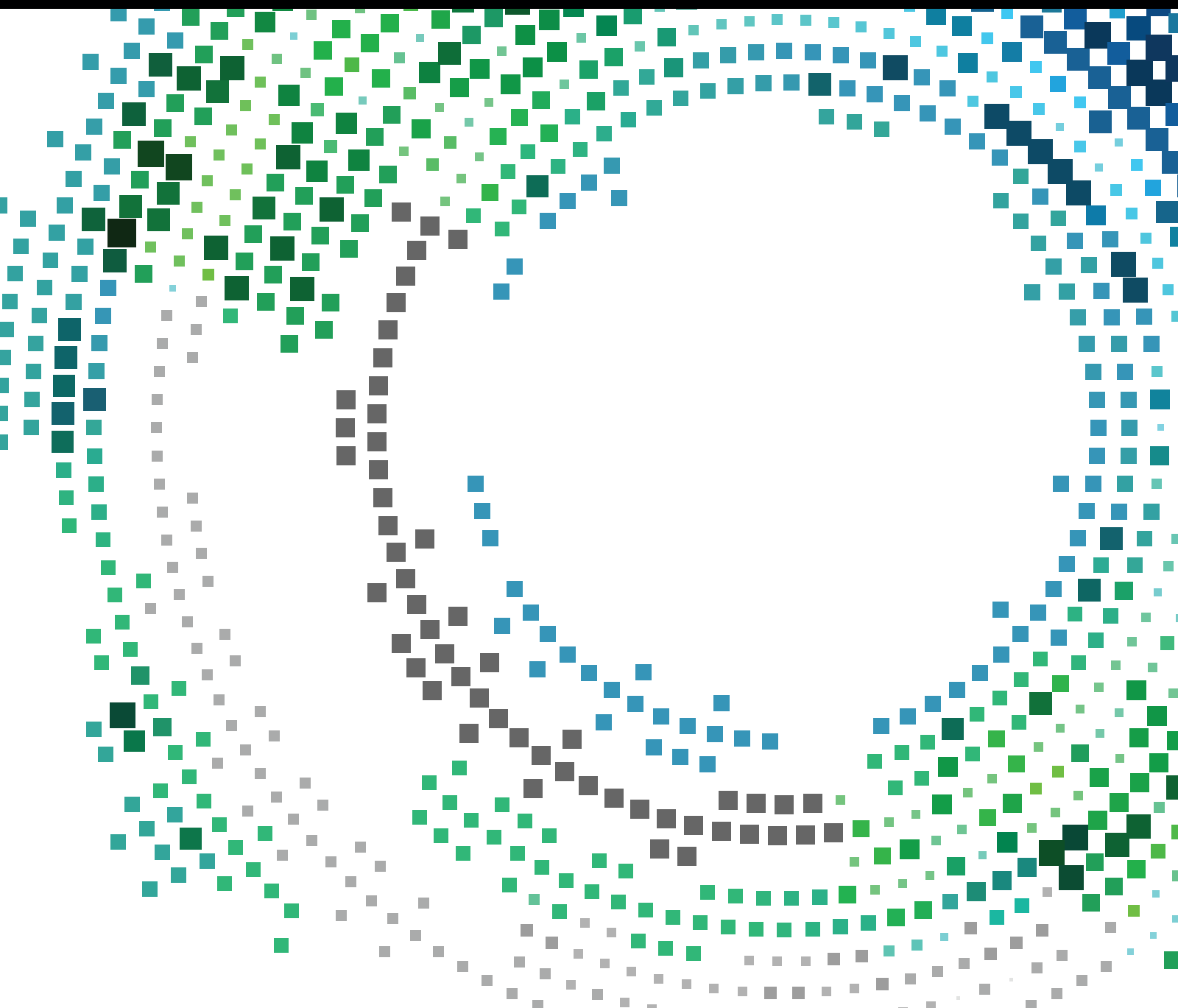


# Challenges for the Future Mobile Communication Systems

Lead Guest Editor: Yujin Lim

Guest Editors: Hideyuki Takahashi, Fan Wu, and Rossana M. C. Andrade





---

# **Challenges for the Future Mobile Communication Systems**


Mobile Information Systems

---

## **Challenges for the Future Mobile Communication Systems**

Lead Guest Editor: Yujin Lim

Guest Editors: Hideyuki Takahashi, Fan Wu,  
and Rossana M. C. Andrade



---

Copyright © 2017 Hindawi. All rights reserved.

This is a special issue published in “Mobile Information Systems.” All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

---

## Editorial Board

M. Anastassopoulos, UK  
C. Agostino Ardagna, Italy  
J. M. Barcelo-Ordinas, Spain  
Alessandro Bazzi, Italy  
Paolo Bellavista, Italy  
Carlos T. Calafate, Spain  
María Calderon, Spain  
Juan C. Cano, Spain  
Salvatore Carta, Italy  
Yuh-Shyan Chen, Taiwan  
Massimo Condoluci, UK  
A. de la Oliva, Spain  
Jesus Fontecha, Spain

Jorge Garcia Duque, Spain  
L. J. García Villalba, Spain  
Michele Garetto, Italy  
Romeo Giuliano, Italy  
Javier Gozalvez, Spain  
Francesco Gringoli, Italy  
Peter Jung, Germany  
Dik Lun Lee, Hong Kong  
Sergio Mascetti, Italy  
Elio Masciari, Italy  
Maristella Matera, Italy  
Franco Mazzenga, Italy  
Eduardo Mena, Spain

Massimo Merro, Italy  
Jose F. Monserrat, Spain  
Francesco Palmieri, Italy  
J. J. Pazos-Arias, Spain  
Vicent Pla, Spain  
Daniele Riboni, Italy  
Pedro M. Ruiz, Spain  
Michele Ruta, Italy  
Stefania Sardellitti, Italy  
Floriano Scioscia, Italy  
Laurence T. Yang, Canada  
J. Zhang, Australia

# Contents

---

## **Challenges for the Future Mobile Communication Systems**

Yujin Lim, Hideyuki Takahashi, Fan Wu, and Rossana M. de Castro Andrade  
Volume 2017, Article ID 1298659, 1 page

## **A Localization Based Cooperative Routing Protocol for Underwater Wireless Sensor Networks**

Nadeem Javaid, Hammad Maqsood, Abdul Wadood, Iftikhar Azim Niaz, Ahmad Almogren, Atif Alamri, and Manzoor Ilahi  
Volume 2017, Article ID 7954175, 16 pages

## **Weighted-DESYNC and Its Application to End-to-End Throughput Fairness in Wireless Multihop Network**

Ui-Seong Yu, Ji-Young Jung, Eutteum Kong, HyungSeok Choi, and Jung-Ryun Lee  
Volume 2017, Article ID 2504604, 10 pages

## **A Collusion-Resistant and Privacy-Preserving Data Aggregation Protocol in Crowdsensing System**

Chang Xu, Xiaodong Shen, Liehuang Zhu, and Yan Zhang  
Volume 2017, Article ID 3715253, 11 pages

## **Performance Analysis of User Pairing Algorithm in Full-Duplex Cellular Networks**

Wonjong Noh, Wonjae Shin, and Hyun-Ho Choi  
Volume 2017, Article ID 8182150, 12 pages

## **Network Intelligence Based on Network State Information for Connected Vehicles Utilizing Fog Computing**

Seongjin Park and Younghwan Yoo  
Volume 2017, Article ID 7479267, 9 pages

## **Distributed Association Method Assisted by Cell for Efficiency Enhancement of Wireless Networks**

Jaesung Park  
Volume 2017, Article ID 9701267, 7 pages

## **Principles, Applications, and Challenges of Synchronization in Nature for Future Mobile Communication Systems**

Hyun-Ho Choi and Jung-Ryun Lee  
Volume 2017, Article ID 8932631, 13 pages

## **Delay Tolerance in Underwater Wireless Communications: A Routing Perspective**

Safdar Hussain Bouk, Syed Hassan Ahmed, and Dongkyun Kim  
Volume 2016, Article ID 6574697, 9 pages

## Editorial

# Challenges for the Future Mobile Communication Systems

**Yujin Lim,<sup>1</sup> Hideyuki Takahashi,<sup>2</sup> Fan Wu,<sup>3</sup> and Rossana M. de Castro Andrade<sup>4</sup>**

<sup>1</sup>Department of Information Technology Engineering, Sookmyung Women's University, Seoul 04310, Republic of Korea

<sup>2</sup>Research Institute of Electrical Communication, Tohoku University, Sendai 980-8577, Japan

<sup>3</sup>Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

<sup>4</sup>Department of Computer Science, Federal University of Ceará, 2853 Benfica, Fortaleza, CE, Brazil

Correspondence should be addressed to Yujin Lim; [yujin91@sookmyung.ac.kr](mailto:yujin91@sookmyung.ac.kr)

Received 18 April 2017; Accepted 18 April 2017; Published 30 May 2017

Copyright © 2017 Yujin Lim et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As the wireless data traffic has been increased explosively, the future mobile communication system that can accommodate the traffic demand in a cost-effective and optimal manner attracts more and more attention in both academia and industry. Many standards bodies, countries, and major companies have proposed visions for the future mobile communication system and there have been a lot of proposals from the architectural changes to specific algorithms. For example, ITU specified the key performance requirements of 5G networks and 3GPP presented the roadmap for 5G standard deployment. NGNM presented use cases of 5G applications.

Future mobile communication systems will need to interconnect heterogeneous mobile communication systems and technologies and get them to work together so as to optimize services requirements and operational conditions. For example, software defined network is supposed to be deployed for core and access networks, machine-type communications will prevail, and self-organized mechanisms are introduced to increase network performance and reduce operational costs under highly variable and unpredictable network conditions. Even though a significant number of studies taking evolutionary approaches and revolutionary approaches are in progress to give shape to the vision of the future mobile communication system, more efforts are still required to crystallize the future mobile communication system.

The scope of this special issue is in line with recent contributions from academia and industry on the recent activities that tackle the technical challenges to concretize the future mobile communication system. For the current issue,

we are pleased to introduce a collection of papers covering a range of topics as follows:

- (i) Technologies for connected vehicle systems
- (ii) Technologies for cellular networks
- (iii) Technologies for wireless multihop networks
- (iv) Technologies for underwater wireless networks
- (v) Technologies for future mobile communication systems

## Acknowledgments

As always, we appreciate the high quality submissions from authors and the support of the community of reviewers.

*Yujin Lim*  
*Hideyuki Takahashi*  
*Fan Wu*  
*Rossana M. de Castro Andrade*

## Research Article

# A Localization Based Cooperative Routing Protocol for Underwater Wireless Sensor Networks

Nadeem Javaid,<sup>1</sup> Hammad Maqsood,<sup>1</sup> Abdul Wadood,<sup>2</sup> Iftikhar Azim Niaz,<sup>1</sup>  
Ahmad Almogren,<sup>2</sup> Atif Alamri,<sup>2</sup> and Manzoor Ilahi<sup>1</sup>

<sup>1</sup>COMSATS Institute of Information Technology, Islamabad 44000, Pakistan

<sup>2</sup>Research Chair of Pervasive and Mobile Computing, College of Computer and Information Sciences, King Saud University, Riyadh 11633, Saudi Arabia

Correspondence should be addressed to Nadeem Javaid; nadeemjavaidqau@gmail.com

Received 4 December 2016; Revised 14 March 2017; Accepted 27 March 2017; Published 30 May 2017

Academic Editor: Yujin Lim

Copyright © 2017 Nadeem Javaid et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Localization is one of the major aspects in underwater wireless sensor networks (UWSNs). Therefore, it is important to know the accurate position of the sensor node in large scale applications like disaster prevention, tactical surveillance, and monitoring. Due to the inefficiency of the global positioning system (GPS) in UWSN, it is very difficult to localize a node in underwater environment compared to terrestrial networks. To minimize the localization error and enhance the localization coverage of the network, two routing protocols are proposed; the first one is mobile autonomous underwater vehicle (MobiL-AUV) and the second one is cooperative MobiL (CO-MobiL). In MobiL-AUV, AUVs are deployed and equipped with GPS and act as reference nodes. These reference nodes are used to localize all the nonlocalized ordinary sensor nodes in order to reduce the localization error and maximize the network coverage. CO-MobiL is presented in order to improve the network throughput by using the maximal ratio combining (MRC) as diversity technique which combines both signals, received from the source and received from the relay at the destination. It uses amplify-and-forward (AF) mechanism to improve the signal between the source and the destination. To support our claims, extensive simulations are performed.

## 1. Introduction

In recent years, UWSNs are getting more attention due to useful applications such as military surveillance, oil exploration, and seismic and environmental monitoring. The UWSN consists of sensor nodes which are deployed inside the water. The main task of these sensor nodes is to sense the underwater environment and transmit the sensed data to the onshore sink by using acoustic signals. At the same time, localization is one of the major issues in UWSNs [1], especially in the design of the routing protocol. Due to harsh underwater environments, data must be gathered with the location information. Therefore, a sensor node must know its location at the time of deployment to recover the data from the exact locations. Although in manual deployment sensor nodes know their exact location, however, due to dynamic nature of underwater, manual deployment is not feasible. In

this case, sensor nodes are randomly deployed in order to collect the information from the areas which are difficult to reach. Localization is very important in many applications like a battlefield area, catastrophic areas, and so on. Due to water currents, localization is a challenging task in UWSNs. Generally, acoustic waves are used in underwater communication; these waves have high propagation delay and extremely low bandwidth which lead to high probability of localization error. Unlike terrestrial WSNs, the underwater network does not support GPS for position estimation. Thus, efficient localization schemes are needed to estimate the position(s) of the node(s).

Nowadays, cooperative communication has gained much attention in UWSNs due to its unique ability of joining erroneous packets at the destination by using cooperative diversity techniques. While minimizing the energy consumption with fewer retransmissions, a cooperative process involves



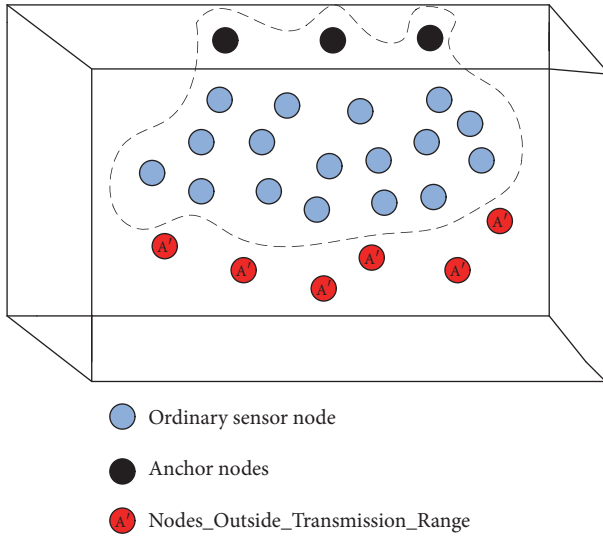


FIGURE 1: Anchor nodes coverage problem.

the neighboring nodes to relay the data of a particular source node to the destination. The main techniques used to relay data from the source node to the destination node are AF and decode-and-forward (DF). In the AF relaying technique, the relay node amplifies the received signal and then transmits the amplified version of the signal towards the destination, whereas in the DF scheme the relay node decodes the received signal and then transmits the signal to the intended destination.

After the localization, the next objective is data routing. For this purpose various attempts in the literature have been proposed. In MobiL [2], anchor nodes deployed on water surface are used as reference nodes for the localization of ordinary nodes. However, nodes that are not in the transmission range of these anchor nodes are localized as shown in Figure 1. The nodes with label  $A'$  are not in the transmission range of anchor nodes which results in higher chance of the localization error. The anchor nodes do not provide enough coverage to localize the nodes. So, the remaining nonlocalized nodes will make reference to the other ordinary nodes resulting in a high localization error.

To encounter these problems, two localization routing schemes are proposed in this paper: (i) MobiL-AUV [3] and CO-MobiL. In MobiL-AUV, three mobile AUVs are deployed underwater at predefined depth ( $d$ ). The mobile AUVs accelerate towards the surface to find their own three-dimensional coordinates via a GPS satellite and dive back to the underwater as shown in Figure 2. The mobile AUVs cover a large network volume and act as reference nodes for nonlocalized nodes, minimizing the localization error due to minimum distance between mobile AUVs and nodes.

We also present a region based cooperative routing mechanism as shown in Figure 3. The nodes closer to the onshore sink (nodes in transmission range of sink) are considered in region one, and the rest of the nodes are considered in region 2. Nodes in region 1 directly communicate with the sink consuming less energy. On the other hand, nodes in region 2 communicate via a cooperatively chosen best relay.

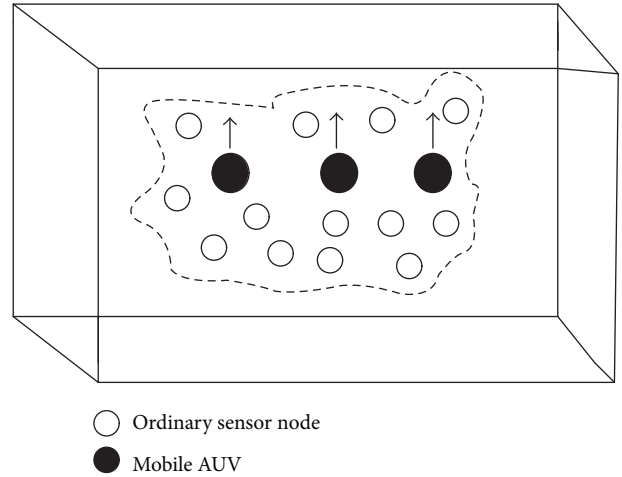


FIGURE 2: AUVs coverage area.

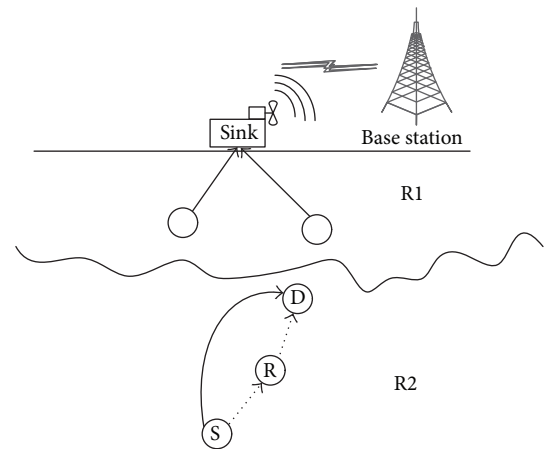


FIGURE 3: Region base cooperation.

The remainder of the paper is organized as follows. The related existing work is summarized in Section 2. Section 3 discusses our scheme in a detail. Then, simulation results are presented and discussed in Section 4 and finally Section 5 concludes our work by discussing the findings of our schemes.

## 2. Related Work

In this section, we discuss recent routing protocols for UWSNs in two categories: (i) nodes mobility and localization based and (ii) cooperation based.

**2.1. Node Mobility and Localization.** In [4], the authors propose a distributed localization technique for UWSNs. The authors introduce two types of nodes: anchor nodes and ordinary nodes. The anchor nodes are used as reference nodes which are localized, whereas the ordinary nodes do not know their location. The main focus is on the localization of ordinary nodes by using distributed localization techniques.

The authors in [5] propose a centralized localization technique in which each sensor node sends its location information to the onshore sink. The scheme results in a higher energy utilization as well as a high localization error. The centralized localization schemes are not suitable for a high level UWSN task.

Performance analysis of ranged base and distributed localization techniques and protocols are presented in [6]. The authors conclude that the existing protocols that are made for small scale networks cannot be used in large scale networks.

Received signal strength (RSS) and ad hoc on-demand distance vector protocol to detect the attacks on the localization mechanism is proposed in [7]. The authors choose the best and secured neighbors for data forwarding process, such that data is transmitted through the qualified secured neighbors.

The authors in [8] propose correction received signal strength index (RSSI) localization scheme for UWSNs. The reference nodes send beacon message; on receiving a message, the nodes use median mean filtering to process the information and by the help of weighted centroid technique, the unknown node finds its three-dimensional coordinates.

Optimization and minimum cost factor is briefly explained in [9]. The authors focus on localizing the non-localized nodes by utilizing lower number of anchored nodes. To efficiently use the anchor nodes, the authors propose the greedy mechanism to select the anchor nodes pair. This scheme also introduces the confidence interval methodology to handle the localization error issues.

In [10], multihop localization process is proposed. This scheme has two phases: in the first phase, the nodes find the shortest link between anchored nodes and ordinary nodes by using a greedy algorithm. In the second phase, the non-localized nodes find their positions by using trilateration scheme.

The time difference of arrival (TDOA) based localization technique for a heterogeneous network is proposed in [11]. In this scheme, there are three types of nodes: the target nodes, the anchored nodes, and the reference nodes. In the first step, the TDOA of minimum four nodes sets is calculated and in the second step the values with position information of localized nodes are used to find the position of the target nodes.

The authors in [12] propose a scalable localization technique. The protocol has two phases: anchored node localization and nonlocalized node localization. On the basis of node's past mobility, the nodes predict their future position. During anchor nodes localization, the anchor nodes use the lateration technique to find their respective coordinates.

In [13], the authors present three different deployment strategies and briefly explain their effects on localization. The authors conclude that tetrahedron deployment strategy shows good results as compared to other deployment strategies in terms of the localization error.

An analysis of localization process is proposed in [14]. The authors make use of asynchronous clock to make anchor nodes asynchronous. The asynchronous anchor nodes help

other nodes to find their position in an effective way with less localization error.

Localization mechanism for nonlocalized nodes is discussed in [15]. The protocol consists of two parts: in the first part, sensor nodes estimate their distance from the anchored reference node and in the second part sensor nodes utilize their estimated distance to find their coordinates with a trilateration mathematical technique.

