes) density, (effective) communication distances. Then the mean or least square mean of connectivity is optimum estimation value, and meanwhile the network may be microstable (less intermoving) or reliable.*

Note. In ZigBee, there are phenomena of near neighbor ring, hidden nodes, and so on. In WirelessHart, there are phenomena of limited connective number, loss of noisy node, and so on. They have not been studied in detail in the paper.

3.3. Changeable Algebraic Connectivity. There are some different kinds of model of time-varying topologies: (1) the switching topology which refers to a deterministically time-varying model where, at each time instant, the network adopts a topology from a known set; (2) random topologies, in the presence of random failures caused by working under uncertain interference, for instance, changes in the environment, mobility of the nodes, asynchronous sleeping periods, or randomized communication protocols; the topology of a WSN varies randomly with time; (3) E-R model.

When wireless sensor networks are working under uncertain interference, nodes and links may lose effect momentarily or permanently. And the topology may be different with random topologies.

When the links are added or removed unpredictably from the set at any time, the graph can be looked to as the realization of a random process. WSN are normally exposed to random communication failures caused by uncertain interference, and these communication impairments cause abrupt changes in the connectivity of the network, which are described by means of a random graph,

$$\begin{aligned} A_k &= A_{k-1} \cdot B_t, \\ B_t &= I, \quad t < t_c, \\ B_t &= I + \Delta B, \quad t > t_c. \end{aligned} \tag{6}$$

When considering formula (6), the connectivity matrix A is changed after time t_c. And meanwhile the algebraic connectivity of graph is changed.

Then the algebraic connectivity λ_{N-1} is random variable; its distribution with the losing effective nodes is important to improve the robustness of complex networks, by optimization of complex networks topology or by repairing the damage nodes of complex networks.

4. Simulation and Test Results

The state of the network can be simplified as linear system:

$$\dot{x}(t) = -\mathbf{L}x(t), \tag{7}$$

where $x(t) = [x_1(t), x_2(t), \ldots, x_N(t)]^T$ is the vector of all states at time t and $\mathbf{L} \in \Re^{N \times N}$ is the Laplacian matrix associated with the graph.

A discrete implementation of the expression of (7) is

$$x_i(k+1) = \sum_{j \in N_i \cup \{i\}} w_{ij} x_j(k), \quad k \geq 0. \tag{8}$$

Different from weighted algebraic connectivity, w_{ij} is nonzero weight assigned by node i to the information received from node j, satisfying

$$\sum_{j \in N_i \cup \{i\}} w_{ij} = 1. \tag{9}$$

More generally, the linear control systems are described by the following state equation:

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{10}$$

where $x(t) = \{x_1(t), x_2(t), \ldots x_N(t)\}^T$, which is the state of a system of N nodes at time t. A is the $N \times N$ adjacency matrix of the network representing the system. B is the $N \times M$ input matrix, which identifies the nodes where the input signals are imposed. The input signal vector $u(t) = \{u_1(t), u_2(t), \ldots u_N(t)\}^T$ is a time-dependent input signal vector. The state of each node at each time step is controlled by the linear combination of the elements of the input vector.

4.1. Weighted Algebraic Connectivity and Not Weighted Algebraic Connectivity. In wireless condition monitoring network, supposing whether nodes can be connected or not only relies on its effective communication distance. As shown in Figure 4, the density of nodes and the topology of networks are two important factors to research communication links.

(1) The Relationship between Density and Connectivity. Simulate method: in certain area, using different numbers of nodes circulate its connectivity.

The connectivity is defined as the successful connection possibility when random deploy nodes in this area 100 times. When numbers of nodes are above than or equal to 10, the networks have a reliability of connectivity (as shown in Figure 5). So the theory of effective distance is less than 0.47.

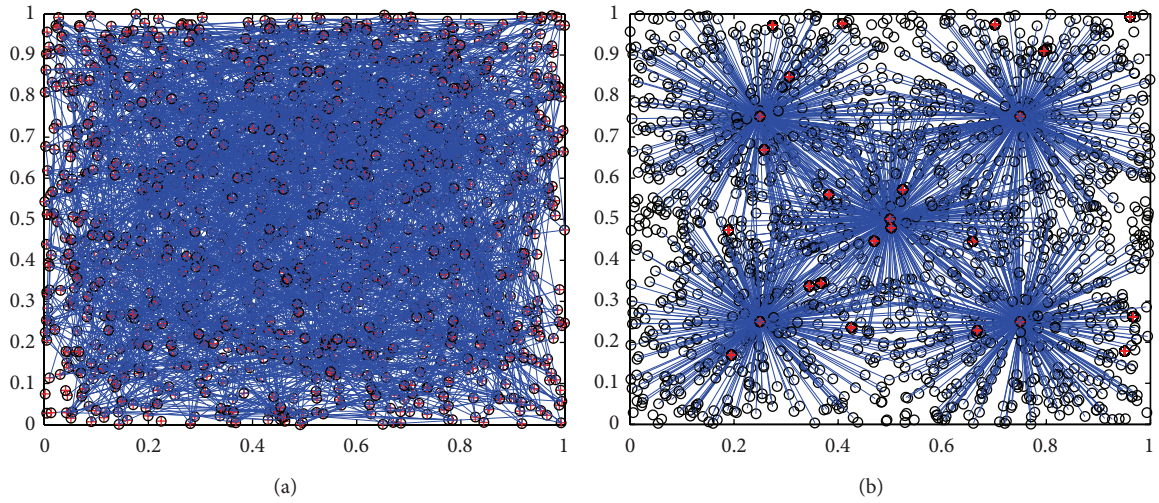


Figure 4: Different topology and its communication links in a certain area (left is random mesh; right is star).

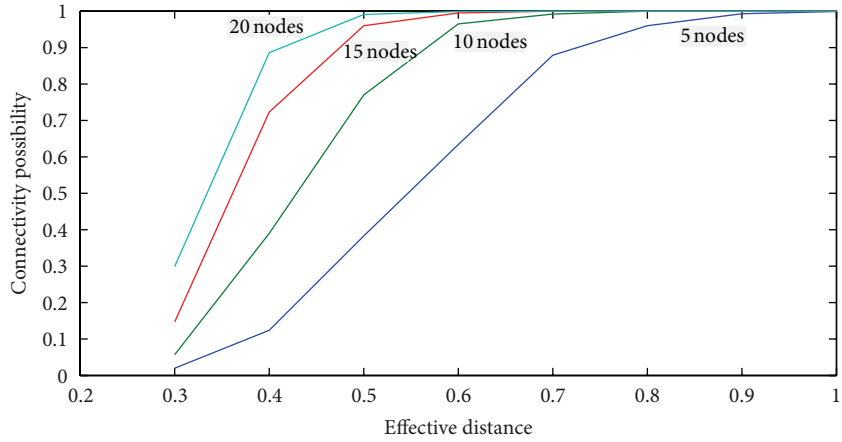


Figure 5: The relationship between density and connectivity.

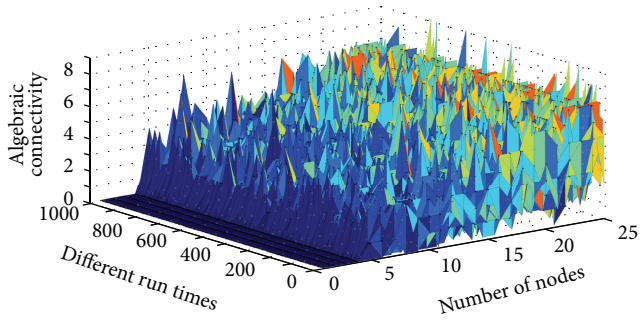


Figure 6: The relationship between algebraic connectivity and connectivity.