The authors [16] propose a hybrid localization scheme for ocean sensor networks. This localization scheme is divided into three steps; in the first step, the group nodes are localized to act as anchor nodes, in the second step with the help of anchor nodes, the nonlocalized nodes find their coordinates to become localized, and in the final step, the free drifting nodes find their location by applying floating node localization algorithm (FLAP).

In [26], a network access mechanism is presented in three phases: (i) network discovery, (ii) finding of a relay path, and (iii) association for multihop underwater acoustic local area network. It achieved the network lifetime, high packet delivery ratio with the balanced energy consumption.

Wu et al. [27] propose a network coding based routing protocol for UWSNs. In this scheme, the authors present three algorithms: (i) time-slot based routing (TSR) to reduce the sensor nodes conflict due to underwater dynamic nature, (ii) time-slot based balanced routing (TSBR) in order to minimize the energy consumption, and (iii) time-slot based balanced network coding algorithm to balance the energy consumption over long routing paths.

An energy efficient depth based routing protocol is proposed in [28]. It considers a low depth sensor node that has high residual energy to transmit a data packet to maximize the network lifetime. It performs better in dense deployment; however in sparse conditions, it consumes more energy resulting in low network lifetime and lower packet delivery ratio.

Authors in [29] propose an energy efficient routing protocol based on a physical distance and a residual energy ( $ERP^2R$ ) for UWSNs.  $ERP^2R$  uses physical addresses of nodes to minimize the location error. It also enhanced the network lifetime with balanced energy consumption.

An energy efficient routing algorithm for UWSN inspired by ultrasonic frog mechanism is presented in [30]. In order to improve the network lifetime, it utilizes the gravity function to indicate the attractiveness of network nodes for each other. This scheme achieves comparatively a high network lifetime and more packet acceptance ratio.

Cao et al. propose a balanced transmission algorithm for UWSNs [31]. Authors divide the transmission into two phases: (i) an energy efficient algorithm based on the optimal distance to optimize the energy consumption of the network and (ii) data load algorithm which balances the data load on the sensor nodes. This protocol achieves the energy efficiency by balancing the data load. The comparison of the discussed routing protocols (Node Mobility and Localization) is given in Table 1.

TABLE 1: State-of-the-art related work: Node Mobility and Localization.

Technique	Features	Achievements	Limitations
MobiL: a 3-dimensional localization scheme for mobile UWSNs [2]	Handles nodes mobility and nodes localization	Mobility model for mobile nodes, efficient node localization with less error	Large distance between ordinary nodes and reference nodes resulting in localization error
Design and implementation of a time synchronization-free distributed localization [4]	Event-driven time synchronization-free distributed localization scheme	Successful efficient nodes localization	ToA based distance estimation method is used, which causes error if delay occurs
A survey of architectures and localization techniques for UASNs [5]	Survey of UASNs and localization techniques	Briefly explaining the existing localization techniques and their drawbacks	Performance analysis of different localization methods with different mobility models is neglected
Performance evaluation of localization algorithms in large scale underwater sensor networks [6]	Comparison of ranged base and distributed localization schemes	Analysing different schemes and explaining briefly their advantages and disadvantages	Analysis of different localization methods is not performed
Localization using multilateration with RSS based random transmission directed localization [7]	RSS based efficient random transmission directed localization is introduced	RSS is used for distance estimation, multilateration technique is used for efficient nodes localization	RSS greatly is affected by channel conditions
A distance measurement wireless localization correction algorithm based on RSSI [8]	Node localization mechanism based on RSSI	Efficient correction of estimated RSSI values, efficient measuring of the coordinates of nodes, minimizing error during localization process	RSSI does not accurately localize nodes due to multipath propagation and fading effects
Minimum cost localization problem in three-dimensional ocean sensor networks [9]	Minimum localization cost and handles distance problem	Finding best anchor set to measure each node coordinate efficiently	If confidence interval threshold increases the anchor nodes selection will be affected
A multihop localization algorithm in UWSNs [10]	Multihop localization algorithm to handle the void node localization problem	Efficient mechanism for nodes localization	Not suitable for large scale applications
TDOA based target localization in inhomogeneous UWSNs [11]	TDOA based localization scheme, introduced for inhomogeneous WSNs	Achieving improved localization coverage area	Affected by channel conditions, time synchronization is very important
Scalable localization with mobility prediction for UWSNs [12]	Localization scheme for large scale applications with mobility forecasting	Efficient node mobility prediction and improved localization	High energy consumption
Impacts of deployment strategies on localization performance in UASNs [13]	Efficient node deployment strategies and their effects on localization coverage area and error are introduced	Successfully achieving larger coverage area and less error because of efficient node deployment mechanism	Localization is not very accurate
On-demand asynchronous localization for UWSNs [14]	On-demand nodes localization scheme using asynchronous anchor nodes	Successfully localizing both types of nodes passive and active	Passive nodes localization accuracy is less than the active nodes
A three-dimensional localization algorithm for UASNs [15]	Iterative and distributed nodes localization algorithm	Improved coverage area with less error	When node mobility increases the localization error increases
Localization for drifting restricted floating ocean sensor networks [16]	Efficient nodes localization scheme	Large number of nodes finding their three-dimensional coordinates successfully	Higher network deployment cost

In the discussed existing routing protocols, the impact of nodes mobility is not taken into account, which results in high localization error and degrades the performance of the network. Therefore, to mitigate the localization error due to water currents, in our proposed work, the speed of the node is computed in order to estimate the coordinates of the sensor

node deployed in the network field. The details of procedure are given in Section 3.1.

*2.2. Cooperative Routing.* The authors in [17] propose an energy efficient cooperative routing protocol. It is a cross layer protocol for cooperative communication where if the

destination node fails to receive the data packet successfully then the nearby relay node retransmits the data packet to the destination node. This protocol achieves energy efficiency compared to its counterpart schemes.

A cross layer cooperative communication protocol is proposed in [18]. The authors utilize the power allocation and route selection information during the route selection phase; a link with minimum chances of collision is selected for the data transmission. This protocol achieves high network throughput; however, the mechanism of forwarder selection is not reliable.

An incremental qualified relay selection for the cooperative communication is presented in [19]. When the destination node fails to receive the data via direct transmission, the relay node was used to retransmit the same data packet. The authors computed closed form expressions for bit error rate (BER) and outage probability. It achieves less probability of error and high network throughput at the cost of high energy consumption.

In [20], the authors propose an efficient cooperative routing protocol. In this technique, the selection of the best relay depends on the channel conditions and distance between the source node and the neighbor node. The best relay is selected from the set of qualified neighboring nodes. The acknowledgement mechanism helps to reduce the redundant packets resulting in high reliability; however, the relay selection mechanism is not energy efficient.

Authors in [32] propose an energy efficient routing protocol for cooperative transmission. Two problems are tackled in this proposed scheme; the first one is the energy consumption and the other one is the power allocation for cooperative transmissions. This scheme mainly focuses on the distance between different nodes for efficient power allocation.

A cooperative routing protocol for UWSNs is proposed in [33]. There are three types of nodes: the source nodes, the relay nodes, and the destination nodes. When the data is transmitted from the source to the destination, if the destination node fails to receive error-free data then the relay node has to retransmit the data towards the destination. There are two scenarios; in the first scenario single relay node is selected whereas in the second scenario two relay nodes participate in the data retransmission.

In [21], a cooperative routing protocol is presented to minimize the collision in the network. Authors formulated the problem with the use of mixed integer nonlinear programming (MINLP). They used branch-and-bound algorithm to reduce the search space and save the energy for solving mixed integer nonlinear programming problems.

To achieve energy efficiency, the authors in [22] propose a scheme which allocates a power at each hop between the sender and the receiver/sink. Based on the allocated power, each sensor node computes its signal-to-noise ratio (SNR) of the link (between the source and the destination) and distance between the source and the destination. It performs better in terms of network lifetime and end-to-end delay.

In [23], the authors present a cooperative strategy for data transmission in a clustered network. They divided the scheme

hello packet transmission into two phases: (i) intercluster and (ii) intracluster. In intercluster, the cluster head (CH) broadcasts the data packet directly to the neighbor CHs and via potential relay nodes only if any CH is not in its transmission range. Similarly, in intracluster, the CH forwards its data packets to all sensor nodes within the cluster, the nodes which receive hello packet successfully, selected as potential relay nodes. However, it consumes more energy when the nodes density is high in a cluster.

Authors propose a cooperative routing algorithm in [24]. This algorithm considers a propagational delay and BER of the link to select the best incremental relay for data forwarding between the source and the destination. However, it achieves high performance of the network at the cost of more end-to-end delay.

To improve the network throughput, a cooperative scheme is presented [25]. This scheme uses partner nodes mechanism for cooperation and for the selection of each node, SNR of the link and propagational is checked; if the link met the criteria then a node is elected as a partner node to perform cooperation. However, this scheme enhanced the packet delivery ratio with higher stability period at the cost of high energy consumption. Summary of the cooperative routing schemes is presented in Table 2.

In this subsection, numerous cooperative routing algorithms are stated. The energy consumption is high due to duplicate packets at the destination in the discussed schemes, due to energy constraint of sensors; in this paper, we have proposed a CO-MobiL, in which relay node only transmits data packet towards the respective destination, when the received packet is erroneous or not received by the destination. The working mechanism of the proposed work is given in Section 3.2.

### 3. Proposed Work

In this paper, two contributions are presented: (i) an AUV based node localization scheme (MobiL-AUV) and (ii) a cooperative routing protocol (CO-MobiL). In the first, non-localized nodes are localized by using an efficient AUV aided localization mechanism. In the second scheme, after the successful localization of node regions based cooperative data routing is carried out. Details are as follows.

*3.1. MobiL-AUV.* We consider a three-dimensional underwater network, where  $n$  numbers of sensor nodes are randomly deployed. With the help of onboard pressure sensors, sensor nodes are capable of determining their depth. All nodes can communicate with mobile AUV as well as with other deployed nodes. We assume that  $N$  is the number of mobile AUVs that are deployed inside the water at depth  $d$ . These mobile AUVs periodically accelerate towards the water surface to get their own three-dimensional coordinates information by using the GPS [34, 35]. After getting the required information, these dive back into the water and broadcast localization beacon messages. These mobile AUVs act as reference nodes for all other underwater nodes.

TABLE 2: State-of-the-art related work: cooperative routing.

Technique	Features	Achievements	Limitations
Joint cooperative routing and power allocation for collision minimization in WSNs with multiple flows [17]	Cooperative routing protocol, optimal power allocation to sensor nodes	Collision probability is minimized	High computational cost
Energy efficient cooperative communication for data transmission in WSNs [18]	Reliable and efficient data communication cooperative routing protocol	High throughput and reliable data delivery	Mobility of sensor node is neglected; qualified forwarder node selection mechanism is not accurate
Performance analysis of cooperative diversity with incremental-best-relay technique over Rayleigh fading channels [19]	Incremental relaying cooperation is implemented	Less probability of error, higher throughput, and reliability	High energy consumption
Exploiting cooperative relay for reliable communications in UASNs [20]	Cooperative scheme that increases the throughput and reliability of the network	High network throughput	Nonefficient relay node selection mechanism
Optimal and near-optimal cooperative routing and power allocation for collision minimization in WSNs [21]	MINLP problems are solved with branch-and-bound algorithms	Minimizing the end-to-end delay by reducing the search space	Only efficient for predefined network topologies
Co-UWSN: cooperative energy efficient protocol for UWSNs [22]	Incremental relay is selected on the basis of distance and SNR of the link	Comparatively less energy consumption	ACK on each packet resulted in high end to delay
EECCC (energy efficient cooperative communication in clustered WSNs) [23]	Clustering and cooperative communication in the network	Comparatively less energy consumption with direct communication	High energy consumption in dense deployment due to redundant transmissions
Relay selection and optimization algorithm of power allocation based on channel delay for UWSNs [24]	Relay is selected on the basis of BER and propagational delay	Improving the network lifetime	High end-to-end delay
Cooperative partner nodes selection criteria for cooperative routing in UWSNs [25]	Partner node selection on the basis of depth threshold and propagational delay	Comparatively high packet acceptance ratio	High energy consumption

All the ordinary sensor nodes have a communication link with mobile AUVs and also with other sensor nodes deployed underwater. The mobile AUVs form a link with GPS satellite via radio waves and the use acoustic waves to establish a link with underwater nodes. There are two phases in the Mobil-AUV localization scheme.

- (1) Mobility forecasting
- (2) Localization process

(1) *Mobility Forecasting.* In mobile UWSN, water currents have a strong impact on the mobility of sensor nodes. For the sake of simplicity we only focus on  $x$  and  $y$  coordinates. We ignore the  $z$  coordinate because two distinct coordinates (points) always compute a straight line between the source and the destination in the given transmission range. In this case, a straight data path must be established between the sender and the receiver node to avoid the use of extra energy. Considering an underwater scenario, the node  $k$  at depth  $d$  experienced underwater current velocity  $v_k = (v_k^x(d), v_k^y(d), 0)$ , where  $v_k^x$  and  $v_k^y$  are the velocities of the node in  $x$  and  $y$  directions, respectively. Due to spatial correlation in underwater networks, mobile sensor nodes show a

group-like behavior [2]. So, by using the group movement property as shown in (1), we can find the velocity of the node.

$$v_k^x(d) = \sum_{m=1}^{n_{\text{neigh}}} R_{m,k} v_m^x(d)$$

$$v_k^y(d) = \sum_{m=1}^{n_{\text{neigh}}} R_{m,k} v_m^y(d)$$
(1)

In (1),  $n_{\text{neigh}}$  represents the number of neighbors of sensor node  $k$ . The Euclidean distance between node  $m$  and node  $k$  is presented as  $r_{mk}$  and  $R_{m,k} = (1/r_{m,k}) / \sum_{m=1}^{n_{\text{neigh}}} (1/r_{m,k})$  represent the interpolation coefficient.

After the velocity estimation, each ordinary node in the network interchanges the speed beacon message periodically with other nodes. These special beacon messages carry the speed information of respective node. By utilizing the speed information of a neighboring node, every ordinary node in the network can predict its mobility pattern.

(2) *Localization Process.* After the mobility prediction of ordinary nodes, the process of localization starts. During this localization process the mobile AUVs broadcast localization

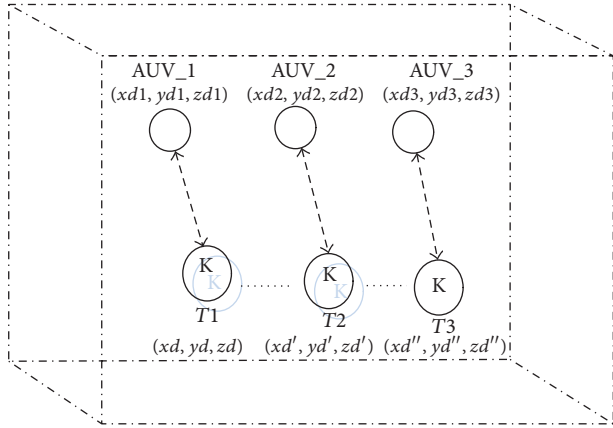


FIGURE 4: MobiL-AUV localization process.

beacon message to the ordinary nodes. The ordinary sensor nodes listen to these beacon messages. On receiving the beacon messages all the nodes estimate their distance from the mobile AUV by using an efficient measurement technique. As shown in Figure 4, the ordinary sensor node  $k$  initially at position  $(x_d, y_d, z_d)$  listens to the localization beacon message. At  $T_1$  from a mobile AUV located at  $(x_{d1}, y_{d1}, z_{d1})$  the ordinary node calculates the distance with respect to the mobile AUV by using the following equation [9].

$$\text{Distance } (d_k) = (T_{\text{send}} - T_{\text{recv}}) \times V, \quad (2)$$

where  $T_{\text{send}}$  is the time when the mobile AUV broadcasts the localization beacon message,  $T_{\text{recv}}$  is the time when the sensor node receives the beacon message, and  $V$  is the speed of sound in an underwater environment. The above process is repeated at  $T_2$  and  $T_3$ , until the ordinary nodes receive beacon messages from  $n_{\text{AUV}}$  different mobile AUVs. The flow of localization process is shown in Figure 5.

By substituting values of velocity  $(v_k^x(d), v_k^y(d))$  in (3), we can estimate the values of  $(x_{d'}, y_{d'})$  and  $(x_{d''}, y_{d''})$ .

$$\begin{aligned} x_{d'} &= x_d + (T_2 - T_1) \times v_k^x(d) \\ x_{d''} &= x_{d'} + (T_2 - T_1) \times v_k^x(d) \\ y_{d'} &= y_d + (T_2 - T_1) \times v_k^y(d) \\ y_{d''} &= y_{d'} + (T_2 - T_1) \times v_k^y(d). \end{aligned} \quad (3)$$

Here, in (3),  $(x_d, y_d, z_d)$ ,  $(x_{d'}, y_{d'}, z_{d'})$ , and  $(x_{d''}, y_{d''}, z_{d''})$  are the coordinates of node  $k$  and  $(x_{d1}, y_{d1}, z_{d1})$ ,  $(x_{d2}, y_{d2}, z_{d2})$ , and  $(x_{d3}, y_{d3}, z_{d3})$  are the coordinates of mobile AUVs at times  $T_1$ ,  $T_2$ , and  $T_3$ , respectively. By substituting values in (4), a node can estimate its position.

$$\begin{aligned} (x_d - x_{d1})^2 + (y_d - y_{d1})^2 + (z_d - z_{d1})^2 &= d_1^2 \\ (x_{d'} - x_{d2})^2 + (y_{d'} - y_{d2})^2 + (z_{d'} - z_{d2})^2 &= d_2^2 \\ (x_{d''} - x_{d3})^2 + (y_{d''} - y_{d3})^2 + (z_{d''} - z_{d3})^2 &= d_3^2. \end{aligned} \quad (4)$$

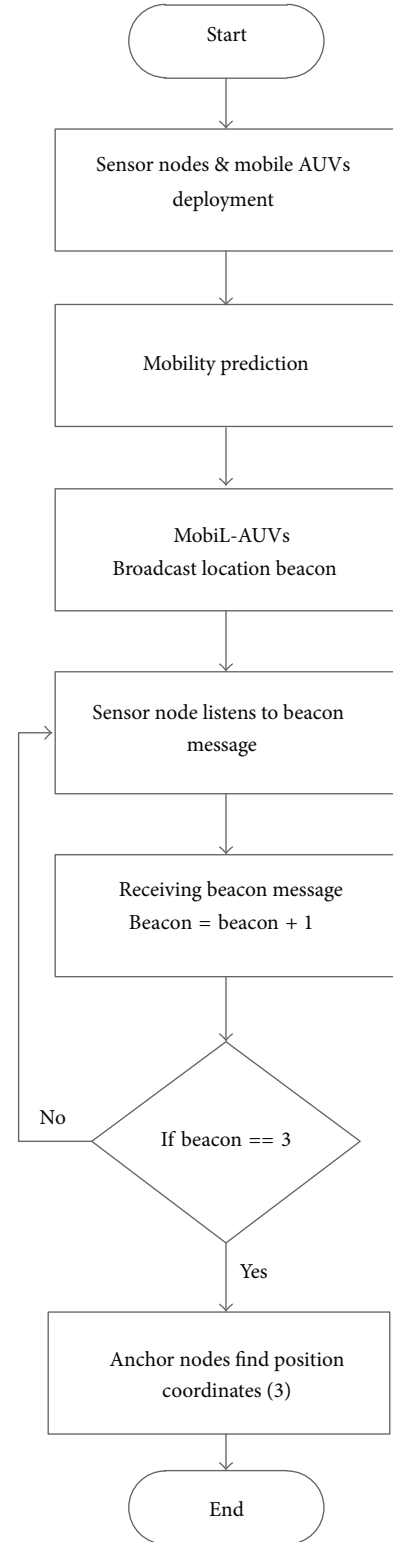


FIGURE 5: Flow chart of complete localization process.

3.2. CO-MobiL. After the process of mobility and localization, the process of data routing starts. For routing purposes, researchers have proposed different routing protocols [17–20]. In our proposed scheme, the source nodes broadcast

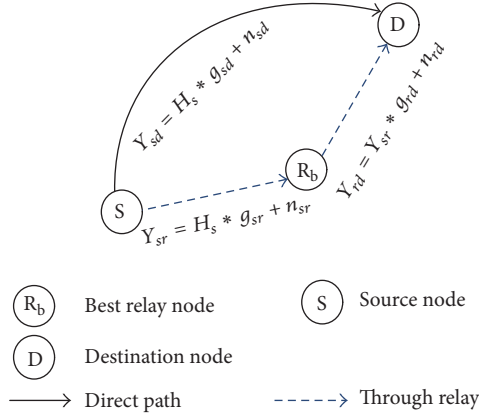


FIGURE 6: Network model.

data packets towards the destination. If the destination fails to receive the error-free packet then the best selected relay node will transmit data packet to the destination. At the destination, the MRC is used as a diversity combining technique. The relaying technique we used in our scheme is amplify-and-forward.

The underwater environment is harsh due to which the transmitted signals experience the noise, the multipath fading, and the interference. Every link from the source to the destination and from the relay to the destination is affected by the Rayleigh fading and additive white Gaussian noise (AWGN). The Rayleigh fading scatters the acoustic signal before its arrival at the destination because of less dominant propagation towards the destination and AWGN is used to simulate the background ambient noise of the channel.