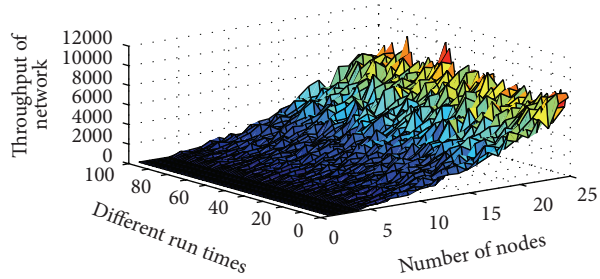


Figure 7: The relationship between throughput of network and density.

This means, in wireless condition monitoring network, an effective distance of random network should be larger than half of monitoring area. Considering its physical communication ability, the edge of monitoring cell is defined as twice times of its effective distance.

(2) The Relationship between Algebraic Connectivity and Density When the Effective Distance Is about 0.5. As shown in Figure 6, discrete algebraic connectivity is varied with the density (or number of nodes in certain area); the useful value of algebraic connectivity is a flat plain of certain density (10–20 nodes in a cell). The reason is too large the density

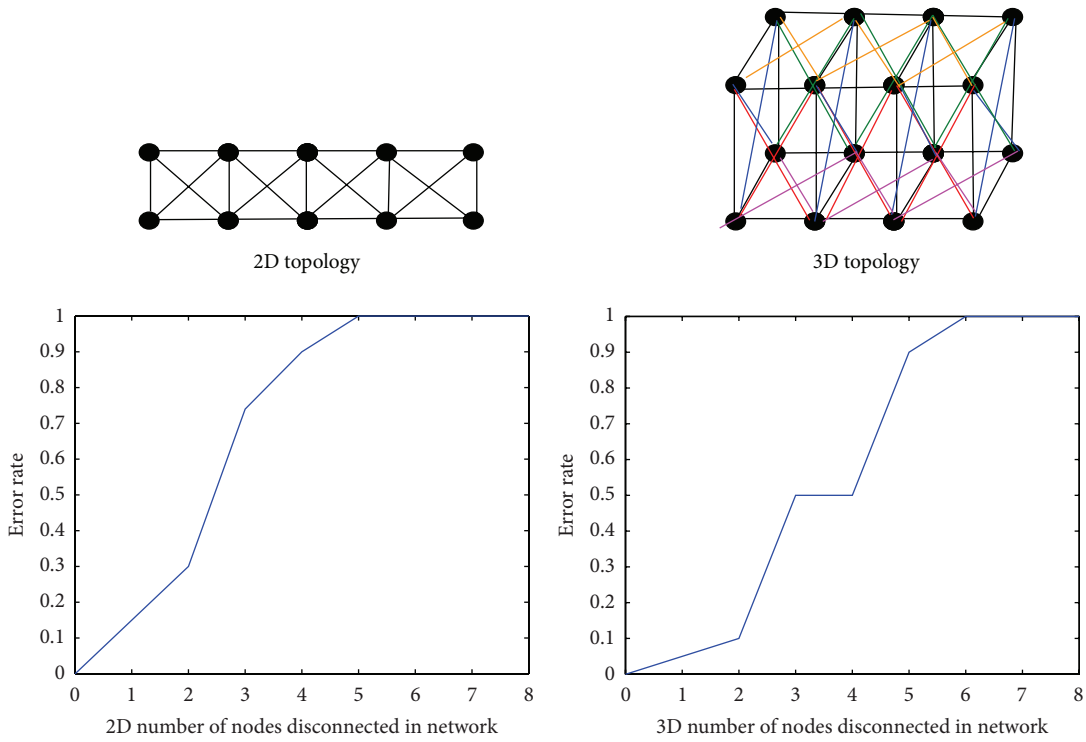


Figure 8: The relationship between random destruction number of node and connectivity.

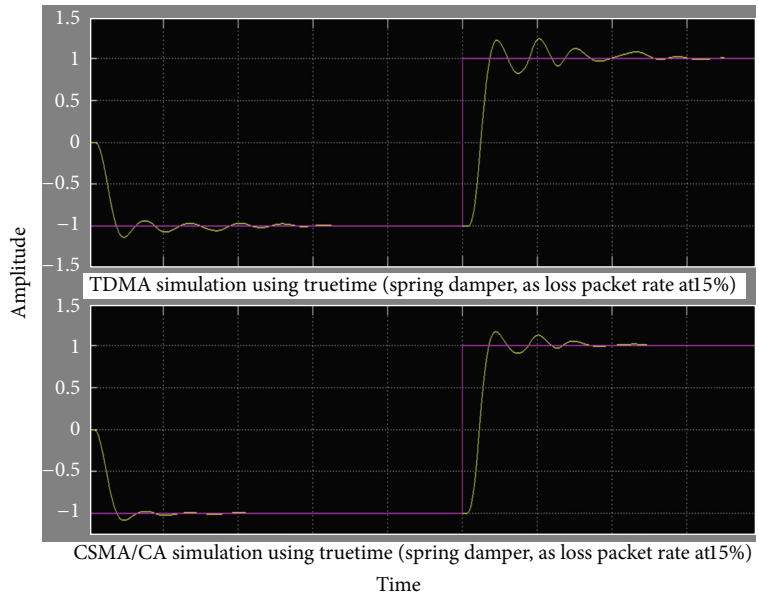


Figure 9: The relationship between loss packet rate and step response of spring damper.

form another different subgraphs and outlier of algebraic connectivity.

(3) The Relationship between Network Throughput and Density. In simulation, throughput is only comprised of the transmitting sense data; the network management steam is not considered. As shown in Figure 7, when density of network is large, the throughput of network became very large, such that the value is almost 10 k byte per second. So the density of network should not be large.

As discussed in (2) and (3), this gives the property of density weighted algebraic connectivity.

4.2. Robustness Analysis of Random Complex Network

(1) The Relationship between Random Destruction Number of Node and Connectivity. As shown in Figure 8, when random destruction of node is larger than its degree, then the connection of network can be destroyed.

(2) The Robustness Analysis of Random Complex Network. Consider a new graph that includes the plant's sensors and actuator, where the plant is controlled using a multihop wireless network. In applied layer application, condition monitoring parameters are measured by sensors and transmitted in networks; the networks construction changed with uncertain interference; this output is different with ordinary output with its random delay, packets loss rate, and so on.

There are two kinds of data; one kind is distributed data. It combined lots of nodes to get a useful data, as shown in formula

$$x[k+1] = (A + \Delta A)\, x[k] + (B + \Delta B)\, u[k], \qquad y[k] = (C + \Delta C)\, x[k]. \tag{11}$$

The another kind is standalone data, when can be described as a function, like

$$y_i(t) = f(x_i(t)) + \Delta f(x_i(t)). \tag{12}$$

An example delay of wireless HART network is

$$T\_d = -0.0011 \times n^2 + 0.24 \times n - 1.3333, \quad n > 50, \qquad T\_d = 0.1029 \times n + 2, \quad n < 50. \tag{13}$$

Suppose that loss of effective nodes is relay nodes, and then delay time value is changed in normality distribution way; its means is zero. Its standard deviation has a relationship with connectivity.

As a result, if delay time is shorter than sample time, delays will do less interference with LTI system. And if the packet loss rate was less than 15% (use true time simulate using CSMA/CA, TDMA), spring damper system should work normally with little performance degrade (as shown in Figure 9).

If the system is not connected, should repair it. This robust control problem is to find control scheme to obtain system robustness. That includes two factors: the first is topology selection; the second is repairing scheme.

Assumption 5. If the connected degree distribution was Poisson distribution, the Packet loss rate under uncertain interface should follow the Poisson distribution.

Then this becomes a time-space transmitting processing problem.