The signal transmitted from the source node to the destination and the relay node to the destination is shown in Figure 6. The data packets are transmitted to both the destination and the relay node. If the source node fails to receive an error-free packet then the relay node transmits the same data packet from the alternative path to the destination, where the MRC diversity technique is used to combine the data packets to get the error-free packet. Its mathematical expression is given in (5), (6), and (7).

$$Y_{sd} = H_s \times g_{sd} + n_{sd}, \quad (5)$$

where  $Y_{sd}$  is the directly transmitted signal from the source to the destination,  $H_s$  denotes the channel from the source node to the destination node/sink,  $g_{sd}$  represents the broadcast information between the source and the destination, and  $n_{sd}$  shows the ambient noise added to channel  $H_s$ .

$$Y_{sr} = H_s \times g_{sr} + n_{sr}. \quad (6)$$

$Y_{sr}$  denotes the transmitted signal between the source and the relay as shown in Figure 6;  $g_{sr}$  shows the information transmitted by the source to the relay. Similarly,  $n_{sr}$  is the ambient noise added to channel  $H_s$ .

$$Y_{rd} = Y_{sr} \times g_{rd} + n_{rd}. \quad (7)$$

When the destination fails to receive data packets from the source node then relays transmit data packets according

to (7), where  $Y_{rd}$  is the transmitted data from the relay to the destination,  $g_{rd}$  denotes the information forwarded by the relay to the destination, and  $n_{rd}$  represents the noise added to the information  $Y_{sr}$  on the channel  $H_s$ .

There are two phases in CO-MobiL protocol.

- (1) Initialization phase
- (2) Data forwarding phase

In the first phase, nodes are deployed and each node finds its qualified relay and destination. This information is used in the second phase to transmit data from source to destination. The flow chart of CO-MobiL is shown in Figure 7.

(1) *Initialization Phase.* In this phase, nodes are randomly deployed underwater. Each node in the network finds a qualified relay node and destination for data transmission.

(i) *Relay Selection.* The relay selection mechanism is shown in Algorithm 1. All nodes in the network broadcast hello packet within their transmission range; the hello packet format is shown in Figure 8. On receiving the packet, each node sends a response packet to the nodes from which they receive a hello packet. A response packet is shown in Figure 9; it contains information about the node ID, depth, residual energy, and SNR of each node in the network. On receiving the response message, each node computes the weight function to find the qualified relay node in the network. The relay is selected based on the following parameters: SNR, residual energy (RE), depth of a node ( $D_i$ ), and delay ( $T_{\text{delay}}$ ) between the source node and the relay node. Each node maintains its neighbor information to gather data from the source node and then from these neighbors, a relay node is selected on the basis of a maximum weight function. The weight function of each node is computed as given in

$$W_{\text{weight}} = \frac{\text{RE} \times \text{SNR}}{D_{\text{depth}} \times T_{\text{delay}}} \quad (8)$$

(ii) *Destination Selection.* The destination node is selected on the basis of depth and residual energy. A node with higher residual energy which lies outside the depth threshold ( $D_{\text{dth}}$ ) but within the transmission range ( $T_{\text{range}}$ ) of the source node is a qualified destination ( $(T_{\text{range}}) \geq d_i \geq D_{\text{dth}}$ ). Figure 10 shows the destination node selection mechanism.  $D_{\text{dth}}$  is computed according to  $D_{\text{dth}} = d_i - 1$ , where  $d_i$  is the depth of any node of the network. The objective of  $D_{\text{dth}}$  is to minimize the number of hops to reduce the energy consumption. The weight function presented in (9) has been used to find the destination for a particular source node.

$$W_{\text{weight}} = \frac{\text{RE}}{d_i} \quad (9)$$

(2) *Data Forwarding Phase.* In this phase, the source node generates data packets and forwards these towards the destination and the relay node. At the destination node, a decision is made on the basis of instantaneous SNR between the source and the destination, whether the data received from the direct path is acceptable or a relay will be used for the data

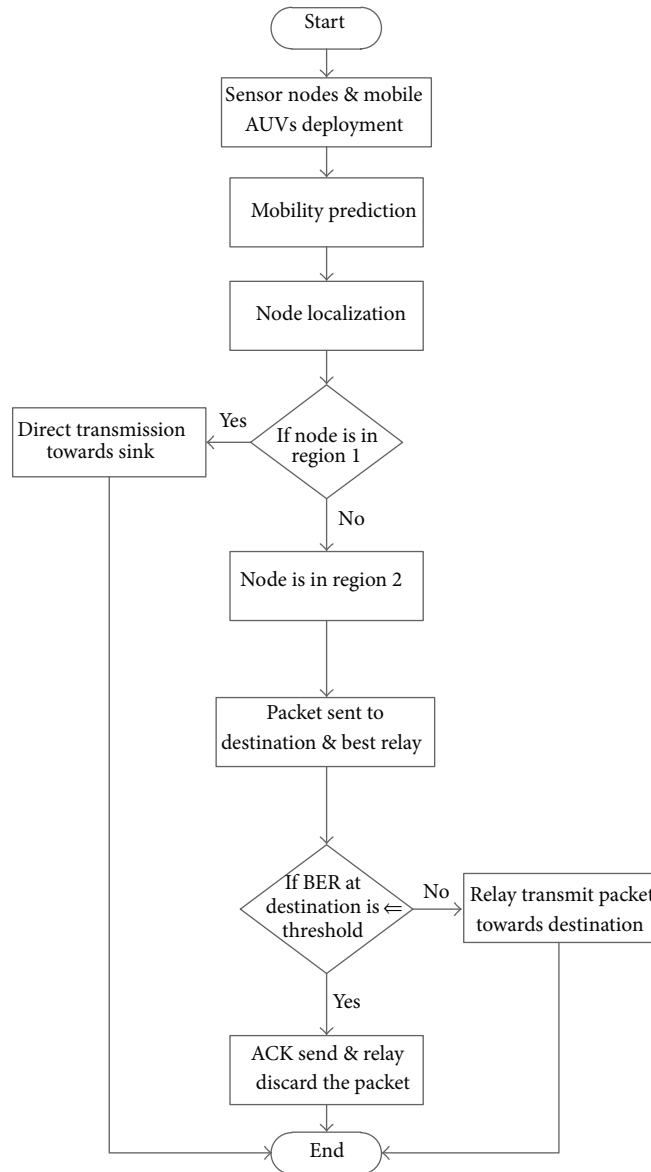


FIGURE 7: Complete cooperation process.

- (1)  $S_{source}$ : Source node
- (2)  $RE_{energy}$ : Residual energy of node
- (3)  $SNR_{sr}$ : Signal to Noise ratio between source and expected relay
- (4)  $D_{depth}$ : Depth of the node
- (5)  $T_{delay}$ : Delay between source and expected relay
- (6)  $S_{source}$  broadcast control packet
- (7) Neighboring nodes send response message
- (8) **if** Response message received = True, **then**;
- (9)     Compute  $W_{weight}$  function using equation (8).
- (10) **end if**
- (11) Node with maximum value of  $W_{weight}$  function is considered best relay

ALGORITHM 1: Relay node selection.



Node ID	Residual energy	Depth
---------	-----------------	-------

FIGURE 8: Control packet format.

Node ID	Residual energy	Depth	SNR <sub>(Link)</sub>
---------	-----------------	-------	-----------------------

FIGURE 9: Response packet format.

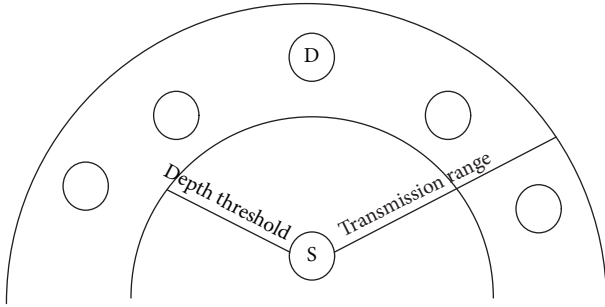


FIGURE 10: Destination node selection.

retransmission. The instantaneous SNR [36] is computed according to

$$\gamma_{AF} = \gamma_{s,r,d} + \gamma_{s,d}, \quad (10)$$

where  $\gamma_{AF}$  denotes amplify-and-forward signal,  $\gamma_{s,d}$  is the channel instantaneous SNR between the sender and the receiver, and  $\gamma_{s,r,d}$  is the equivalent instantaneous SNR between the source and the destination through the relay node. The equivalent instantaneous SNR of the relayed data/signal is computed as given in

$$\gamma_{s,r,d} = \frac{\gamma_{s,r}\gamma_{r,d}}{\gamma_{s,r} + \gamma_{r,d} + 1} \quad (11)$$

The network is divided into two regions as shown in Figure 11. The division reduces the energy consumption on the sensor nodes deployed near the sink. Due to the cooperation, the number of retransmissions increases at the destination resulting in more energy consumption. In order to mitigate the energy consumption and to prolong the network lifetime, we have limited the cooperation process in one region and in the second region direct data transmission is carried out. The working mechanism of both regions is discussed in the following subsections.

(i) *Direct Transmission Region.* The nodes closer to the onshore sink (nodes in transmission range of sink) are considered in region one, whereas the nodes that are not in the transmission range of onshore sink are considered in second region. Nodes that belong to region one forward their data packets towards the sink via direct transmission. As the distance between the source node and the sink in region one

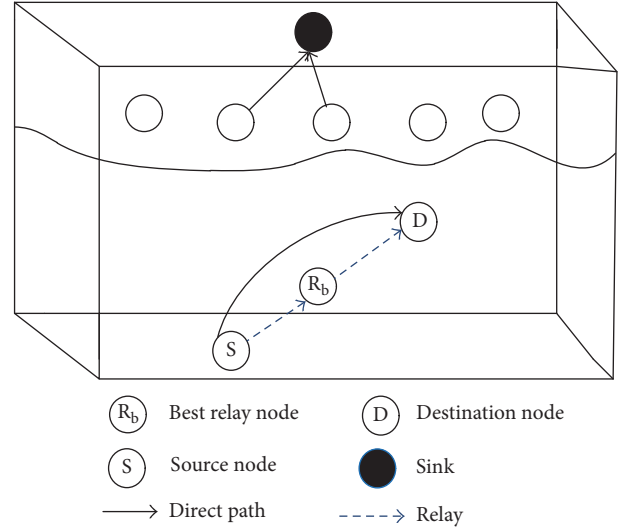


FIGURE 11: Data forwarding scenario.

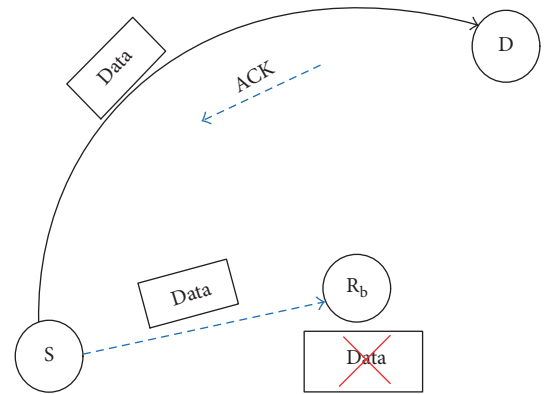


FIGURE 12: Data forwarding mechanism (ACK).

is not very large, lower amount of energy is consumed during direct transmissions and moreover the bit error rate (BER) in direct transmission is less compared to multihopping. In case there is no direct communication link between the nodes of region 1 and the sink then data forwarding is carried out via cooperation.

(ii) *Cooperative Region.* In region two, nodes transmit a data packet towards the destination via cooperation. During the cooperation process, the source node transmits data packets towards the destination and the best relay node. On receiving the data packet, the relay node waits for the acknowledgement (ACK) or negative acknowledgement (NACK) from the destination node. At the destination node BER is computed; if the BER is less than the BER threshold, the packet is accepted at the destination and ACK is sent. The relay node overhears ACK and discards the packet. This scenario is shown in Figure 12.

If the BER at destination is greater than the defined BER threshold, then NACK is sent and the relay node amplifies the received data and transmits the amplified data packets

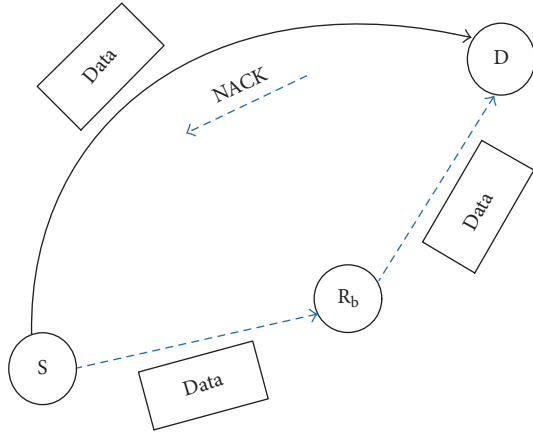


FIGURE 13: Data forwarding mechanism (NACK).

towards the destination. This scenario is shown in Figure 13. At the destination, MRC technique is used as a diversity combining technique. The signals from all links from the source to the destination and from the relay to the destination are added together to generate an original data packet.

*Amplify-and-Forward.* The relaying technique used in the proposed scheme is AF. In this technique, the source node sends data packets to the destination using a direct link; if the destination is unable to detect the packets using the direct link, the relay forwards the data to the destination. The relay amplifies the data and transmits the amplified data to destination.

#### 4. Experiments and Discussions

The performance of our proposed schemes (MobiL-AUV and CO-MobiL) is validated via simulations. We have claimed that region based cooperative routing minimizes the localization error and improves the data delivery ratio compared to existing traditional localization schemes (e.g., multihop localization, three-dimensional localization). Through simulation results we have showed that our schemes perform better than the existing compared schemes (MobiL and UDB).

The network field is considered of size  $500\text{ m} \times 500\text{ m}$  with transmission energy 2 joules (J) and receiving energy of 0.1J, and three AUVs are deployed to localize the ordinary nodes. The simulation parameters are given in Table 3 [2].

##### 4.1. Performance Metrics

- (i) Alive nodes: they are nodes that have a sufficient amount of energy to transmit a data packet from the source to the destination.
- (ii) Dead nodes: they are nodes that do not have a sufficient energy to transmit data from the source to the destination.

TABLE 3: Simulation setting parameters.

Simulation parameter	Value
Sensor nodes	50–200
Network dimension	$500\text{ m} \times 500\text{ m} \times 500\text{ m}$
Localization threshold	1 m
Transmission energy	2 J
Reception energy	0.1 J
Transmission range	100 m
Transmission frequency	22 KHz
Node mobility	1–5 m/s

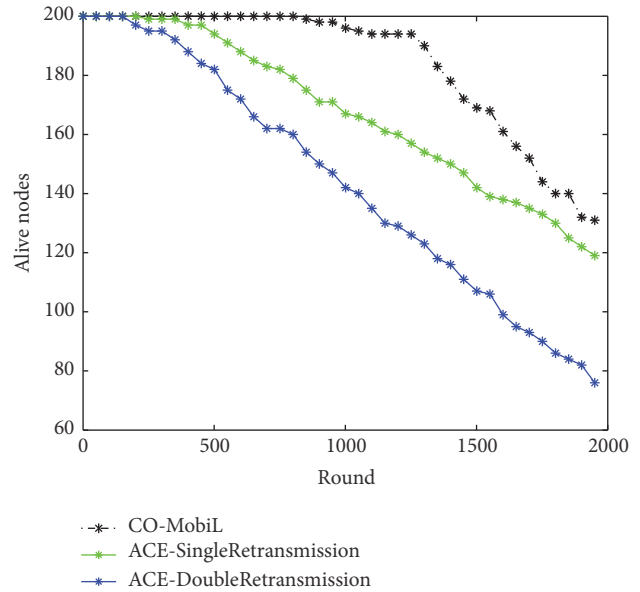


FIGURE 14: Alive nodes.

- (iii) Throughput: it means the successfully delivered data packets at the sink.
- (iv) Energy consumption: it is the total amount of energy consumed in the network by all nodes.
- (v) Localization error: it is the Euclidean distance between the original position of a sensor node  $(x_d, y_d, z_d)$  and the estimated position of a sensor node  $(x_d', y_d', z_d')$ .
- (vi) Localization coverage: It is the ratio of the number of successfully localized nodes to the total number of deployed nodes in the whole network. A node is successfully localized if its localization error is less than a predefined localization error threshold.

*4.2. Performance Discussion.* The alive nodes of all schemes are shown in Figure 14. The results show that our scheme outperforms the existing techniques; there are more alive nodes in our than others as shown in Figure 14. In ACE-DoubleRetransmission scheme [33], two relay nodes are used for a data transmission which increase the energy cost. More nodes are involved in a data transmission from the source to the destination which results in rapid energy depletion

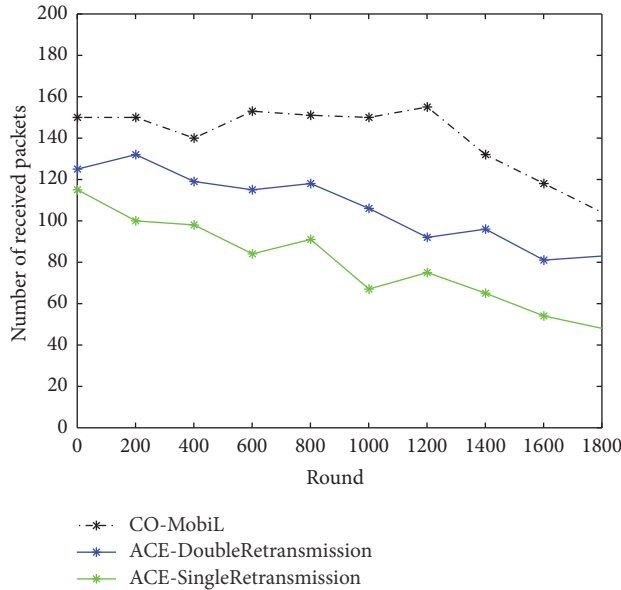


FIGURE 15: Throughput.

of nodes and the network lifetime is decreased. In ACE-SingleRetransmission, all nodes are involved in cooperation process; thus the energy consumption of the protocol is greater, whereas in our protocol due to efficient relay selection there is no need for the second relay to retransmit data to the destination. The nodes nearer to the sink directly transmit data to the sink, as the distance between the sink and these nodes is not so large; thus avoiding cooperation in the vicinity of the sink is to balance the energy consumption of the network.

Figure 15 represents the throughput for all the simulated schemes. The throughput of our scheme is more as compared to the other schemes. Due to the efficient selection of a relay node, a large number of data packets reached their destination successfully. If the destination fails to receive an error-free packet from the source, then the relay forwards data to the destination. In the other two schemes the relay is selected on the basis of depth and residual energy. The link between the source and the relay is ignored due to which the existing schemes experience packet drop and result in decreased throughput, whereas in the proposed scheme the relay is selected on the basis of depth, residual energy, and SNR of the link between the source and the relay. Due to the best selection of the relay, a greater number of data packets are successfully received at the destination. By using direct transmission in region 1 (i.e., closer to the sink), the probability of the BER decreases. As a result, high throughput is achieved.

In Figure 16, the energy consumption for all simulated schemes is shown. In ACE-DoubleRetransmission scheme, due to the selection of two relays, more energy is consumed. Whenever multiple relay nodes are used for retransmissions it results in high energy consumption. In ACE-SingleRetransmission scheme, all nodes are involved in the cooperation process which increases the energy consumption, whereas in our proposed scheme the region based cooperation is introduced. Nodes that are far from the onshore

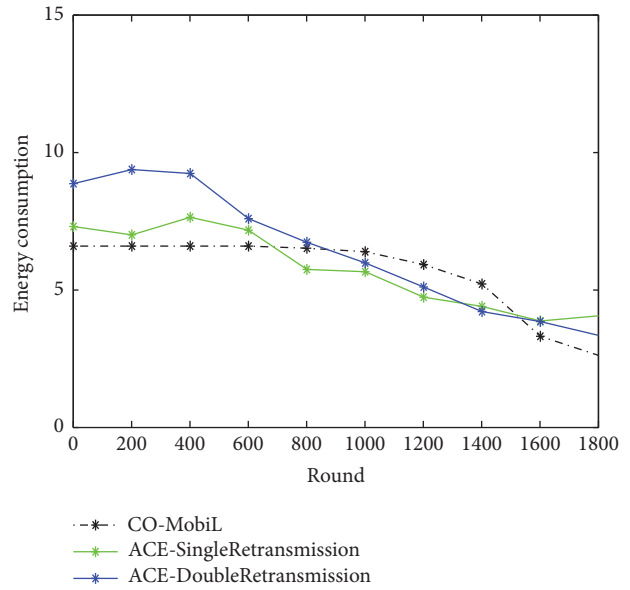


FIGURE 16: Energy consumption.

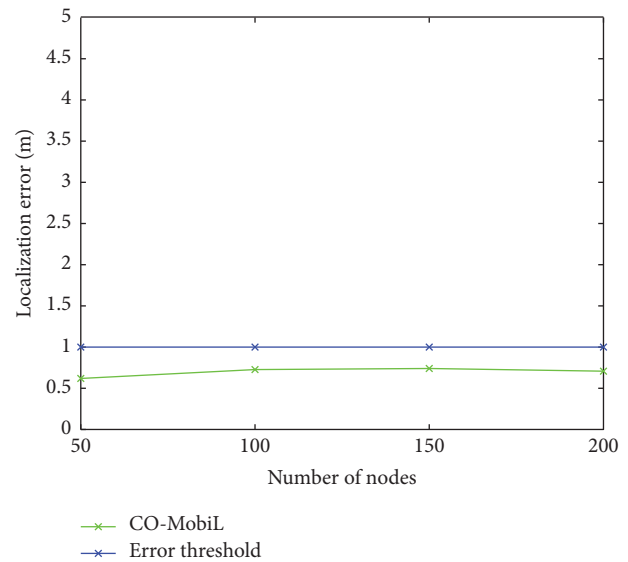


FIGURE 17: Localization error.

sink transmit data packets to the destination by cooperative routing and the other nodes that are part of region one (in the transmission of sink) directly transmit the data packets to the sink. Due to division of the field, the distance between the nodes of region one and the sink is reduced which results in less energy consumption.

The effects of localization error are shown in Figure 17. During simulations, we kept localization threshold 1m. The localization error parameter is plotted by varying the number of sensor nodes from 50 to 200. During the computation of localization error, the mobile AUV nodes are kept aside because these nodes are already localized. With the increase in the communication distance between ordinary nonlocalized nodes and mobile AUVs, the localization error increases. To reduce the localization error, in proposed scheme mobile

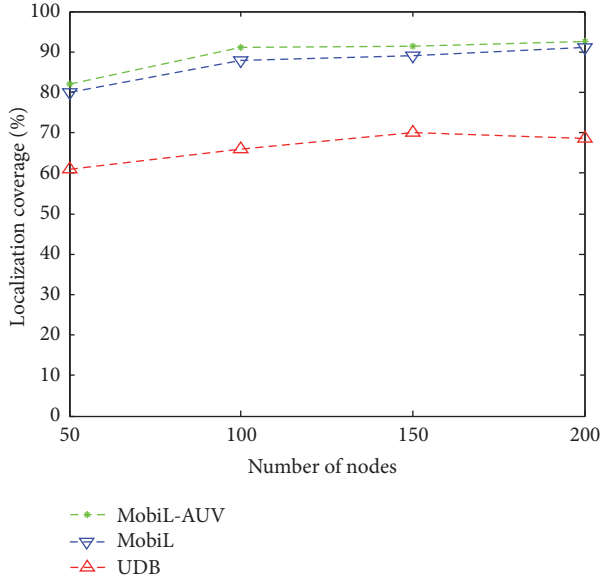


FIGURE 18: Localization coverage.

AUVs cover the large network volume that results in less localization error.

The localization coverage is shown in Figure 18. The result shows that our scheme MobiL-AUV outperforms MobiL [2] and UDB [37], in terms of localization coverage. With the increase in the number of nodes in proposed scheme, the localization coverage increases up to 80%. Our proposed scheme results in less localization error due to which a large number of nodes get localized. The localization coverage depends on the transmission range of the reference nodes. With the increase in the transmission range of reference nodes, more numbers of nodes get their position coordinates. Due to mobile AUVs, the greater number of ordinary non-localized nodes exists in the transmission range that results in the successful localization of large number of the ordinary nodes. In MobiL, reference nodes at the water surface are used to localize the network nodes, due to which a minimum number of nodes get localized, whereas, in UDB, directional beacons are broadcasted which are bounded by angle which varies from 10 to 90 degrees. Due to the involvement of angle, area of the beacon message is restricted due to which a lower number of nodes receive beacon messages resulting in less localization as compared to our proposed localization scheme (MobiL-AUV).

In Figure 19, the localization of the proposed scheme (MobiL-AUV) is presented at numerous time intervals with respect to node mobility. When the node mobility is 1 meter per second (m/s) as shown in Figure 19(a), MobiL-AUV broadcasts beacon message after every 1 second over a predefined path to localize the sensor nodes. The normalized values are shown in Figure 19(a); the success ratio increases with the increase in the nodes density from 50 to 200. It can be clearly seen that when beacon message interval is 1 s (1 second), the localization ratio increases as the node density changes from 50 to 200. The success ratio at lower interval

(1 s) is more because nodes receive beacon messages more often to localize themselves, whereas, at high intervals (2 s, 3 s), nodes receive fewer beacons resulting in fewer localized nodes in the network. Therefore, the success ratio is lower when the value of the time interval is high. As it can be seen from Figure 19(b), at 1 s the success ratio is 92% and for 2 s and 3 s, the localization success ratio is 89% and 86%, respectively. When the node mobility is 3 m/s, the localization success slightly decreases because nodes move out of the range from MobiL-AUV. As it is shown in Figure 19(b), with the increase in the node speed, the success ratio is affected. Similarly, From Figure 19(c), it further decreases as the node moves with the speed of 5 m/s. Therefore, we have used (2), to acquire the coordinates of the nodes by considering the speed of the nodes which gave us high success in estimating the coordinates of the nodes.

## 5. Performance Trade-Offs

Trade-offs made in our proposed protocol are given in Table 4. CO-MobiL achieves high network throughput at the cost of energy due to cooperation as shown in Figures 15 and 16, respectively. Figures 17 and 18 show that our proposed scheme MobiL-AUV minimized the localization error and enhanced the network localization coverage up to 80%, however, at the cost of energy consumption due to the localization of each node. In ACE-SingleRetransmission, all the nodes in the network take part in the cooperation process due to which packet delivery ratio increases at the cost of high energy consumption. Similarly, in ACE-DoubleRetransmission, two relays participate in the cooperation process to achieve network throughput by compromising on energy.

## 6. Conclusion

To minimize the localization and enhance the network throughput with efficient energy consumption, two routing protocols are proposed in this paper: MobiL-AUV is proposed for the localization of the network. In this scheme, three mobile AUVs are deployed and equipped with GPS and act as reference nodes. With the help of these reference nodes, all the ordinary sensor nodes in the network are localized. It achieved comparatively low localization error and high network coverage because of less distance between mobile AUVs and nodes. On the other hand, CO-MobiL is presented to utilize the localization done by the MobiL-AUV. In this scheme, the network is divided into two regions that minimized the data load and reduced the energy consumption of sensor nodes deployed near the sink. The use of MRC as diversity technique enhanced the network throughput, however, at the cost of energy. Extensive simulation results showed that MobiL-AUV performs better in terms of network coverage and localization success ratio than MobiL and UDB, whereas CO-MobiL outperformed its compared existing schemes (ACE-SingleRetransmission and ACE-DoubleRetransmission) in terms of throughput and energy consumption.

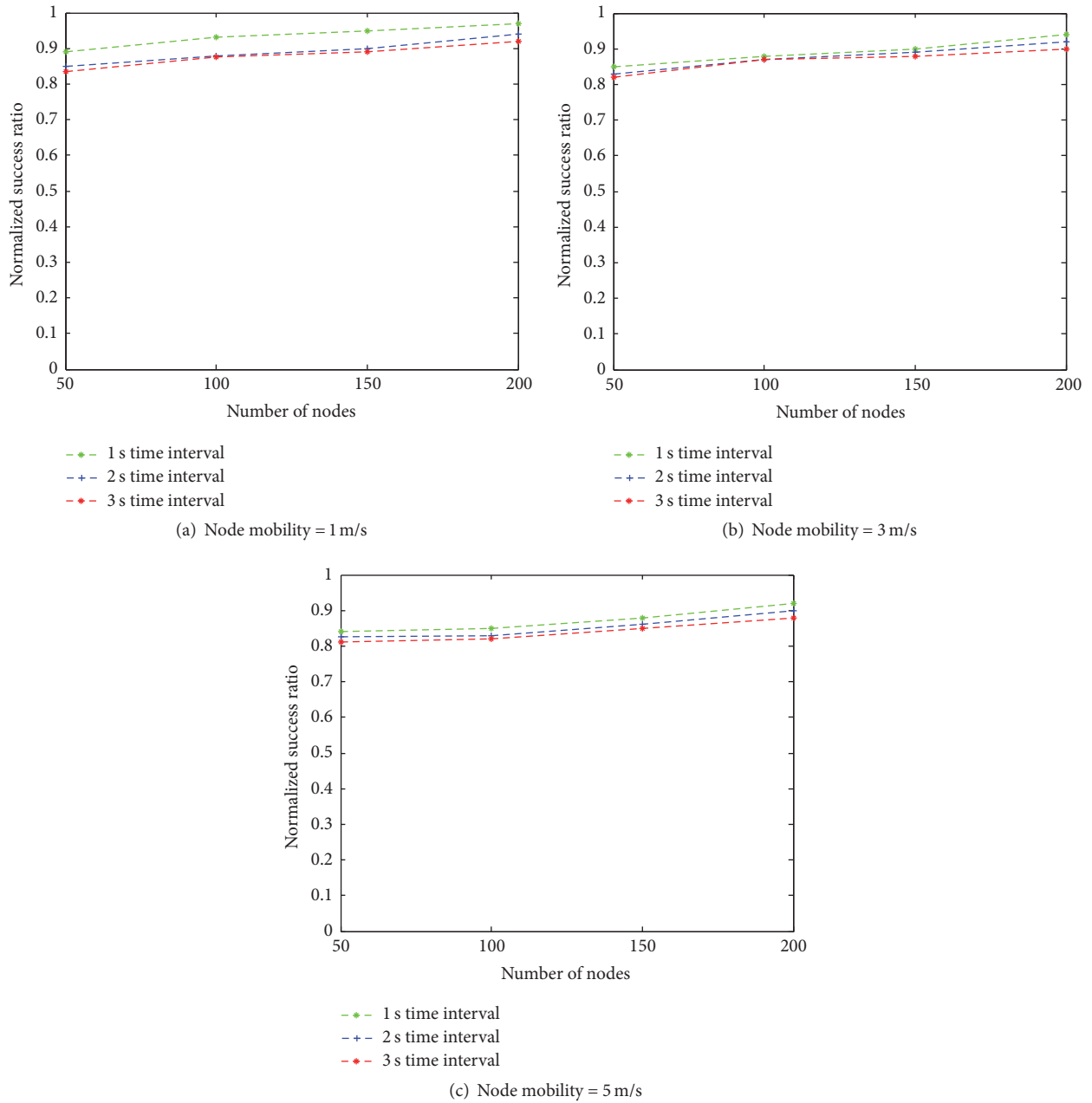


FIGURE 19: Normalized localization success ratio with respect to node mobility at various time intervals.

TABLE 4: Performance trade-off.

Protocol	Achieved parameters	Figure	Compromised parameter	Figure
CO-MobiL	Network lifetime and throughput	Figures 14 and 16	Energy consumption	Figure 17
MobiL-AUV	High localization coverage and low localization error	Figure 18	Energy consumption	Figure 17
ACE-SingleRetransmission	Throughput	Figure 16	Energy consumption	Figure 17
ACE-DoubleRetransmission	Throughput	Figure 16	Energy consumption	Figure 17

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This project was full financially supported by the King Saud University, through Vice Deanship of Research Chairs.

## References

- [1] P. Kirci and H. Chaouchi, "Recursive and ad hoc routing based localization in wireless sensor networks," *Computer Standards and Interfaces*, vol. 44, pp. 258–263, 2016.
- [2] T. Ojha and S. Misra, "MobiL: a 3-dimensional localization scheme for Mobile Underwater Sensor Networks," in *Proceedings of 2013 National Conference on Communications, NCC 2013*, ind, February 2013.
- [3] H. Maqsood, N. Javaid, A. Yahya, B. Ali, Z. A. Khan, and U. Qasim, "MobiL-AUV: AUV-aided localization scheme for underwater wireless sensor networks," in *Proceedings of 2016 10th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*, pp. 170–175, Fukuoka, Japan, July 2016.
- [4] S. Kundu and P. Sadhukhan, "Design and implementation of a time synchronization-free distributed localization scheme for underwater acoustic sensor network," in *Proceedings of 2015 2nd International Conference on Applications and Innovations in Mobile Computing, AIMoC 2015*, pp. 74–80, ind, February 2015.
- [5] M. Erol-Kantarci, H. T. Mouftah, and S. Oktug, "A survey of architectures and localization techniques for underwater acoustic sensor networks," *IEEE Communications Surveys and Tutorials*, vol. 13, no. 3, pp. 487–502, 2011.
- [6] G. Han, A. Qian, X. Li, J. Jiang, and L. Shu, "Performance evaluation of localization algorithms in large-scale Underwater Sensor Networks," in *Proceedings of 2013 8th International ICST Conference on Communications and Networking in China, CHI-NACOM 2013*, pp. 493–498, chn, August 2013.
- [7] D. Sivakumar and B. Sivakumar, "Localization using multilateration with RSS based random transmission directed localization," *Middle-East Journal of Scientific Research*, vol. 22, no. 1, pp. 45–50, 2014.
- [8] J. Xiong, Q. Qin, and K. Zeng, "A distance measurement wireless localization correction algorithm based on RSSI," in *Proceedings of 7th International Symposium on Computational Intelligence and Design, ISCID 2014*, pp. 276–278, chn, December 2014.
- [9] C. Zhang, Y. Liu, Z. Guo, G. Sun, and Y. Wang, "Minimum cost localization problem in three-dimensional ocean sensor networks," in *Proceedings of 2014 1st IEEE International Conference on Communications, ICC 2014*, pp. 496–501, aus, June 2014.
- [10] Z. Zhu, W. Guan, L. Liu, S. Li, S. Kong, and Y. Yan, "A multi-hop localization algorithm in underwater wireless sensor networks," in *Proceedings of the 6th International Conference on Wireless Communications and Signal Processing (WCSP' 14)*, pp. 1–6, Hefei, China, October 2014.
- [11] S. Poursheikhali and H. Zamiri-Jafarian, "TDOA based target localization in inhomogenous underwater wireless sensor network," in *Proceedings of 5th International Conference on Computer and Knowledge Engineering, ICCKE 2015*, pp. 1–6, irn.
- [12] Z. Zhou, Z. Peng, J.-H. Cui, Z. Shi, and A. Bagtzoglou, "Scalable localization with mobility prediction for underwater sensor networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 3, pp. 335–348, 2011.
- [13] G. Han, C. Zhang, L. Shu, and J. J. P. C. Rodrigues, "Impacts of deployment strategies on localization performance in underwater acoustic sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 3, pp. 1725–1733, 2015.
- [14] P. Carroll, S. Zhou, K. Mahmood, H. Zhou, X. Xu, and J.-H. Cui, "On-demand asynchronous localization for underwater sensor networks," in *Proceedings of OCEANS 2012 MTS/IEEE Hampton Roads Conference: Harnessing the Power of the Ocean*, usa, October 2012.
- [15] M. T. Isik and O. B. Akan, "A three dimensional localization algorithm for underwater acoustic sensor networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 9, pp. 4457–4463, 2009.
- [16] H. Luo, K. Wu, Y. Gong, and L. M. Ni, "Localization for drifting restricted floating ocean sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 9968–9981, 2016.
- [17] F. Mansourkiaie and M. H. Ahmed, "Joint cooperative routing and power allocation for collision minimization in wireless sensor networks with multiple flows," *IEEE Wireless Communications Letters*, vol. 4, no. 1, 2015.
- [18] W. Fang, F. Liu, F. Yang, L. Shu, and S. Nishio, "Energy-efficient cooperative communication for data transmission in wireless sensor networks," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 4, pp. 2185–2192, 2010.
- [19] S. S. Ikki and M. H. Ahmed, "Performance analysis of cooperative diversity with incremental-best-relay technique over rayleigh fading channels," *IEEE Transactions on Communications*, vol. 59, no. 8, pp. 2152–2161, 2011.
- [20] Y. Wei and D.-S. Kim, "Exploiting cooperative relay for reliable communications in underwater acoustic sensor networks," in *Proceedings of 33rd Annual IEEE Military Communications Conference, MILCOM 2014*, pp. 518–524, usa, October 2014.
- [21] F. Mansourkiaie and M. H. Ahmed, "Optimal and near-optimal cooperative routing and power allocation for collision minimization in wireless sensor networks," *IEEE Sensors Journal*, vol. 16, no. 5, pp. 1398–1411, 2016.
- [22] S. Ahmed, N. Javaid, F. A. Khan et al., "Co-UWSN: Cooperative energy-efficient protocol for underwater WSNs," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 891410, 2015.
- [23] Z. Zhou, S. Zhou, S. Cui, and J. Cui, "Energy-efficient cooperative communication in a clustered wireless sensor network," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3618–3628, 2008.
- [24] Y. Li, Z. Jin, X. Wang, and Z. Liu, "Relay selection and optimization algorithm of power allocation based on channel delay for UWSN," *International Journal of Future Generation Communication and Networking*, vol. 9, no. 2, pp. 103–112, 2016.
- [25] A. Umar, M. Akbar, Z. Iqbal, Z. A. Khan, U. Qasim, and N. Javaid, "Cooperative partner nodes selection criteria for cooperative routing in underwater WSNs," in *Proceedings of 5th National Symposium on Information Technology: Towards New Smart World, NSITNSW 2015*, sau, February 2015.
- [26] Z. Liao, D. Li, and J. Chen, "A network access mechanism for multihop underwater acoustic local area networks," *IEEE Sensors Journal*, vol. 16, no. 10, pp. 3914–3926, 2016.
- [27] H. Wu, M. Chen, and X. Guan, "A network coding based routing protocol for underwater sensor networks," *Sensors*, vol. 12, no. 4, pp. 4559–4577, 2012.

- [28] A. Wahid and D. Kim, "An energy efficient localization-free routing protocol for underwater wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 307246, 11 pages, 2012.
- [29] A. Wahid, S. Lee, and D. Kim, "An energy-efficient routing protocol for UWSNs using physical distance and residual energy," in *Proceedings of the OCEANS '11*, pp. 1–6, Santander, Spain, June 2011.
- [30] M. Xu, G. Z. Liu, and H. F. Wu, "An energy-efficient routing algorithm for underwater wireless sensor networks inspired by ultrasonic frogs," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 351520, 12 pages, 2014.
- [31] J. Cao, J. Dou, and S. Dong, "Balance transmission mechanism in underwater acoustic sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 429340, 12 pages, 2015.
- [32] H. Yang, F. Ren, C. Lin, and B. Liu, "Energy efficient cooperation in underwater sensor networks," in *Proceedings of 2010 IEEE 18th International Workshop on Quality of Service, IWQoS 2010*, chn, June 2010.
- [33] H. Nasir, N. Javaid, M. Murtaza et al., "ACE: Adaptive cooperation in EEDBR for underwater wireless sensor networks," in *Proceedings of 9th IEEE International Conference on Broadband and Wireless Computing, Communication and Applications, BWCCA 2014*, pp. 8–14, chn, November 2014.
- [34] M. Erol, L. F. M. Vieira, and M. Gerla, "AUV-aided localization for underwater sensor networks," in *Proceedings of Proceeding of the 2nd Annual International Conference on Wireless Algorithms, Systems, and Applications (WASA '07)*, pp. 44–54, Chicago, Ill, USA, August 2007.
- [35] A. K. Othman, A. E. Adams, and C. C. Tsimenidis, "Node discovery protocol and localization for distributed underwater acoustic networks," in *Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services*, Guadeloupe, France, February 2006.
- [36] S. S. Ikki and M. H. Ahmed, "Performance analysis of incremental-relaying cooperative-diversity networks over Rayleigh fading channels," *IET Communications*, vol. 5, no. 3, pp. 337–349, 2011.
- [37] H. Luo, Y. Zhao, Z. Guo, S. Liu, P. Chen, and L. M. Ni, "UDB: using directional beacons for localization in underwater sensor networks," in *Proceedings of 2008 14th IEEE International Conference on Parallel and Distributed Systems, ICPADS'08*, pp. 551–558, aus, December 2008.

## Research Article

# Weighted-DESYNC and Its Application to End-to-End Throughput Fairness in Wireless Multihop Network

Ui-Seong Yu,<sup>1</sup> Ji-Young Jung,<sup>1</sup> Eutteum Kong,<sup>1</sup> HyungSeok Choi,<sup>2</sup> and Jung-Ryun Lee<sup>1</sup>

<sup>1</sup>*School of Electrical & Electronic Engineering, Chung-Ang University, Seoul, Republic of Korea*

<sup>2</sup>*Agency for Defense Development, Daejeon, Republic of Korea*

Correspondence should be addressed to Jung-Ryun Lee; jrlee@cau.ac.kr

Received 8 December 2016; Revised 21 March 2017; Accepted 29 March 2017; Published 13 April 2017

Academic Editor: Hideyuki Takahashi

Copyright © 2017 Ui-Seong Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The end-to-end throughput of a routing path in wireless multihop network is restricted by a bottleneck node that has the smallest bandwidth among the nodes on the routing path. In this study, we propose a method for resolving the bottleneck-node problem in multihop networks, which is based on multihop DESYNC (MH-DESYNC) algorithm that is a bioinspired resource allocation method developed for use in multihop environments and enables fair resource allocation among nearby (up to two hops) neighbors. Based on MH-DESYNC, we newly propose weighted-DESYNC (W-DESYNC) as a tool artificially to control the amount of resource allocated to the specific user and thus to achieve throughput fairness over a routing path. Proposed W-DESYNC employs the weight factor of a link to determine the amount of bandwidth allocated to a node. By letting the weight factor be the link quality of a routing path and making it the same across a routing path via Cucker-Smale flocking model, we can obtain throughput fairness over a routing path. The simulation results show that the proposed algorithm achieves throughput fairness over a routing path and can increase total end-to-end throughput in wireless multihop networks.

## 1. Introduction

Wireless multihop networks are used in various environments such as sensor networks, Internet of Things (IoT), wireless body area networks (WBAN), wireless ad hoc network (WANET), wireless mesh network (WMN), and vehicle ad hoc network (VANET). In such a large number of fields, various devices such as sensors, smart phones, and automobiles that support various types of wireless communication continue to increase in number, so it is essential to utilize resources more efficiently.

Because a wireless node can communicate directly with only the node within the transmission range, in order for a source-destination pair to communicate with each other over a long distance, nodes serving as routers for delivering messages are required. In such wireless multihop networks, bottleneck-node problems can occur when the source and destination nodes communicate via a multihop routing path. The bottleneck node means the node whose allocated bandwidth is the lowest across the routing path connecting

the source and the destination. In a multihop environment, bottleneck nodes are the main cause of decrease of end-to-end throughput. This bottleneck node is mainly caused by sharing the limited amount of resources while crossing certain other paths in the multihop communication process [1].

For the purpose of organizational communication between nodes, a routing process that operates in a distributed way is essential for a node to efficiently communicate with other nodes. In general, the performance of a routing process such as an end-to-end throughput between source and destination pair is mainly restricted by a bottleneck node, which serves as a relay node on multiple routing paths or suffers from a low link quality and thus is allocated the smallest bandwidth (lowest data rate) among the nodes in the routing path [2]. As a result, these bottleneck nodes can decrease the end-to-end throughput of the routing path and increase the queuing delay between source and destination nodes. One simple method to solve the bottleneck-node problem is to replace the routing path with one of higher



performance, but this method has a limitation: there might not be a sufficient number of nodes in the network, meaning that multiuser diversity cannot be exploited in the given network environment. For this reason, researchers have tried to solve the bottleneck-node problem via scheduling, mostly by giving a high priority to bottleneck nodes for data processing [3].