In every time slot different density (data transmitting rate), the most possibility of losing packet is shown in Figures 10 and 11.

Conclusion. It is obvious that the cluster-tree, star-mesh topologies are easier than only mesh network to repair the network for its small algebraic connectivity and communication links. And it uses self-repairing or deploying new nodes to repair network.

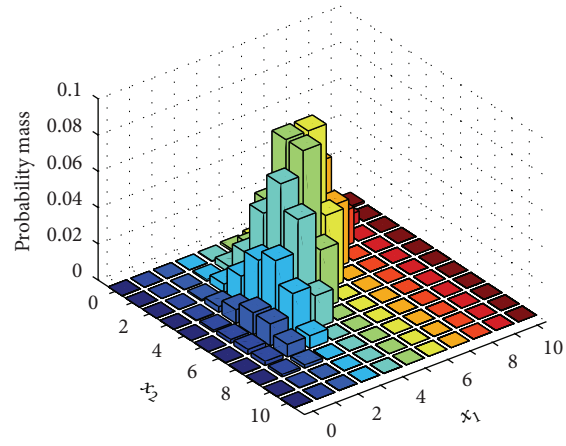


Figure 10: The time and space domain losing packet rate at lower space density.

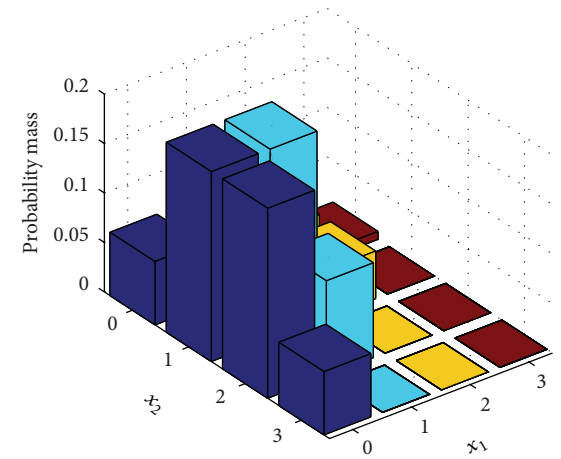


Figure 11: The time and space domain losing packet rate at higher space density.

But center nodes are more important and fragile than other nodes in these topologies. If they are to be novel stronger than other nodes, in median and large networks, this topology may have large possibility to be more robust than mesh topology in median and large scale networks. And, in small scale network, mesh topology may have large possibility to be more robust than other topology.

5. Conclusions

The cluster-tree, star-mesh topologies are easier than only mesh network to repair the network and may have large possibility to be more robust than mesh topology in median and large scale networks. In small scale network, mesh topology may have large possibility to be more robust than other topology.

If random destruction of node was larger than its algebraic connectivity, the connection of network should be broken out.

The properties of density weighted algebraic connectivity are two factors: one is that the useful value of algebraic connectivity is in certain density; the other is an effective network in which the density of network should not be large.

Acknowledgments

For support and contributions to discuss the authors would like to thank the rest of MCC (Mobile Computer Center) of UESTC (University of Electronic Science and Technology of China), Canada partner of ISTP, and others.

References

[1] R. Albert and A. L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, pp. 47–97, 2002.

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

[3] M. Boguñá and R. Pastor-Satorras, "Class of correlated random networks with hidden variables," *Physical Review E*, vol. 68, no. 3, Article ID 036112, 2003.

[4] R. Albert, H. Jeong, and A.-L. Barabási, "Error and attack tolerance of complex networks," *Nature*, vol. 406, no. 6794, pp. 378–382, 2000.

[5] R. Cohen, S. Havlin, and D. Ben-Avraham, "Efficient immunization strategies for computer networks and populations," *Physical Review Letters*, vol. 91, no. 24, Article ID 247901, 2003.

[6] S. Havlin and R. Cohen, *Complex Networks: Structure, Robustness and Function*, Cambridge University Press, 2010.