However, under the contention-based medium access protocol, typical scheduling methods to resolve the bottleneck-node problem require high complexity and additional procedures for prioritization for scheduling. Furthermore, in wireless multihop networks, scheduling-based solutions for the bottleneck-node problem show poor scalability because of rapid changes in the network topology. Thus, scheduling-based solutions for the bottleneck-node problem are not suitable considering the high computational complexity, network overhead, and low scalability.

## 2. Related Work

The unfair and insufficient resource allocation of a bottleneck node causes the throughput decrease and delay increase in the end-to-end packet delivery of a routing path. To resolve this problem, the fair use of throughput in multihop networks is emphasized [3]. Fair use of throughput in multihop networks can be implemented in three ways: (1) the selection of the routing path to avoid the bottleneck node and thus guarantee throughput fairness in the routing process, (2) the fair allocation of bandwidths among the nodes on the routing path via the multiple-access control (MAC) protocol, and (3) the use of cross-layer optimization for both MAC and routing protocols to overcome the fundamental limits of these protocols.

*2.1. Related Works: Routing Protocol.* In an effort to solve the bottleneck-node problem via routing procedures, the concept of the parallel path has been proposed. In [4], multiple disjoint routing paths for a source-destination pair were provided through the unit cell division and signaling message exchange and constant per-node transmission rate is maintained. In [5], the authors obtained the minimum hop bandwidth requirements and network optimization and extended the straightforward precomputation scheme to solve the bottleneck for QoS, by using a straightforward precomputation scheme of standard Bellman-Ford algorithm. In [6], an optimal path was selected using distance-vector routing techniques with consideration of QoS, and the estimated available bandwidth of the path was set to the maximum bandwidth. In an environment where the network nodes have mobility and thus the network topology dynamically changes, these efforts to solve bottleneck-node problem via routing protocols have limits, as the prompt adaptation and preservation of fair resource allocation are challenging because of signaling overheads and the redundant delay time required to reestablish the routing path via techniques such as route reply (RREP) or route request (RREQ) messages.

*2.2. Related Works: MAC Protocol.* In multihop ad hoc networks, end-to-end bandwidth and delay are highly dependent

on the network topology [7]. Therefore, the MAC layer which controls the resource allocation of a node can be a method for throughput fairness among nodes on the same routing path. Fair resource allocation among users is provided by controlling the frame length of a user for adjusting the amount of data transferred in a scheduling-based MAC protocol (such as the time-division multiple-access (TDMA) protocol) or the contention window size of a user for the control of the user-based transmission opportunity in a contention-based MAC protocol (such as CSMA/CA) [8]. In [9], a method that ensures fairness using flow control and a virtual queue-scheduling scheme is suggested [9]. In [10], a fixed-length bitmap vector is used in each packet header for exchanging slot timing information between immediate neighbors and those separated by up to two hops. Each node iteratively moves its slot near the middle of the slots of its neighbors until an allocation pattern becomes stable and fair allocation of resources is obtained for all the nodes in the neighborhood.

*2.3. Related Works: Cross-Layer Approach.* Although each communication layer has its specific functionality such as congestion control at the transport layer, routing at the network layer, and scheduling/power control at the MAC/physical (PHY) layer, it is expected that interactions between the various layers are necessary for achieving the optimal performance [11]. In [12], distributed multiuser scheduling was proposed, which is a carrier sense multiple access with collision avoidance-based technique considering both the PHY and MAC layers. This method calculates the signal-to-noise ratio (SNR) of each node and determines the contention window size according to the average SNR. In the distributed queuing collision avoidance scheme proposed in [13], a virtual priority function is used for queuing scheduling to ensure fairness and collision avoidance. The dual-based method is implemented as follows. At the transport layer, end-to-end sessions adjust their rates in a distributed manner to attain proportionally fair session rates given specific link rates. At the link layer, the link-attempt probabilities are adjusted with local information, so that the bandwidth bottlenecks are alleviated and the aggregate utilities can be further increased [14].

In this paper, we propose a resource allocation scheme called W-DESYNC based on multihop DESYNC in wireless network, as a tool artificially to control the amount of resource allocated to the specific user and thus to achieve throughput fairness over a routing path. In W-DESYNC, the amount of resource allocated to each node is decided not by the number of neighbor nodes but by the weight factor of the node. In order to make the weight factors of nodes on the same routing path the same, we apply Cucker-Smale flocking model to the weight factors of node where the weight factor of a node is set to SNR and therefore achieves the throughput fairness over a routing path. In this way, the proposed W-DESYNC method enables the nodes on the routing path to have the same throughput and the bottleneck problem can be solved to improve the network performance.

The configuration of the paper is as follows. Section 3 discusses the previous MH-DESYNC. In Section 4, we

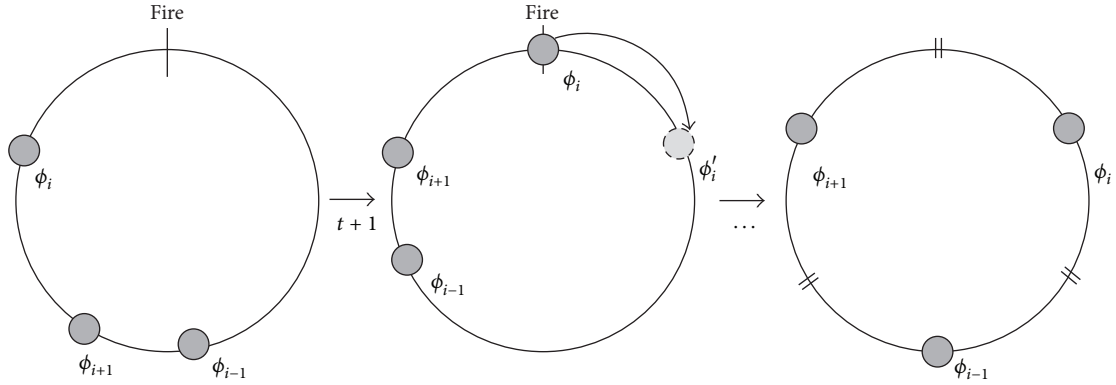


FIGURE 1: Phase update in DESYNC.

explain the proposed W-DESYNC method, which will act as a tool to assign priorities on the links of a routing path, and the detailed algorithm to achieve throughput fairness over a routing path by applying Cucker-Smale flocking model for the purpose of making the weight factors of the links the same across a routing path. Section 5 evaluates the performances of the proposed scheme and Section 6 concludes.

### 3. Previous DESYNC and MH-DESYNC Methods

In this section, we review bioinspired algorithms that have been proposed, with the purpose of providing a solution for distributed resource allocation in multihop networks. Bioinspired methods are derived by modeling the operational principles of living organisms in nature through observation and expressing these principles in mathematical formulae. Bioinspired algorithms have autonomous properties, such as self-organization, self-learning, and self-management, rather than being externally controlled by a centralized coordinator. Such characteristics are highly suitable for application to distributed processing and device-to-device communication, which are strongly required functionalities in wireless multihop networks.

**3.1. DESYNC.** We review the DESYNC method, which is based on the inverse concept of firefly synchronization [15–17]. Firefly synchronization imitates a phenomenon where fireflies with different glittering phases glitter simultaneously through interaction among them. Conversely, the DESYNC method shows desynchronization between glittering phases of fireflies by trying to make the glittering phase of a node as far as possible from the glittering phases of other fireflies. As a result, the glittering phases of all fireflies become evenly spaced in a distributed way.

Suppose that there are  $N$  nodes in a fully connected network. Each node performs a firing periodically with a period  $T$ . The phase of node  $i$  at time  $t$  is denoted as  $\phi_i(t) \in [0, 1]$ , where the phases 0 and 1 are identical and  $0 \leq i \leq N-1$ . When node  $i$  reaches the node of its cycle ( $\phi_i(t) = 1$ ), it fires to indicate the termination of its cycle to the other nodes. After firing, the node resets its phase to  $\phi_i(t^+) = 0$ . Node  $i$

records the times of the following two firing events: the one that precedes its own firing (previous firing  $\phi_{i+1}(t)$ ) and the one that occurs just afterwards (next firing  $\phi_{i-1}(t)$ ). Node  $i$  calculates the midpoint of its two reference phases as  $\phi_{\text{mid}} = (1/2)[\phi_{i+1}(t) + \phi_{i-1}(t)]$  and jumps towards it as follows:

$$\phi'_i(t) = (1 - \alpha)\phi_i(t) + \alpha\phi_{\text{mid}}(t), \quad (1)$$

where  $\alpha \in [0, 1]$  is a parameter that scales how far node  $i$  moves from its current phase towards the desired midpoint. Figure 1 shows the phase update procedure in DESYNC.

**3.2. Application of DESYNC to Resource Allocation in TDMA.** The operational principle of the DESYNC can be applied to TDMA-based networks to obtain distributed and fair resource allocation among all the nodes in a network with the assumption that all nodes are fully connected. In the so-called DESYNC-TDMA, the amount and the range of TDMA time slots allocated for the data transfer of the node  $i$  at time  $t+1$  are determined by two reference phases of node  $i$  at time  $t$ ,  $\phi_{i-1}(t)$  and  $\phi_{i+1}(t)$ . The left-mid phase and right-mid phase of node  $i$  at time  $t$  are defined as follows.

$$\begin{aligned} \phi_{i,\text{left-mid}}(t) &= \frac{1}{2} [\phi_i(t) + \phi_{i-1}(t)], \\ \phi_{i,\text{right-mid}}(t) &= \frac{1}{2} [\phi_i(t) + \phi_{i+1}(t)]. \end{aligned} \quad (2)$$

Based on the above left- and right-mid phases, node  $i$  occupies the TDMA time slots beginning at  $\phi_{i,\text{left-mid}}(t)$  and ending at  $\phi_{i,\text{right-mid}}(t)$ . In this way, all of the nodes occupy the nonoverlapping time slots that cover a period  $T$  evenly. Figure 2 shows the updating process of firing phase and the TDMA slot allocation procedure using left- and right-mid phases of node  $i$ .

**3.3. Multihop DESYNC (MH-DESYNC).** Because the DESYNC-TDMA method assumes a fully connected network topology, it causes a hidden-node problem and thus cannot be directly applied to multihop networks. Therefore, the MH-DESYNC method was proposed, with the purpose of facilitating collision-free and fair resource allocation among not only one-hop neighbors but also two-hop neighbors in wireless multihop networks [18].

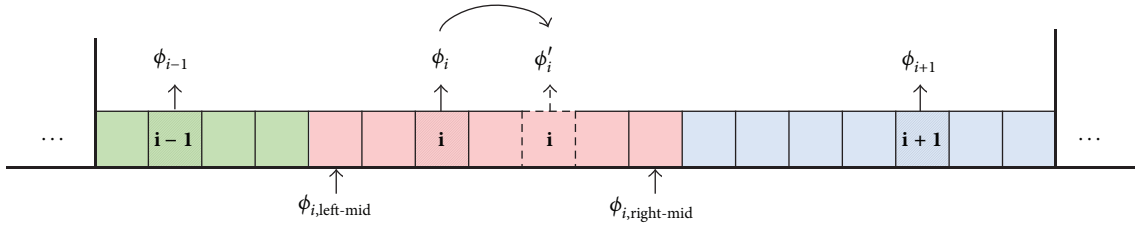


FIGURE 2: TDMA time-slot allocation of node  $i$  in DESYNC-TDMA.

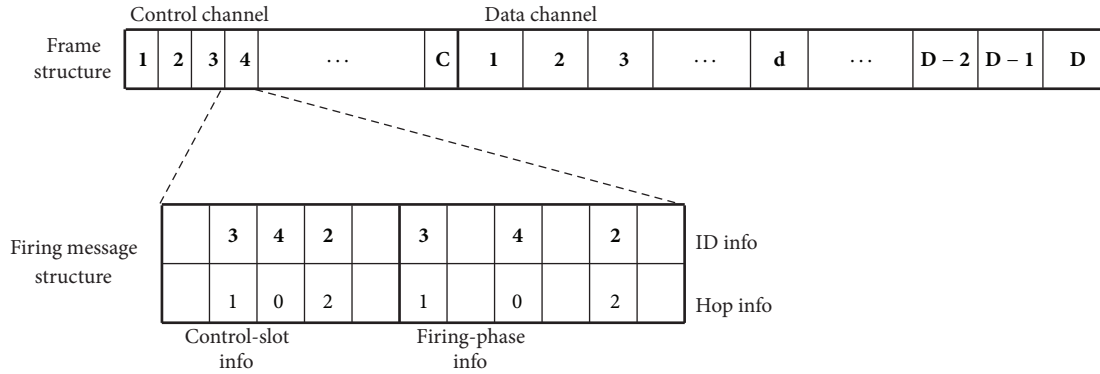


FIGURE 3: Physical and logical firing structure.

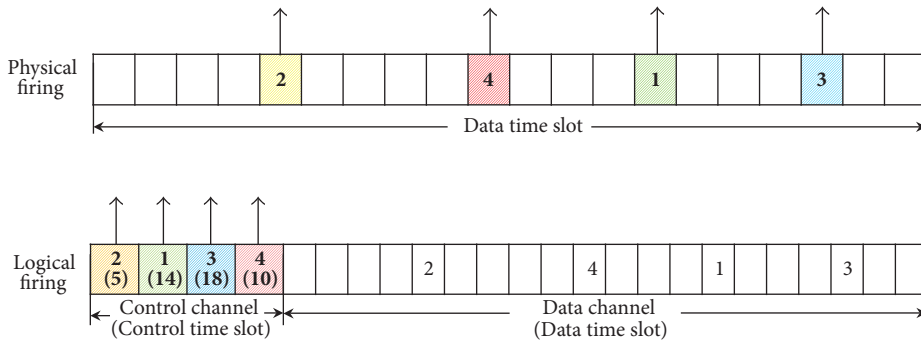


FIGURE 4: Frame structure and firing message in MH-DESYNC.

Compared with the DESYNC-TDMA method, the main difference of the MH-DESYNC method is virtual firing. In the DESYNC-TDMA method, the firing of a node is identified by the broadcasting of the firing phase of the node to its neighboring nodes via a firing signal. The firing of a node is a series of short interrupt messages that are transmitted in a given data time slot, and the firing signal is referred to as a physical firing signal, as shown in Figure 3. In contrast, with virtual firing, each node records both its own firing phase and its one-hop neighboring nodes firing phases in its own firing message, which is generated in every frame and is sent to its one-hop neighboring nodes via the control time slot of the frame. It is comparatively noted that the physical firing signal in the DESYNC-TDMA method notifies the phase of a node implicitly, by the location of the data time slot where a series of interrupt messages are sent. The

virtual firing of the MH-DESYNC method is facilitated by a new frame structure and the firing-message structure, which are shown in Figure 4. In addition, the detailed operational procedures of the MH-DESYNC method for control time-slot allocation, firing-phase allocation and updating, and data time-slot allocation are provided. Regarding control time-slot allocation, the initial control time-slot allocation procedure for a newly entering node and the detailed method for control time-slot collision detection and resolution are provided. For firing-phase allocation and updating, the initial firing-phase allocation and updating procedures for nodes that succeed in cooccupying a control time slot and the firing-phase updating procedure are explained. Regarding data time-slot allocation, the methods for detecting and resolving firing-phase collision and allocating an actual data time slot to each node are provided.

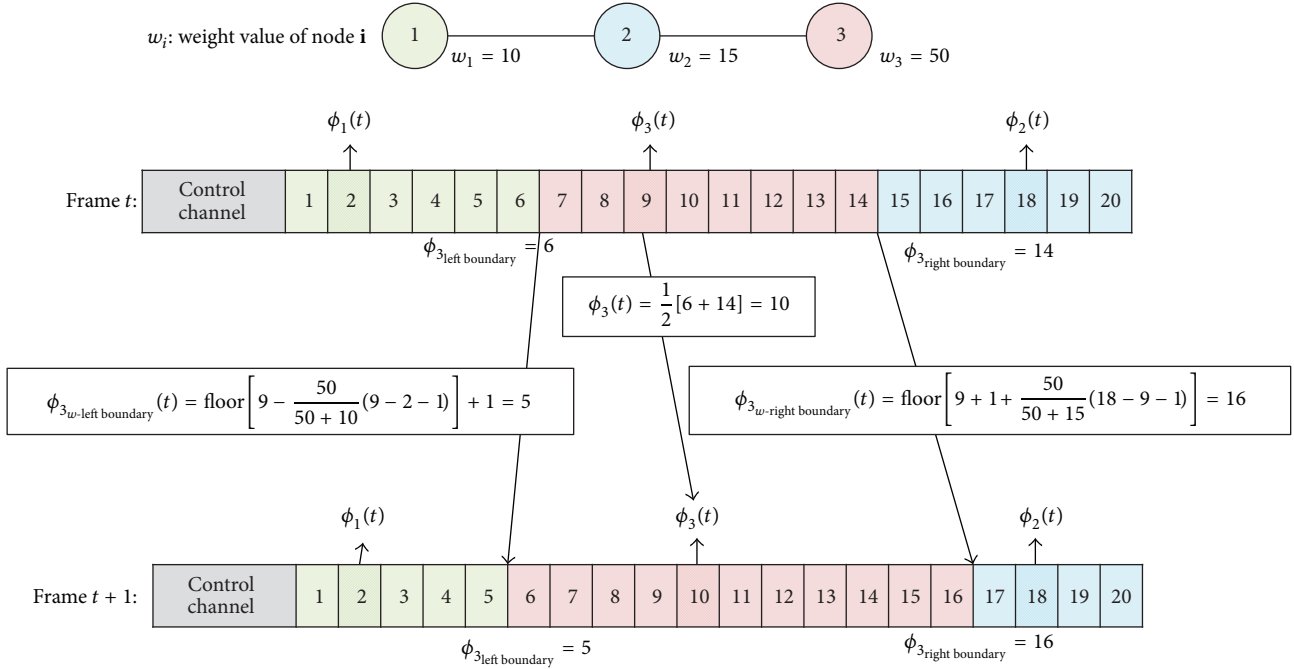


FIGURE 5: Firing-phase update data and time-slot allocation through firing in the W-DESYNC method.

#### 4. Proposed Algorithm for Throughput Fairness over a Routing Path

4.1. *Proposed W-DESYNC Method.* The existing MH-DESYNC method can carry out fair resource allocation under wireless multihop network in a distributed way [18]. Specifically, in MH-DESYNC, the amount of resource allocated to node  $i$  is  $1/(N1(i) + N2(i) + 1)$ , where  $N1(i)$  and  $N2(i)$  are the numbers of one-hop and two-hop neighbor nodes of node  $i$ , respectively. Here, it is noted that the amount of resource allocated to a node under MH-DESYNC is determined passively; that is, a node cannot determine the amount of resource allocated to itself but it is just determined by the network environment. As we mentioned in Section 1, this unfair resource allocation naturally involves the so-called bottleneck node and causes a max-min problem on the end-to-end throughput of a routing path.

In this section, the proposed W-DESYNC method is described as a tool to provide throughput fairness over a routing path. While the left boundary (right boundary) of node  $i$  is determined by the *midpoint* of the firing phases of nodes  $i - 1$  ( $i + 1$ ) and  $i$  in MH-DESYNC, the proposed W-DESYNC method dynamically determines both the forward mid and backward mid of a node, considering the amount of bandwidth necessary for each node. The relative priority for the amount of bandwidth of a node is expressed by a weight factor. Let the weight factor of node  $i$  be  $w_i$ . Then, the forward mid and backward mid for each node  $i$  are calculated as follows.

$$\arg \min_{\forall i \in N, (i \neq k)} \left( \phi_{\text{left boundary}} = \text{ceil} \left[ \phi_i(t) - \frac{w_i}{w_{i-1} + w_i} (\phi_i(t) - \phi_{i-1}(t) - 1) \right] + 1 \right),$$

$$\arg \min_{\forall i \in N, (i \neq k)} \left( \phi_{\text{right boundary}} = \text{ceil} \left[ \phi_i(t) + 1 + \frac{w_i}{w_{i-1} + w_i} (\phi_i(t) - \phi_{i-1}(t) - 1) \right] \right). \quad (3)$$

Here, it is noted that the order of the firing phases can be changed under the W-DESYNC algorithm, whereas it cannot be changed once it is decided under the DESYNC and MH-DESYNC algorithms. Thus, the W-DESYNC method requires each node  $i$  to update the indices of its forward and backward nodes, by calculating the left boundary and right boundary through (3) for all neighbors of node  $i$  at every frame. The resources allocated to node  $i$  at frame  $t$  are given by

$$\left[ \phi_{i_{\text{left boundary}}(t)} \cdot \phi_{i_{\text{right boundary}}(t)} \right]. \quad (4)$$

After the resource allocation is finished through the aforementioned process, the firing phase is updated, as follows:

$$\phi'_i(t) = \frac{1}{2} \left[ \phi_{i_{\text{left boundary}}(t)} + \phi_{i_{\text{right boundary}}(t)} \right]. \quad (5)$$

Figure 5 shows the operational procedure of the W-DESYNC method when there are nodes 1, 2, and 3 with  $w_1 = 10$ ,  $w_2 = 15$ , and  $w_3 = 50$ .

4.2. *Application of Cucker-Smale Flocking Model to W-DESYNC for Throughput Fairness over a Routing Path.* The W-DESYNC method is designed to dynamically control the amount of resources allocated to each node by using a weight factor. Here, we explain the method to facilitate throughput fairness among the nodes on the given routing path by using

TABLE I: Comparison of Cucker-Smale flocking model and the proposed algorithm.

	Operational rule	Parameter	Flocking object
Cucker-Smale flocking	$v_i(t+1) - v_i(t) = \frac{\lambda}{N} \sum_{j=1}^N \psi( x_j(t) - x_i(t) ) (v_j(t) - v_i(t))$	Position ( $x$ ), velocity ( $v$ )	Direction, velocity
The proposed algorithm	$w_i(n+1) - w_i(n) = \frac{1}{\ N_1(i)\ } \sum_{j \in N_1, j \neq i} (w_j(n) - w_i(n))$	Link quality ( $w_i$ )	Link throughput

the Cucker-Smale flocking model. In this flocking model, each autonomous entity controls its velocity and moving direction through interaction with neighboring entities. The CS flocking model is described as follows. Suppose that there are  $N$  autonomous entities. The position and velocity vector of the  $i$ th entity in  $\mathbb{R}^3$  at time  $t \in \mathbb{N}$  are denoted as  $x_i(t)$  and  $v_i(t)$ , respectively. The CS flocking model is given as

$$\frac{dx_i}{dt}(t) = v_i(t), \quad (6)$$

$$v_i(t+1) - v_i(t) = \frac{\lambda}{N} \sum_{j=1}^N \psi(|x_j(t) - x_i(t)|) (v_j(t) - v_i(t)), \quad (7)$$

for  $i = 0, \dots, N$ , and  $t > 0$ , where  $\lambda$  and  $\psi(\cdot)$  are the coupling strength and the function denoting the communication range that quantifies the way the entities influence each other, respectively. The communication-range function  $\psi(\cdot)$  is a nonnegative function denoting the distance between entities [19], and some possible  $\psi(\cdot)$ s are given by

$$\begin{aligned} \psi_1(|x_j - x_i|) &= 1, \\ \psi_2(|x_j - x_i|) &= 1_{|x_j - x_i| \leq r}, \\ \psi_3(|x_j - x_i|) &= \frac{1}{(1 + |x_j - x_i|^2)^\beta}, \end{aligned} \quad (8)$$

where  $r$  and  $\beta$  have positive values. According to (7), each entity adds a weighted average for the differences between the entity and its neighboring entities each time. By executing (7) repeatedly in a distributed manner, the collective behavior of each entity, that is, flocking, is obtained. Time-asymptotic flocking phenomena are expressed to satisfy the following equations:

$$\begin{aligned} \lim_{t \rightarrow \infty} |v_i(t) - v_j(t)| &= 0 \quad \text{for } i \neq j, \\ \sup_{0 \leq t < \infty} |x_i(t) - x_j(t)| &< \infty \quad \text{for } i \neq j. \end{aligned} \quad (9)$$

In [19], Cucker and Smale showed that if the communication has long-range interaction, global unconditional flocking occurs, which means that the velocities of all agents converge to the same asymptotic velocity.

For the purpose of making the weight factors of all nodes on the routing path the same, we employ this Cucker-Smale flocking model. Using this model, we could make the

weight factor of the link the same across a routing path and thus obtain the throughput fairness over a routing path. The detailed procedure is described as follows.

Let  $BW_i(n)$ ,  $\text{SNR}_{i,j}(n)$ , and  $d_{i,j}(n)$  be the bandwidth of node  $i$  allocated at frame  $n$ , the received SNR at node  $i$  from node  $j$  at frame  $n$ , and the throughput of node  $i$  at frame  $n$ , respectively. Then we have

$$d_{i,j}(n) = BW_i(n) \log_2(1 + \text{SNR}_{i,j}(n)). \quad (10)$$

Here, the SNR information of a node is shared with its one-hop neighbor nodes of the routing path (1) by using control messages such as RREQs and RREP's occurring in the routing path setup process, (2) by piggybacking data transmission, or (3) by overhearing the messages sent by the one-hop neighbor nodes. We use  $d_{i,i+1}(n)$  as the weight factor of node  $i$  at frame  $n$  for throughput fairness; that is,  $w_i(n) = d_{i,i+1}(n)$ . Thus, the throughput fairness of the nodes in a routing path is solved by synchronizing  $w_i(n)$  for all  $i$ .

Let  $N_1(i)$  and  $w_i(n)$  be the set of one-hop neighbor nodes of node  $i$  that are located in the set of one-hop neighbor nodes of node  $i$  and the weight factor of node  $i$  at frame  $n$ , respectively. Then, we apply the C-S flocking model to the weight factor of node  $i$  by updating the weight factor of node  $i$  as follows:

$$\begin{aligned} w_i(n+1) - w_i(n) &= \frac{1}{\|N_1(i)\|} \sum_{j \in N_1, j \neq i} (w_j(n) - w_i(n)), \end{aligned} \quad (11)$$

where  $\|\cdot\|$  represents the cardinality of the set. The iterative and distributive execution of (11) results in convergence of  $w_i$ .

Table 1 compares original Cucker-Smale flocking model and the proposed algorithm with respect to its operational rule, parameter, and flocking object. It is noted that the main target to synchronize in the Cucker-Smale flocking model is the directions and velocities of flying birds (or autonomous objects) while that is link throughput of a routing path (expressed by the weight factor) in the proposed algorithm.

**4.3. Application to Throughput Fairness over Multiple Routing Paths.** Application of the CS flocking model to the weight factor of a node on a given routing path yields the throughput fairness of the routing path. However, some nodes may belong to more than one path, for example, node 2 shown in Figure 6. Because (11) is assumed to be applied to a node on a single routing path, the resource allocation procedure for a node belonging to more than two routing paths must be defined. For this purpose, we simply assume that a node

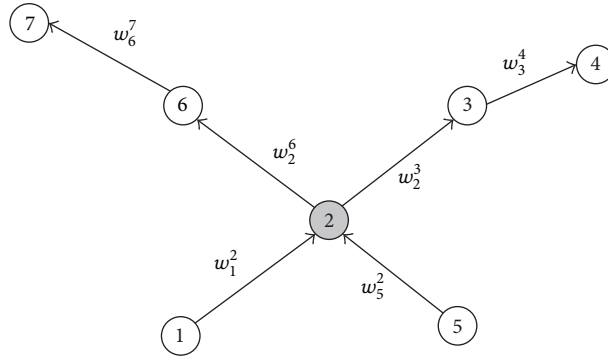


FIGURE 6: Node 2 on the multiple routing paths.

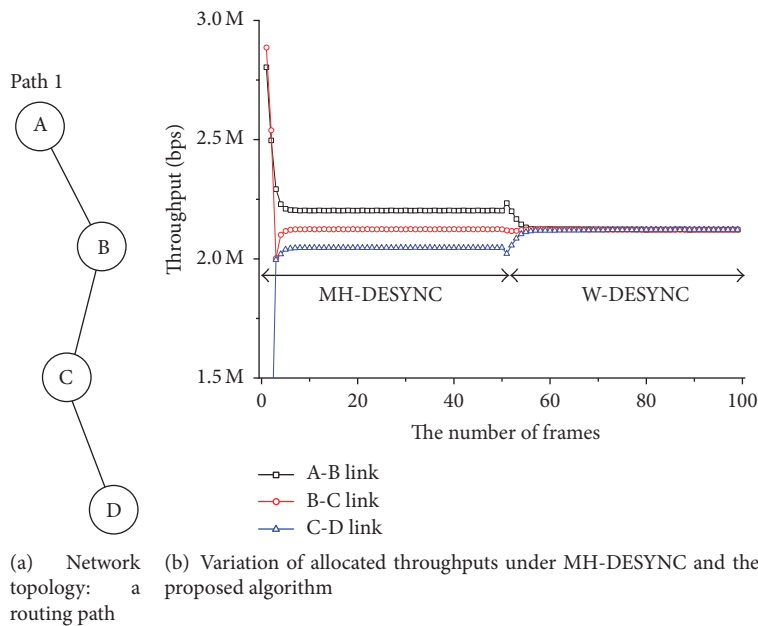


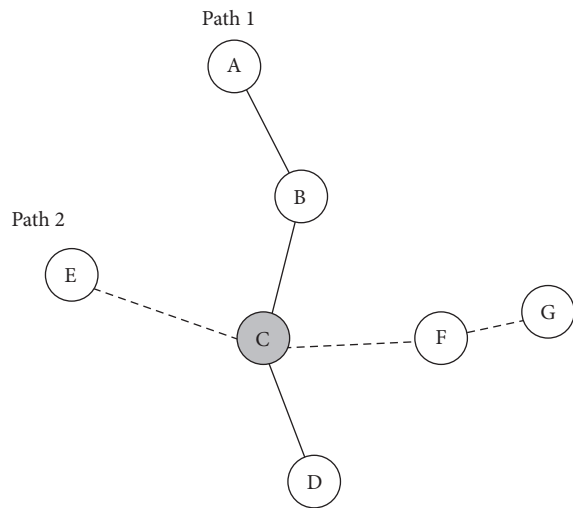
FIGURE 7: Variation of available bandwidth for a routing path.

on each of the  $k$  different routing paths executes firing  $k$  times in a single frame. Thus, the resources of a node on multiple routing paths are exclusively allocated as many times as the number of routing paths that the node belongs to, in a single frame. We therefore allocate the resources of the node on multiple routing paths to be used for each routing path. Let  $w_i^j(n)$  be the weight factor of the link from node  $i$  to node  $j$ . In Figure 6, node 2 is on two routing paths (path 1-2-3-4 and path 5-2-6-7) and performs firing twice so as to obtain the bandwidth for the routing paths of 1-2-3-4 and 5-2-6-7 by using its two weight factors  $w_2^3(n)$  and  $w_2^6(n)$ , respectively. In Table 1, the concepts of C-S flocking model and W-DESYNC are mapped. Cucker-Smale flocking model synchronizes direction and velocity from position and velocity. W-DESYNC uses Cucker-Smale flocking model to equalize the throughput from the weight factor.

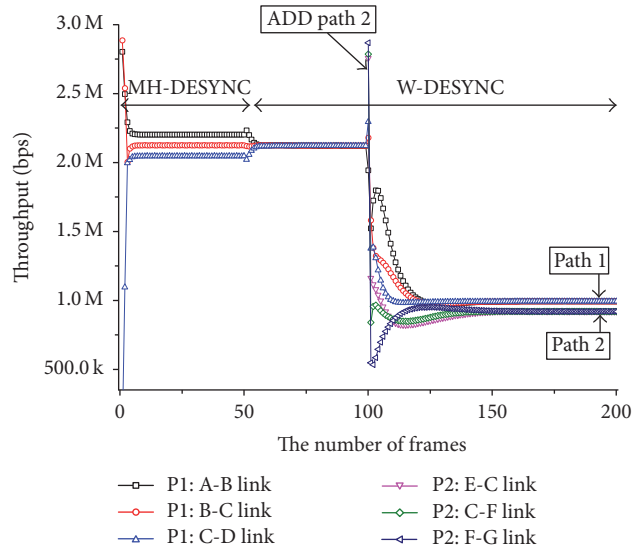
### 5. Performance Evaluation

In this section, we evaluate the performance of the proposed W-DESYNC method compared with the MH-DESYNC method. Here, it is noticed that the performance of the DESYNC method is not considered, because it was proposed for the use in fully connected networks, so it cannot be applied to the multihop environment unlike to the W-DESYNC and MH-DESYNC methods.

*5.1. Application of the Proposed Algorithm to Single Routing Path.* Figure 7(a) shows the routing path that consists of nodes A, B, C, and D in a wireless multihop network. The available bandwidth of each node is determined by using MH-DESYNC when the frame number is up to 50 frames and after that, the proposed algorithm using W-DESYNC is applied to the same routing path. Figure 7(b) shows that node



(a) Network topology: two routing paths



(b) Variation of allocated throughputs under MH-DESYNC and the proposed algorithm

FIGURE 8: Variation of available bandwidth for two routing paths.

C is allocated the lowest bandwidth when MH-DESYNC is used for the path A-B-C-D. The end-to-end throughput of the path A-B-C-D is therefore limited by the bandwidth of the link of bottleneck node C. On the other hand, we can verify that the amount of bandwidths allocated to each link converges to the same when the proposed algorithm is applied, which shows that the bottleneck-node problem is resolved with the use of the proposed algorithm.

**5.2. Application of the Proposed Algorithm to Multiple Routing Paths.** Figure 8(a) shows the network topology with two routing paths; specifically the new path E-C-F-G is added to the path A-B-C-D in Figure 7(a). In this figure, node C becomes a bottleneck node where two paths share. Figure 8(b) shows the variation of available bandwidth when the new path E-C-F-G is added at the frame number of 100. The result shows that the link throughputs in each routing path converge to the same value although the converged link throughput of each routing path is different. From the result, we can confirm that the proposed algorithm can prevent nodes from being bottleneck nodes that decrease end-to-end throughput of the routing path even when some nodes are shared by multiple routing paths.

**5.3. Network Level Simulation.** Finally, we measure and compare the performances of the proposed W-DESYNC method and the MH-DESYNC method in an environment where 100 nodes are randomly deployed and the network size is  $500 \times 500 \text{ m}^2$ , as shown in Figure 9(a). Source and destination nodes are arbitrarily chosen, and the routing paths are set up through ad hoc on-demand distance-vector routing (AODV). Simulation parameters used in the simulation runs are summarized in Table 2.

TABLE 2: Simulation parameters.

Parameter	Value
The number of nodes	100
Routing protocol	AODV
System bandwidth	11 Mbps
Transmission range	100 m
Network size	500 m by 500 m
The number of control slots	40
Control slot size	0.0747 ms
The number of data slots	10,000
Data slot size	0.8988 $\mu\text{s}$
Frame size	11.235 ms
Packet size	324 bytes (including MAC header of 20 bytes and IP header of 24 bytes)
Packet generation interval	20 ms

Figure 9(b) shows the average end-to-end bandwidth of a routing path as a function of the number of routing paths while adding a routing path one by one. The results show that as the number of paths increases, more nodes share the limited resource among neighbors, and thus the average end-to-end bandwidths of a routing path decrease under both methods. However, the result shows that the proposed W-DESYNC method achieves higher end-to-end bandwidth than the MH-DESYNC method. Figure 9(c) shows the total end-to-end throughput as a function of the number of routing paths in the network when the proposed and MH-DESYNC methods are used. Here, routing paths are added one by one in the network, similar to the case of Figure 9(b), and packets are generated every 20 ms at the source node and transmitted via the given routing path to the destination

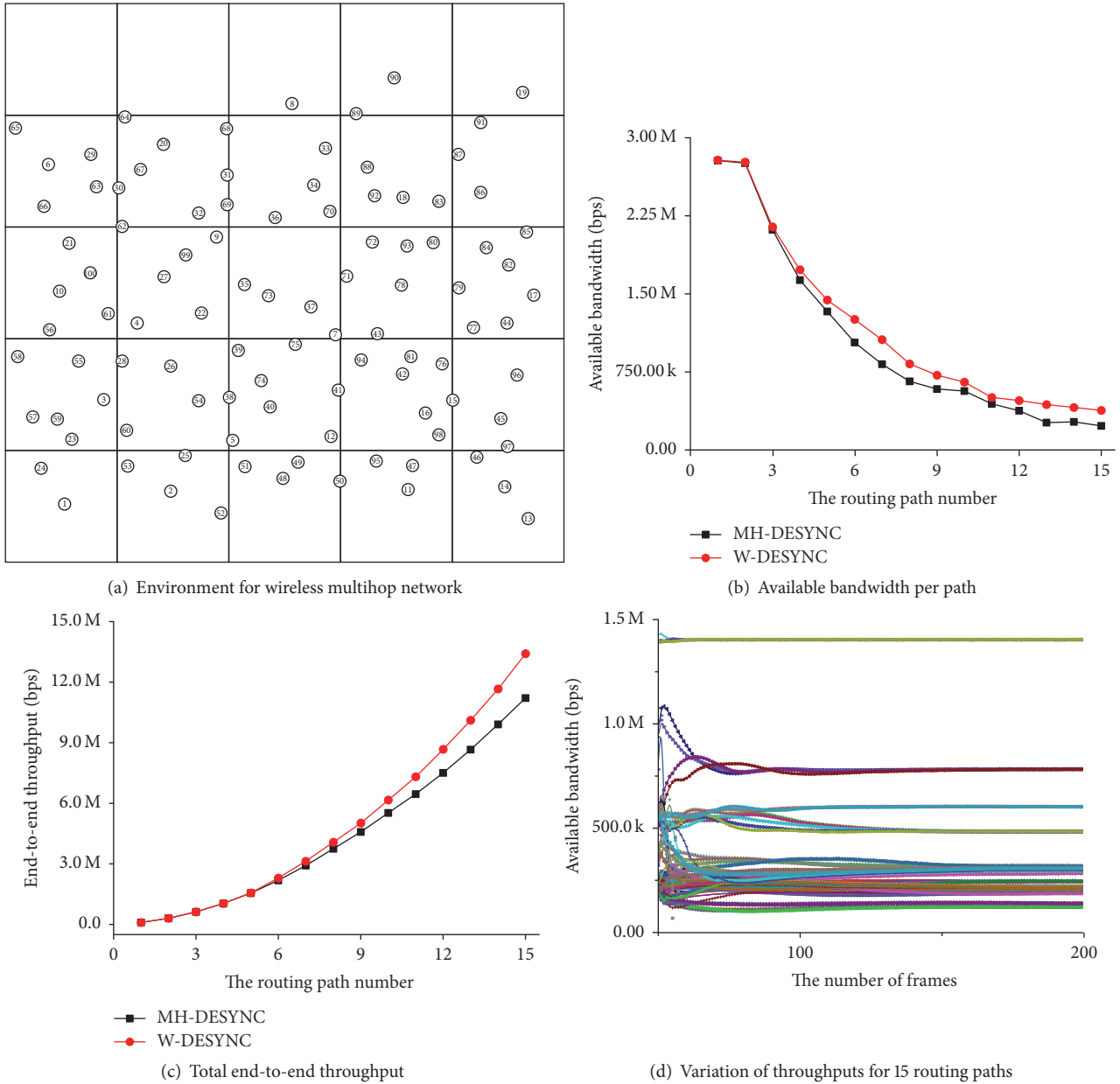


FIGURE 9: Wireless multihop performance.

node. According to Figure 9(c), the throughput fairness among all the nodes in the routing path via the W-DESYNC method enhances the total end-to-end throughput in the network. Figure 9(d) shows the variation of the throughput of each routing path when there are 15 routing paths in the network. The end-to-end throughput of the routing paths converges over time, indicating the stability of the proposed method.

### 6. Conclusion

In this method, we proposed an algorithm for resolving the performance degradation caused by the bottleneck node in a routing path. The W-DESYNC method is proposed as a technique to enable nodes to be allocated resources according

to their weight factors. To obtain throughput fairness among the links on a routing path, a bioinspired flocking model is applied to the updating procedure of the weight factor, which is derived from the link-quality information. In addition, the multiple-firing procedures for facilitating multiple resource allocation for a node on multiple routing paths are introduced. Simulation results show that the proposed method achieves the throughput fairness of the routing paths and increases the network throughput by efficiently solving the bottleneck-node problem in multihop networks.

### Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.



## Acknowledgments

This research was supported by Agency for Defense Development (ADD-IBR-245). This research was supported by the Chung-Ang University Research Scholarship grants in 2015.

## References

- [1] S. Abdel Hamid, H. S. Hassanein, and G. Takahara, *Routing for Wireless Multi-Hop Network*, Springer, 2013.
- [2] B. Melander, M. Bjorkman, and P. Gunningberg, "New end-to-end probing and analysis method for estimating bandwidth bottlenecks," in *Proceedings of the IEEE Global Telecommunication Conference (GLOBECOM '00)*, vol. 1, pp. 415–420, November 2000.
- [3] M.-F. Tsai, N. Chilamkurti, J. H. Park, and C.-K. Shieh, "Multipath transmission control scheme combining bandwidth aggregation and packet scheduling for real-time streaming in multipath environment," *IET Communications*, vol. 4, no. 8, pp. 937–945, 2010.
- [4] W.-Y. Shin, S.-Y. Chung, and Y. H. Lee, "Parallel opportunistic routing in wireless networks," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6290–6300, 2013.
- [5] A. Orda and A. Sprintson, "Precomputation schemes for QoS routing," *IEEE/ACM Transactions on Networking*, vol. 11, no. 4, pp. 578–591, 2003.
- [6] R. Hou, K.-S. Lui, F. Baker, and J. Li, "Hop-by-hop routing in wireless mesh networks with bandwidth guarantees," *IEEE Transactions on Mobile Computing*, vol. 11, no. 2, pp. 264–277, 2012.
- [7] X. Jia, D. Li, and D. Du, "QoS topology control in ad hoc wireless networks," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '04)*, vol. 2, pp. 1264–1272, March 2004.
- [8] A. V. Babu and L. Jacob, "Fairness analysis of IEEE 802.11 multi-rate wireless LANs," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 5, pp. 3073–3088, 2007.
- [9] K. Ronasi, V. W. S. Wong, and S. Gopalakrishnan, "Distributed scheduling in multihop wireless networks with maxmin fairness provisioning," *IEEE Transactions on Wireless Communications*, vol. 11, no. 5, pp. 1753–1763, 2012.
- [10] F. Yu, T. Wu, and S. Biswas, "Toward in-band self-organization in energy-efficient MAC protocols for sensor networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 2, pp. 156–170, 2008.
- [11] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1452–1463, 2006.
- [12] S. Tang, "Distributed multiuser scheduling for improving throughput of wireless LAN," *IEEE Transactions on Wireless Communications*, vol. 13, no. 5, pp. 2770–2781, 2014.
- [13] J. Alonso-Zarate, C. Verikoukis, E. Kartsakli, A. Cateura, and L. Alonso, "A near-optimum cross-layered distributed queuing protocol for wireless LAN," *IEEE Wireless Communications*, vol. 15, no. 1, pp. 48–55, 2008.
- [14] X. Wang and K. Kar, "Cross-layer rate optimization for proportional fairness in multihop wireless networks with random access," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1548–1559, 2006.
- [15] J. Degeysys, I. Rose, A. Patel, and R. Nagpal, "DESYNC: self-organizing desynchronization and TDMA on wireless sensor networks," in *Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN '07)*, pp. 11–20, April 2007.
- [16] J. Degeysys, I. Rose, A. Patel, and R. Nagpal, "Self-organizing desynchronization and TDMA on wireless sensor networks," in *Proceedings of the 1st Workshop on Bio-Inspired Design of Networks (BIOWIRE '07)*, pp. 192–203, Cambridge, UK, April 2007.
- [17] A. Patel, J. Degeysys, and R. Nagpal, "Desynchronization: the theory of self-organizing algorithms for round-robin scheduling," in *Proceedings of the 1st International Conference on Self-Adaptive and Self-Organizing Systems (SASO '07)*, pp. 87–96, July 2007.
- [18] Y.-J. Kim, H.-H. Choi, and J.-R. Lee, "A bioinspired fair resource-allocation algorithm for TDMA-based distributed sensor networks for IoT," *International Journal of Distributed Sensor Networks*, vol. 2016, Article ID 7296359, 10 pages, 2016.
- [19] F. Cucker and S. Smale, "Emergent behavior in flocks," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 852–862, 2007.