[7] H. J. Herrmann, C. M. Schneider, A. A. Moreira, J. S. Andrade Jr., and S. Havlin, "Onion-like network topology enhances robustness against malicious attacks," *Journal of Statistical Mechanics*, vol. 2011, no. 1, Article ID P01027, 2011.

[8] A. A. Moreira, J. S. Andrade Jr., H. J. Herrmann, and J. O. Indekeu, "How to make a fragile network robust and vice versa," *Physical Review Letters*, vol. 102, no. 1, Article ID 018701, 2009.

[9] M. E. J. Newman, "The structure and function of complex networks," *SIAM Review*, vol. 45, no. 2, pp. 167–256, 2003.

[10] C. M. Schneider, A. A. Moreira, J. S. Andrade Jr., S. Havlin, and H. J. Herrmann, "Mitigation of malicious attacks on networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 108, no. 10, pp. 3838–3841, 2011.

[11] B. Hu, F. Li, and H.-S. Zhou, "Robustness of complex networks under attack and repair," *Chinese Physics Letters*, vol. 26, no. 12, Article ID 128901, 2009.

[12] G. Paul, T. Tanizawa, S. Havlin, and H. E. Stanley, "Optimization of robustness of complex networks," *European Physical Journal B*, vol. 38, no. 2, pp. 187–191, 2004.

[13] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, Pa, USA, 2000.

[14] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Mathematical Journal*, vol. 23, pp. 298–305, 1973.

[15] J. G. Restrepo, E. Ott, and B. R. Hunt, "Approximating the largest eigenvalue of network adjacency matrices," *Physical Review E*, vol. 76, no. 5, Article ID 056119, 2007.

[16] A. Jamakovic and P. van Mieghem, "On the robustness of complex networks by using the algebraic connectivity," Technical Report, 2007.

[17] P. van Mieghem, D. Stevanović, F. Kuipers et al., "Decreasing the spectral radius of a graph by link removals," *Physical Review E*, vol. 84, no. 1, Article ID 016101, 2011.

[18] A. Jamakovic and P. van Mieghem, "On the robustness of complex networks by using the algebraic connectivity," in *Proceedings of the 7th International IFIP-TC6 Networking Conference (NETWORKING '08)*, vol. 4982 of *Lecture Notes in Computer Science*, pp. 183–194, 2008.

[19] C. L. Pua and W. J. Pei A, "Michaelson. Robustness analysis of network controllability," *Physica A*, vol. 391, no. 18, pp. 4420–4425, 2012.

[20] A. Barrat, M. Barthelemy, and A. Vespignani, *Dynamical Processes on Complex Networks*, Cambridge University Press, Cambridge, UK, 2009.

[21] F. Sorrentino, M. di Bernardo, F. Garofalo, and G. Chen, "Controllability of complex networks via pinning," *Physical Review E*, vol. 75, no. 4, Article ID 046103, 2007.

[22] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *Nature*, vol. 473, no. 7346, pp. 167–173, 2011.

[23] T. Nepusz and T. Vicsek, "Controlling edge dynamics in complex networks," *Nature Physics*, vol. 8, pp. 568–573, 2012.

[24] C. T. Lin, "Structural controllability," *IEEE Transactions on Automatic Control*, vol. 19, no. 3, pp. 201–208, 1974.

[25] N. J. Cowan, E. Chastain, D. A. Vilhena, and C. T. Bergstrom, "Controllability of real networks," In press. http://arxiv.org/abs/1107.2177.

[26] J. Menche, A. Valleriani, and R. Lipowsky, "Asymptotic properties of degree-correlated scale-free networks," *Physical Review E*, vol. 81, no. 4, Article ID 046103, 2010.

[27] M. Pósfai, Y. Y. Liu, J. J. Slotine, and A. L. Barabási, "Effect of correlations on network controllability," *Scientific Reports*, vol. 3, Article ID 1067, 2013.