## Research Article

# A Collusion-Resistant and Privacy-Preserving Data Aggregation Protocol in Crowdsensing System

**Chang Xu, Xiaodong Shen, Liehuang Zhu, and Yan Zhang**

*Beijing Engineering Research Center of Massive Language Information Processing and Cloud Computing Application, School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China*

Correspondence should be addressed to Liehuang Zhu; [liehuangz@bit.edu.cn](mailto:liehuangz@bit.edu.cn)

Received 8 December 2016; Accepted 15 March 2017; Published 30 March 2017

Academic Editor: Rossana M. C. Andrade

Copyright © 2017 Chang Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the pervasiveness and increasing capability of smart devices, mobile crowdsensing has been applied in more and more practical scenarios and provides a more convenient solution with low costs for existing problems. In this paper, we consider an untrusted aggregator collecting a group of users' data, in which personal private information may be contained. Most previous work either focuses on computing particular functions based on the sensing data or ignores the collusion attack between users and the aggregator. We design a new protocol to help the aggregator collect all the users' raw data while resisting collusion attacks. Specifically, the bitwise XOR homomorphic functions and aggregate signature are explored, and a novel key system is designed to achieve collusion resistance. In our system, only the aggregator can decrypt the ciphertext. Theoretical analysis shows that our protocol can capture  $k$ -source anonymity. In addition, extensive experiments are conducted to demonstrate the feasibility and efficiency of our algorithms.

## 1. Introduction

Recently, smart devices and wireless network have a rapid development. Smart devices, such as smart phone, pad, and smart watch, have become ubiquitous all over the world. They have not only strong and independent computational capability but also rich embedded sensors. The advance of wireless communication technology further makes them connect more tightly, which can be leveraged to develop more applications. People can use the rich embedded sensors to collect different kinds of data, including pictures, sounds, and videos. The strong ability and lower cost of this system derive a new popular paradigm named crowdsensing.

In a typical crowdsensing application, the server, or the aggregator, recruits a group of users to work for him. Having been informed about their sensing work by the aggregator, all the users use their devices to collect data with relevant sensors and upload them to the server through Wi-Fi or 3G/4G network. Recently, a myriad of crowdsensing applications have been developed in different areas such as transportation [1], environment monitoring [2], healthcare [3], and social

network [4]. In this paper, we consider that an aggregator wants to periodically collect data and computes some functions based on them to obtain the desired information. For example, to monitor the health situation of a particular district, the aggregator recruits some users in this district to collect their body temperature or blood oxygen. The users need to contribute their data each hour by sensing relative data and uploading them.

However, most similar applications require users to upload their private data, which may breach individual's security and privacy. Concerned about these threats, users tend to refuse to participate in crowdsensing. Therefore, the user's security and privacy should be protected in a crowdsensing system. Lots of previous works [5–11] have focused on the challenge. Specifically, [5] allows the server to evaluate any multivariate polynomials. However, users need to communicate to generate their encryption key. In [7], the aggregator can only acquire the summation of all the raw data. Both of them did not consider the collusion between users and the aggregator. And [6] gives a solution by using more complex encryption keys. Unfortunately, the

protocol requires  $k + 1$  rounds of key exchange when  $k$  colluding adversaries exist, and [8, 9] focus on multimedia data collection and require data interchange. Li et al. [10] proposed a novel key system to resist collusion attack, but it only supports sum and min aggregation. Zhang et al. [11] first proposed a scheme where the aggregator can acquire all the raw data; thus different functions can be computed in one round. Each user's privacy is protected by delinking data from its source. Thus, the only information the aggregator knows is that a particular data belongs to one of  $k$  users, which is called  $k$ -source anonymity. Nevertheless, there are still some problems in [11]. Specifically, each user owns half part of another user's secret keys and cannot resist collusion attack. The aggregator does not have decryption keys. Therefore, the outside adversary can decrypt the ciphertexts to get all the data if it can eavesdrop all the ciphertexts, which violates the aggregator's benefits.

In this paper, we propose a novel protocol to not only support different aggregation functions but also achieve collusion resistance. In each time period, we use the timestamp as a public parameter. The bitwise XOR homomorphic function is executed in the encryption phase. All the users take their encryption keys and the timestamp as the parameters of encryption functions to generate pseudorandom bit strings as a one-time pad to encrypt the raw data. To prevent the collusion attack, a novel encryption algorithm is designed. The aggregate signature is also taken to protect data integrity and achieve identity authentication.

Our contributions contain three parts:

- (i) We propose a novel protocol to protect users' privacy and resist collusion attack when the aggregator can obtain all the users' raw data and compute any functions based on them. We assume that the aggregator and a fraction of users are not reliable and may collude with each other, and they still cannot obtain any valuable information.
- (ii) We protect the aggregator's benefits by preventing outside adversary from decrypting the ciphertexts. All the ciphertexts are also guarded against being tampered by applying aggregate signature. If any abnormal data is found, the aggregator can require the trust party to get involved according to the signature.
- (iii) We prove that our protocol can achieve  $k$ -source anonymity. Theoretical analysis shows the computational cost of our protocol. In addition, extensive experiments are also conducted to demonstrate that the protocol can be executed efficiently.

The remainder of this paper is organized as follows. Section 2 discusses related work. In Section 3, we present our system model, security model, and design goals. After the introduction of preliminary knowledge in Section 4, we elaborate our aggregation scheme in Section 5 and prove the security of the protocol in Section 6. Section 7 shows our experiment result. Finally, we conclude our paper in Section 8.

## 2. Related Work

The data aggregation issues are first discussed in wireless sensor networks (WSN) [12–16]. Although there are many differences between WSN and crowdsensing, the work about WSN still gives inspirations to solve problems in crowdsensing. Works in [17, 18] consider the user recruitment and incentive in crowdsensing. The papers [19–21] assume a trust server although they are devoted to protect users' privacy. References [5–9, 22] contribute to overcoming the challenge when the aggregator is untrusted. However, bidirectional communication between users is required in these schemes, which is a strong assumption in crowdsensing system. Jung et al. [5] first proposed their product protocol and sum protocol and combined them to evaluate any multivariate polynomials. In the product protocol, each user interchanges his/her public parameter with his/her left and right user while all the users are arranged in a circle. With two public parameters and the secret key, the user can derive a pseudorandom number to execute encryption operations. In the sum protocol, they use modular property to compute summation efficiently. However, the pseudorandom number can only be used once and may breach the privacy if being used in several rounds. Therefore, in each round the setup phase should be executed where users have to communicate with the other two. The collusion between users is also a security issue. Jung et al. [6] tried to solve the problem by using more complex ways to generate pseudorandom number, correspondingly more rounds of key interchange are needed, and Jung et al.'s scheme only supports particular aggregation functions.

In [10, 23], the privacy-preserving aggregation protocol is proposed while communication within users is not required. Zhang et al. [23] allowed the aggregator to obtain the minimum or the  $k$ th minimum value of all the data without knowing them, and they assumed a semihonest aggregator and only supported the single aggregation function. Li et al. [10] proposed a scheme to resist collusion by using a novel key management system. The key dealer generates a key set which contains hundreds or thousands of elements. Then, the set is divided and distributed to the users and aggregator, and each of them owns multiple secret keys. The ability to resist collusion attack relies on the adversary to guess an honest user's keys correctly from a big set. However, this scheme can only support sum and min aggregation.

Different from the schemes which protect privacy by hiding content, Zhang et al. [11] delink all the users' data with source, through which the aggregator can collect all the raw data while it remains unaware of their corresponding owners. Thus, the aggregator can compute any complex aggregation functions on them. Each user has two keys to encrypt data, and he/she shares each of them with the other two users. Therefore, the aggregator can decrypt all the ciphertexts without decryption keys when the bitwise XOR homomorphic functions are employed. However, this paper ignores the collusion attack and cannot protect the aggregator's benefits, because an honest user's secret key can easily be recovered and the aggregator has the same ability with outside adversary without decryption keys.

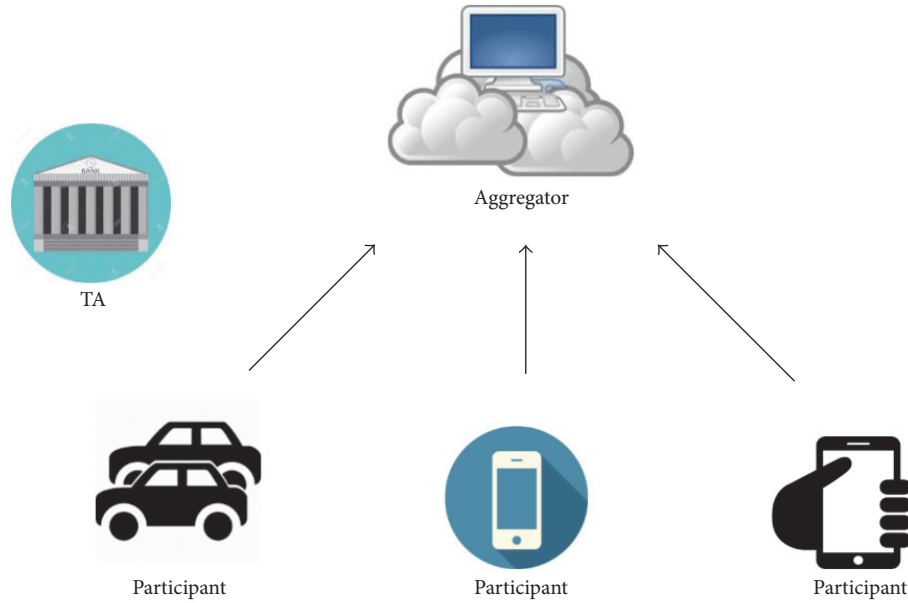


FIGURE 1: System model of data aggregation.

In this paper, we propose a novel protocol to solve the issues which are not dealt with in [11], while protecting the data integrity and achieving authentication and traceability.

### 3. Models and Design Goal

**3.1. System Model.** Our system is comprised of three parties:  $N$  participants, an aggregator, and a trust authority (TA). Assuming that there are  $N$  participants in this system who want to contribute data to the aggregator and get corresponding reward from the aggregator, the aggregator is willing to collect data and compute some functions on these data including addition and production aggregation. Only one-way communication channel is needed from participants to the aggregator, which could be 3G/4G, Wi-Fi, or other kind of channels supported by our system parties. We show our system model in Figure 1 and describe the details as follows:

- (i) *TA.* The TA is responsible for initializing the whole system, which includes registering the aggregator and participants, generating and distributing keys, and revealing and revoking the malicious participants. Once the system initialization phase is finished, the TA is off-line in all the phases except for the occurrence of abnormal behavior.
- (ii) *Participant.* The participants may be mobile users who hold smartphones with various sensors or vehicles with built-in sensors. They wish to sense data and upload them to the aggregator periodically to get reward. Assuming that there are  $n$  participants in our system, which can be numbered as  $u_1, u_2, \dots, u_n$ , they collaborate to push data to the aggregator in each time period, for instance, fifteen minutes per

time, which can be listed as  $t_1, t_2, \dots, t_n$ . Peer-to-peer communication is not required among participants. In the remainder of this paper we will interchangeably use the same meaning for the user and the participant.

- (iii) *Aggregator.* The aggregator periodically collects the participants' data and uses them to compute arbitrary aggregation functions. The aggregation result can be leveraged to get commercial benefits. The data can be time series data, location based data, or any other kind of predefined numerical data.

**3.2. Security Model.** Because the collected data may include users' sensitive information, we mainly focus on the participants' privacy in our security model. Any adversary should not link data with the real data owner, such that the users' privacy will not be compromised, even if any internal adversaries, namely, the malicious users and the aggregator, collude to snoop into the privacy. Meanwhile, if the abnormal data is detected, the TA is involved to reveal the malicious users' real identities and revoke them from the system.

- (i) *TA.* In our system, we assume the TA is fully trusted and cannot be compromised. The communication channel between the TA and participants is secure or can be protected by cryptographic tools.
- (ii) *Participant.* The participants honestly execute the protocol, but they are curious about other participants' data. This assumption is based on the fact that some users can leverage others' valid data to receive reward instead of collecting data by their own, which may consume computation resource, battery, and other resources. A fraction of malicious users may also collude to recover valid users' secret keys,

which compromises users' privacy. Another abnormal behavior is data pollution; that is, some users deliberately upload incorrect data to the aggregator, leading to wrong aggregation result. Many previous studies have focused on this issue but cannot solve the problem perfectly. Thus, in this paper, we provide traceability to identify the malicious users when the abnormal data is found in each round.

- (iii) *Aggregator*. The aggregator is curious but honest. The aggregator can collect all the users' data and has more abilities to breach users' privacy. The untrusted aggregator can also collude with some malicious users to recover users' secret keys and thus link the data with its owner. Other parties may eavesdrop all the users' data to compute the aggregation, thus causing the aggregator's monetary loss.

**3.3. Design Goals.** Under our system model, our design goal is to develop a framework to hold the security properties. We not only protect the privacy of users but also resist against collusion attack launched by internal parties and other attacks such as message tampering launched by outside adversaries. Specifically, the following desirable goals should be achieved.

- (i) *Protecting Participants' Privacy*. When the participants upload their sensing data to the aggregator, we should guarantee not only that outside adversary cannot eavesdrop and tamper the original data but also that other users cannot decrypt the ciphertext. Any tampered data can be recognized by the aggregator and a retransmission request is sent to the user. Any illegal party cannot forge a legal user's signature. The user's data should not be linked with its source by any party including the aggregator.
- (ii) *Safeguarding the Aggregator's Benefits*. We assume any party including outside adversary has the ability to eavesdrop all the uploaded data, and if other people can recover the original data, they can compute any aggregation result and thus seriously damage the aggregator's benefits. Therefore, we design the protocol to prevent illegal parties from getting valuable data.
- (iii) *Computation Efficiency and Accuracy*. The proposed framework should achieve computation efficiency and accuracy; in particular, (i) the users can efficiently encrypt the data and compute the corresponding signature on it; (ii) the aggregator can efficiently verify all the users' signatures and recover the original data accurately; (iii) with all the original data the aggregator can accurately compute any function on them with high efficiency.

## 4. Preliminary

**4.1.  $k$ -Source Anonymity.** Assuming that there is a group of  $k$  users, each user uploads his/her data to the aggregator. Although the aggregator can obtain the exact values of all

data, it still cannot link each data with its owner, because every user's data is hidden in the dataset of  $k$  elements. The more users the group contains, the higher security level the system achieves. For a specific user, if the adversary can only know that the user's data belongs to one of the  $k$  users, we say this user's data holds  $k$ -source anonymity. If all data captures  $k$ -source anonymity in a data aggregation protocol, we say this protocol achieves  $k$ -source anonymity. Intuitively, the definition states if the aggregator cannot efficiently notice whether we switch two data items, the aggregation protocol with  $k$  users is  $k$ -source anonymous.

*$k$ -Source Anonymous.* A data aggregation process or a protocol is  $k$ -source anonymous if  $P(\dots, u_i(d^i), \dots, u_j(d^j), \dots) \stackrel{c}{\equiv} P(\dots, u_i(d^j), \dots, u_j(d^i), \dots)$  is satisfied, where  $k$  denotes the number of users for any group  $U$ ,  $u_i$  and  $u_j$  are two users in  $U$ ,  $\{d^1, \dots, d^k\} \in \{M\}^k$  is data aggregation sample,  $M$  denotes the message space,  $P(\dots, u_i(x_i), \dots)$  denotes the data aggregator's view when running protocol with  $x_i$  ( $x_i \in \{d^1, \dots, d^k\}$ ) as  $u_i$ 's input ( $i = 1, 2, \dots, k$ ), and  $\stackrel{c}{\equiv}$  denotes computational indistinguishability of two random variable ensembles.

Our goal is to design efficient data aggregation protocols that can be used by an untrusted aggregator to collect all users data in a source-anonymous manner.

**4.2. Bilinear Map and Aggregate Signature.** Let  $G_1$  and  $G_2$  be two cyclic groups of prime order  $q$ , and their generators are  $g_1$  and  $g_2$ , respectively. There exists an additional group  $G_T$  such that  $|G_1| = |G_2| = |G_T|$ . A bilinear map is a map  $\hat{e} : G_1 \times G_2 \rightarrow G_T$  with the following properties.

(1) *Bilinear.* For all  $u \in G_1$ ,  $v \in G_2$ , and  $a, b \in \mathbb{Z}$ ,  $\hat{e}(u^a, v^b) = \hat{e}(u, v)^{ab}$ .

(2) *Nondegenerate.*  $\hat{e}(g_1, g_2) \neq 1$ . The aggregate signature scheme [24] employs a full-domain hash function  $H : \{0, 1\}^* \rightarrow G_1$  and comprises the following five phases.

*Key Generation.* For a particular user, pick random  $x \xleftarrow{R} Z_p$ , and compute  $v \leftarrow g_2^x$ . The user's public key is  $v \in G_2$ . The user's secret key is  $x \in Z_p$ .

*Signing.* For a particular user, given the secret key  $x$  and a message  $M \in \{0, 1\}^*$ , compute  $h \leftarrow H(M)$ , where  $h \in G_1$ , and  $\sigma \leftarrow h^x$ . The signature is  $\sigma \in G_1$ .

*Verification.* Given a user's public key  $v$ , a message  $M$ , and a signature  $\sigma$ , compute  $h \leftarrow H(M)$ ; accept the signature if  $\hat{e}(\sigma, g_2) = \hat{e}(h, v)$  holds.

*Aggregation.* For the aggregating subset of users  $U \subseteq \mathbb{U}$ , an index  $i$  is assigned to each user, where  $i \in [1, k]$  and  $k = |U|$ . Each user  $u_i \in U$  provides a signature  $\sigma_i \in G_1$  on a message  $M_i \in \{0, 1\}^*$  of his/her choice. The messages  $M_i$  must all be distinct. Compute  $\sigma \leftarrow \prod_{i=1}^k \sigma_i$ . The aggregate signature is  $\sigma \in G_1$ .

*Aggregate Verification.* Given an aggregate signature  $\sigma \in G_1$  for an aggregating subset of users  $U$ , indexed as before, and the original messages  $M_i \in \{0, 1\}^*$  and public keys  $v_i \in G_2$  for all users  $u_i \in U$ , to verify the aggregate signature  $\sigma$ ,

(1) ensure that the messages  $M_i$  are all distinct, and reject them otherwise;

(2) compute  $h_i \leftarrow H(M_i)$  for  $1 \leq i \leq k = |U|$ , and the aggregate signature is valid if  $\tilde{e}(\sigma, g_2) = \sum_{i=1}^k \tilde{e}(h_i, v_i)$  is satisfied.

## 5. The Collusion-Resistant and Privacy-Preserving Data Aggregation Protocol

In this section, we present our privacy-preserving aggregation scheme to achieve the aforementioned design goals.

*5.1. Overview.* Our proposed scheme can achieve  $k$ -source anonymity while it prevents adversary from tampering users' uploaded messages and generates invalid signatures. During the system initialization phase, the TA generates users' public/secret key for message encryption and authentication and the aggregator's secret key for decryption. When the aggregator wants to collect some data from  $n$  users, it first confirms the time period  $t$  with other users. All the users sense the data and encrypt it with their own secret keys and sign the encrypted data and then collectively upload all the data to the aggregator. If any user does not have data to send, he/she can upload a predefined value, which helps others to decrypt data. After all the data are collected, the aggregator first aggregates all users' signatures and verifies them. If signatures are valid, all the users' original data can be recovered with the aggregator's secret key but cannot be linked with their owners' real identities.

Our scheme consists of three algorithms: *System Initialization* algorithm assigns keys to the users and the aggregator. *Enc&Sign* algorithm encrypts users' data and signs the ciphertext. *Verify&Dec* algorithm verifies and decrypts users' data.

We state the basic idea of the encryption and decryption here. Consider the following equation:

$$s_1 \oplus s_2 \oplus \cdots \oplus s_n = s_1 \oplus s_2 \oplus \cdots \oplus s_n. \quad (1)$$

Then we use  $s_k$  ( $k \in [1, n]$ ) as the key of the pseudorandom hash function  $h$ ; thus

$$\begin{aligned} h_{s_1}(t) \oplus h_{s_2}(t) \oplus \cdots \oplus h_{s_n}(t) \\ = h_{s_1}(t) \oplus h_{s_2}(t) \oplus \cdots \oplus h_{s_n}(t). \end{aligned} \quad (2)$$

We assign the left part of (1) to all users and the right part to the aggregator. They use the same  $t$  as the parameter of  $h$  and  $h_{s_k}(t)$  as the pad to encrypt or decrypt the data, and thus the aggregator can eliminate all the pads with its keys.

However, although all users can collectively compute  $n$  hash functions, the aggregator has to compute  $n$  hash functions every time by himself. Therefore, we move some elements in the right part of (1) to the left:

$$\begin{aligned} s_1 \oplus s_2 \oplus \cdots \oplus s_n \oplus s_1 \oplus s_2 \oplus \cdots \oplus s_{n-q} \\ = s_{n-q+1} \oplus \cdots \oplus s_n. \end{aligned} \quad (3)$$

Thus the aggregator knows fewer secret keys and computes much fewer hash functions, and each user only needs to compute a few more functions. The notations in our scheme are listed in the Notations.

*5.2. System Initialization.* Given the security parameter  $\lambda$ , the TA generates the bilinear parameters  $(q, G_1, G_2, G_T, \tilde{e}, g_1, g_2)$  and chooses two hash functions:  $H : \{0, 1\}^* \rightarrow G_1$  and  $h_s(x)$  which is a function indexed by  $s$  in a pseudorandom function family  $\mathcal{H}_{l, m + \lceil \log_2 n \rceil, l} = \{h_s : \{0, 1\}^{m + \lceil \log_2 n \rceil} \rightarrow \{0, 1\}^l\}_{s \in \{0, 1\}^l}$ .

Then the TA randomly generates  $s_1, s_2, \dots, s_{nc} \in \{0, 1\}^l$  as secret keys, where  $n$  is the number of the users. As the idea described in previous subsection, we distribute these secret keys as follows:

- (i) The keys are randomly divided into  $n$  disjoint subset,  $\hat{S}_1, \hat{S}_2, \dots, \hat{S}_n$ . We use  $\hat{S}_i$  to denote the user  $i$ 's additive secret key set, where  $|\hat{S}_i| = c$ , and  $\hat{S}$  to denote the universal additive set, where  $\hat{S} = \bigcup_{i=1}^n \hat{S}_i$ .
- (ii) The TA randomly chooses  $q$  secret keys from  $S$  to generate a subset  $S_a$  and divides the remaining  $nc - q$  secret keys into  $n$  random disjoint parts, denoted as  $\hat{S}_i$ ,  $i = 1, 2, \dots, n$ , which is called subtractive secret key set. Among them, there are  $nc - q - n \times \lfloor (nc - q)/n \rfloor$  subtractive sets containing  $\lfloor (nc - q)/n \rfloor + 1$  keys, and the other  $n(1 + \lfloor (nc - q)/n \rfloor) - nc + q$  sets contain  $\lfloor (nc - q)/n \rfloor$  keys. We also use  $\hat{S}$  to denote the universal subtractive set, where  $\hat{S} = \bigcup_{i=1}^n \hat{S}_i$ . It is clear that  $\hat{S} = \hat{S} \cup S_a$ .
- (iii) Let  $S_i = \hat{S}_i \cup \hat{S}_i$  for  $i = 1, 2, \dots, n$ , and each  $S_i$  is sent to the user  $i$  as encryption key set. Also, all the keys in  $S_a$  are sent to the aggregator as decryption keys.

The signing key is also generated in this phase. For each user  $i$  ( $i = 1, 2, \dots, n$ ), the TA generates a pseudo ID  $Uid_i = \{0, 1\}^{\lceil \log_2 n \rceil}$  for him in each period, a secret signing key  $sk_i = x_i \stackrel{R}{\leftarrow} \mathbb{Z}$ , and a public signing key  $pk_i = g_2^{x_i}$  for the pseudo ID.

*5.3. Enc&Sign.* In each period  $t$ , before the users upload their data to the aggregator, a sequence number  $\text{seq}(i) \in [1, n]$  is generated for user  $i$ , where  $i = 1, 2, \dots, n$ .  $\{\text{seq}(i)\}_{i=1,2,\dots,n}$  is a permutation of  $\{1, 2, \dots, n\}$ , which is used to scramble the order of users' data. Each user  $i$  encrypts his/her data according to  $\text{seq}(i)$ , and the aggregator does not know the owner of the  $j$ th data after decryption, where  $j = 1, 2, \dots, n$ , because  $\{\text{seq}(i)\}_{i=1,2,\dots,n}$  is unknown for him and thus cannot get  $i$  so that  $\text{seq}(i) = j$ . Considering security issues, the sequence number should be changed randomly. We emphasize that the sequence number can be generated by the TA or through communication among the users.

For user  $i$ , although he/she only owns  $l$ -bit data, he/she has to upload  $n * l$  bits' ciphertext to hide his/her data; otherwise the connection between his/her identity and  $d^i$  can be easily found. Therefore, each user has to compute extra  $(n - 1)l$  bits' ciphertext.

**Input:**

For each user  $i$  ( $i = 1, 2, \dots, n$ ), input his/her pseudo ID  $Uid_i$ , secret encryption key set  $S_i$ , and secret signing key  $sk_i = x_i$ . Each user uploads his/her data  $d^i \in \{0, 1\}^l$  in the time period  $t$ .

The symbol  $a | b$  represents the concatenation of  $a$  and  $b$  and  $\oplus_{h_{s \in S_i}}(x)$  denotes the exclusive-or of all the results of function  $h$  for each element  $s$  in  $S_i$ .

**Output:**

The user  $i$  outputs  $e^i$  and  $\sigma^i$  as follows:

**begin**

- (1) Generate  $n$  random  $l$ -bit strings  $k_j^i$  ( $j = 1, 2, \dots, n$ ) using

$$k_j^i = \oplus_{h_{s \in S_i}}(t | j)$$

- (2) The data  $d^i$  is encrypted as:

$$e^i = (\{0\}^l \oplus k_1^i) | (\{0\}^l \oplus k_2^i) | \dots | (d^i \oplus k_{\text{seq}(i)}^i) | \dots | (\{0\}^l \oplus k_n^i)$$

- (3) The signature is computed as:

$$\sigma^i = H^{x_i}(Uid | e^i) \in G_1$$

- (4) Output  $e^i$  and  $\sigma^i$ .

**end**

ALGORITHM 1: Enc&amp;Sign.

To generate  $n * l$  bits' ciphertext, user  $i$  first uses his/her encryption keys as the secret key of  $h_s(x)$  to generate  $n * l$  bits' one-time pad,  $k_1^i, k_2^i, \dots, k_n^i$ . Notice that all  $k_j^i$  are different, because each of them is generated by different parameter  $t | j$ . To scramble the order of  $d^i$ , we encrypt  $d^i$  with  $k_{\text{seq}(i)}^i$  instead of  $k_j^i$ . Furthermore,  $l$ -bit zero string is encrypted with  $k_j^i$  ( $j \neq \text{seq}(i)$ ). Thus,  $n$  encrypted  $l$ -bit strings are obtained:  $\{0\}^l \oplus k_1^i, \{0\}^l \oplus k_2^i, \dots, d^i \oplus k_{\text{seq}(i)}^i, \dots, \{0\}^l \oplus k_n^i$ . Then all of them are concatenated one by one to generate  $e^i$ . If any user does not have data to upload, he/she can simply set his/her data to 0.

Finally, the signature  $\sigma^i$  is generated. Each user executes Algorithm 1 and then sends  $e^i$  and  $\sigma^i$  to the aggregator with his/her pseudo ID  $Uid_i$ .

**5.4. Verify&Dec.** The aggregator runs Algorithm 2 to fetch the data. After receiving all the ciphertexts from  $n$  users, the aggregator first leverages users' public keys to verify their signatures. If the algorithm outputs  $-1$ , which means some signatures are invalid, the aggregator discards the invalid data and asks for a retransmission. Otherwise all the users' original data can be recovered with the aggregator's secret key.

To decrypt all the users' data, the aggregator needs to take exclusive-OR on all the ciphertexts. Let  $e^a = e^1 \oplus e^2 \oplus \dots \oplus e^n$ . Divide  $e^a$  into  $n$  parts as  $e^a = e_1^a | e_2^a | \dots | e_n^a$ , and each part is a  $l$ -bit string. We know that  $e_{\text{seq}(i)}^a$  is the ciphertext of  $d^i$ , and:

$$\begin{aligned} e_{\text{seq}(i)}^a &= k_{\text{seq}(i)}^1 \oplus k_{\text{seq}(i)}^2 \oplus \dots \oplus k_{\text{seq}(i)}^n \oplus d^i \\ &= (\oplus_{h_{s \in S_1}}(t | \text{seq}(i))) \oplus (\oplus_{h_{s \in S_2}}(t | \text{seq}(i))) \oplus d^i \quad (4) \\ &= (\oplus_{h_{s \in S_a}}(t | \text{seq}(i))) \oplus d^i = c_{\text{seq}(i)} \oplus d^i \end{aligned}$$

Therefore, the aggregator first computes  $C$  in Step (2), and uses  $C$  to decrypt all the ciphertexts. The original data  $D$  is

output in Step (3), which can be divided as  $D = \{m[1, l], m[l+1, 2l], \dots, m[(n-1)l+1, nl]\}$ , where each  $m[x, y]$  is a user's original data. However, the aggregator cannot link each data with its owner, because  $\text{seq}(i)$  is unknown for him.

If the aggregator finds any abnormal data in  $D$ , for example,  $m[x, y]$ , it can request the TA to recover the identity of the malicious user. The aggregator sends all the data containing  $e^i$  and  $\sigma^i$  as well as  $x$  and  $y$  to the TA. If the  $\text{seq}(i)$  is known by the TA, it can directly find the malicious users. Otherwise, the TA recovers the corresponding secret encryption keys and real identities from the pseudo IDs, decrypts all the data and reveals the malicious users' identities.

## 6. Security Analysis

In this section, we analyze our framework and elaborate how our protocol can achieve the design goals under the security model. Specifically, we mainly focus on the following three aspects: why our protocol can hold  $k$ -source anonymity so that the only knowledge the server can get is that the data owner is one of the  $k$  participants in the system, why the collusion attack cannot help adversaries to recover the users secret keys, and why the data integrity can be protected and the identity authentication can be guaranteed.

**6.1. Our Protocol Is  $k$ -Source Anonymous.** As the definition of  $k$ -source anonymity listed in Section 4, we want to prove that if we interchange any two users' data in the same interval, the adversary including the aggregator cannot efficiently tell the difference.

Given a group of participants  $U = \{u_1, u_2, \dots, u_n\}$ , and their corresponding sensing data  $D = \{d^1, d^2, \dots, d^n\} \in M^n$ , where  $M = \{0, 1\}^l$  is the message space of  $d^i$ ,  $i = 1, 2, \dots, n$ . Each user  $u_i$  runs Algorithm 1 to encrypt his/her data  $d^i$  and sends the generated ciphertext  $e^i$  to the aggregator. Note that  $e_i$  has been defined in Step (2) of Algorithm 1. It is obvious

**Input:**

For each user  $i$  ( $i = 1, 2, \dots, n$ ), input his/her pseudo ID  $Uid_i$ , public signing key  $pk_i = g_2^{x_i}$  and uploaded data  $e^i$  and  $\sigma^i$ . The time period  $t$  and the aggregator's decryption keys are requested. The symbol  $|$  and  $\oplus$  have the same meaning in Algorithm 1.

**Output:**

The aggregator outputs all the users' original data  $D = d^1 | d^2 | \dots | d^n$  as follows:

**begin**

(1) Aggregate all the signatures and verify them:

$$\widehat{e} \left( \prod_{i=1}^n \sigma_i, g_2 \right) \stackrel{?}{=} \prod_{i=1}^n \widehat{e} (H(Uid | e^i), pk_i)$$

(2) If the equation in Step (1) does not hold, the algorithm outputs  $-1$ , otherwise continue to compute:

$$c_j = \oplus_{t \in S_a} h_s(t | j) \quad \text{for } j = 1, 2, \dots, n$$

$$C = c_1 | c_2 | \dots | c_n$$

(3) The aggregator calculates the final result as:

$$D = e^1 \oplus e^2 \oplus \dots \oplus e^n \oplus C.$$

(4) Output  $D$ .

**end**

ALGORITHM 2: Verify&amp;Dec.

that the length of  $e^i$  is  $n * l$  bits. Here we divide  $e^i$  into  $n$  parts as follows:

$$e^i = e_1^i | e_2^i | \dots | e_n^i, \quad (5)$$

where  $e_j^i = \{0\}^l \oplus k_j^i$  for  $j = 1, 2, \dots, n$ ,  $j \neq \text{Seq}(i)$ , and  $e_j^i = d^i \oplus k_j^i$  for  $j = \text{Seq}(i)$ . After all the users' data have been collected in this interval, the knowledge what the aggregator learns can be represented as:

$$V = (e_1^1 | e_2^1 | \dots | e_n^1, e_1^2 | e_2^2 | \dots | e_n^2, \dots, e_1^n | e_2^n | \dots | e_n^n), \quad (6)$$

Let any two participants switch their data, denoted as  $u_i$  and  $u_j$ , where  $1 \leq i < j \leq n$ , so all the users' original dataset is:  $D' = \{d^1, \dots, d^{i-1}, d^j, d^{i+1}, \dots, d^{j-1}, d^i, d^{j+1}, \dots, d^n\}$ . Then the aggregator's knowledge is changed to:

$$V' = (e_1^1 | \dots | e_{i-1}^1 | e_j^1 | e_{i+1}^1 | \dots | e_{j-1}^1 | e_i^1 | e_{j+1}^1 | \dots | e_n^1, e_1^2 | \dots | e_{i-1}^2 | e_j^2 | e_{i+1}^2 | \dots | e_{j-1}^2 | e_i^2 | e_{j+1}^2 | \dots | e_n^2, \dots, e_1^n | \dots | e_{i-1}^n | e_j^n | e_{i+1}^n | \dots | e_{j-1}^n | e_i^n | e_{j+1}^n | \dots | e_n^n) \quad (7)$$

According to the definition, if we want to prove that our protocol is  $k$ -source anonymous, it is equivalent to prove:

$$V \stackrel{c}{\equiv} V' \quad (8)$$

holds for any  $i, j$  where  $1 \leq i < j \leq n$ , and  $\forall D \in M^n$ .

To prove the correctness of this equation, we construct a simulator  $S$  to run the same protocol. First, the  $S$  generates a pseudorandom permutation function  $f_p : [1, n] \rightarrow [1, n]$ , and produces a permutation of  $[1, n]$  as  $[f_p(1), f_p(2), \dots, f_p(n)]$ . Then,  $S$  permutes the original dataset  $D$  to be:

$$D'' = \{d^{f_p(1)}, d^{f_p(2)}, \dots, d^{f_p(n)}\} \quad (9)$$

The protocol is executed with the input of  $D''$ , and outputs the corresponding result:

$$V'' = (e_1^{f_p(1)} | e_2^{f_p(1)} | \dots | e_n^{f_p(1)}, e_1^{f_p(2)} | e_2^{f_p(2)} | \dots | e_n^{f_p(2)}, \dots, e_1^{f_p(n)} | e_2^{f_p(n)} | \dots | e_n^{f_p(n)}). \quad (10)$$

Because we cannot distinguish  $D$  and its pseudorandom permutation  $D''$  in polynomial time, the  $V$  and  $V''$  are computationally indistinguishable. Therefore, the following equation holds:

$$V \stackrel{c}{\equiv} V''. \quad (11)$$

Similarly, we can draw the conclusion:

$$V' \stackrel{c}{\equiv} V''. \quad (12)$$

Otherwise,  $D'$  and its pseudorandom permutation  $D''$  can be distinguished in polynomial time. Therefore, we have:

$$V \stackrel{c}{\equiv} V''. \quad (13)$$

**6.2. Our Protocol Is Collusion-Resilient.** The ability to resist collusion attack depends on the size of each user's encryption keys and the aggregator's decryption keys. If we increase the size of  $S_i$ , or  $c$ , and the size of  $S_a$ , or  $q$ , the security level can be enhanced. Furthermore, when more users participate in the aggregation, our protocol can achieve better security.

In our scheme, the malicious users may collude to recover the aggregator's decryption keys, or collude with the aggregator to recover the other honest users' encryption keys. Let  $p_u$  denote the probability with which an honest user's key can be guessed successfully in a single trial,  $p_a$  denote the probability to recover the aggregator's key in a single trial, and



$\gamma$  denote the proportion of malicious users who collude with the aggregator. As we can see in [10], we have:

$$p_u \leq \frac{1}{\binom{(1-\gamma)nc}{c} \cdot \binom{(1-\gamma)n \lfloor (nc-q)/n \rfloor}{\lfloor (nc-q)/n \rfloor}}, \quad (14)$$

$$p_a \leq \frac{1}{\binom{(1-\gamma)nc}{q}}$$

Assuming that there are at most 30% malicious users in the group, and the security requirements are  $p_u \leq 2^{-80}$  and  $p_a \leq 2^{-80}$ . When the number of participants is  $10^2$ ,  $c$  and  $q$  can be set as 7 and 13 respectively, which means that the aggregator owns 13 decryption keys and every user owns 14 encryption keys at most. When the number of participants reaches  $10^3$ , we set the  $c = 5$  and  $q = 9$ .

Therefore, we can see when we set  $n = 100$ ,  $c = 7$ , and  $q = 13$ , or  $n = 1000$ ,  $c = 5$ , and  $q = 9$ ,  $p_u \leq 2^{-80}$  and  $p_a \leq 2^{-80}$  are satisfied. Because the probability to recover secret keys in a single trial is not bigger than  $2^{-80}$ , our protocol is collusion-resilient. Even if the number of users changes, we can adjust  $c$  and  $q$  to resist collusion attack.

**6.3. Our Protocol Achieves Authentication and Data Integrity.** Our protocol leverages aggregate signature as a built-in block to achieve authentication and the users' data integrity. The adversary can break authentication and data integrity if and only if it can forge the aggregate signature. However, the aggregate signature described in Section 4 is proven to be secure under the aggregate chosen-key security model [24]. We know the probability with which the adversary can generate valid aggregate signature is negligible. Furthermore, each user's data is bound with a pseudo ID. Therefore, authentication and data integrity are achieved.

## 7. Performance Evaluation

In this section, we implement our system and evaluate the performance of each instance of progress in our protocol, which demonstrates the efficiency and feasibility of our system. The comparison with other existing aggregation protocols is also performed with experiments and theoretical analysis.

As we can see, the main cost for the participants in our protocol is to encrypt and sign the uploaded data. We simulate participants' steps to examine the computational cost and elaborate the comparison with previous work. The aggregator's efficiency is affected by the verification and decryption, which is also shown to be accepted in the experiment.

**7.1. Implementation and Experimental Settings.** We implement our protocol in a desktop with Intel Core i7-4790 3.60 GHz CPU and 8 GB memory. The compilation environment is Visual Studio 2013 in Windows 10. And the cryptographic functions in the algorithm are provided by the MIRACL library. We use the same hash function as that

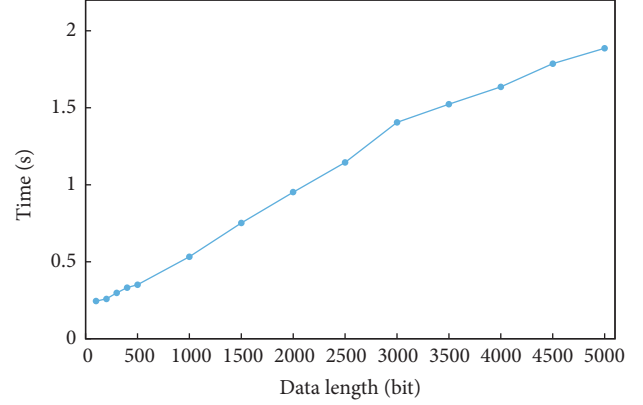


FIGURE 2: Computational cost of encryption at user side.

in [25], which uses HMAC-SHA512 as the pseudorandom function. For each  $l$ -bit raw data, if  $l < 512$ , the HMAC-SHA512 outputs a pseudorandom 512-bit string, which is truncated into  $l$ -bit substrings and taken exclusive-OR on all these substrings. When  $l > 512$ , we take several HMAC-SHA512 functions and concatenate their output, while the remaining part can use the same method as the condition  $l < 512$ .

All the users' data is generated by taking a uniform sample from  $[0, 2^l - 1]$ . The reason why we do not adopt the real-life data is that the value of the user's data does not affect the efficiency of our protocol. We only take experiments on the computational cost in both sides but do not consider the communication cost between them, because each user's data size is  $n * l$  bits, generally tens of or several kilobytes, which can be transmitted in short time. All the algorithms in our protocol are executed for 10 times. We take the average running time as the final output.

**7.2. Computational Cost at User Side.** All the computational cost at user side depends on Algorithm 1. In our scheme, if a user wants to encrypt the raw data, he/she needs to compute  $n * k$  pseudorandom functions, where  $k$  is the number of the user's keys; namely,  $k = |S_i|$ , for  $i = 1, 2, \dots, n$ .

Figure 2 shows the result of our computation time. We set the group size as  $n = 1000$ ,  $\gamma = 0.3$ ,  $p_u < 2^{-80}$ , and  $p_a < 2^{-80}$ . According to [10], we let  $c = 5$ , which means that each user owns nearly 10 encryption keys. During each time period, each user computes 10 pseudorandom functions and takes 10000 exclusive-OR operations on  $l$ -bit data. We can see that our algorithm is efficient and can be applied in real environment. In this figure, the encryption time increases linearly with the data length. If the data length is smaller than 2000 bits, the encryption time is no more than one second. When the data length reaches 5000 bits, we can finish the encryption within two seconds.

Figure 3 shows the relationship between the encryption time and the number of users when the data length is 1000 bits. Obviously, as the number of users increases, each user needs to compute more ciphertexts. However, it only

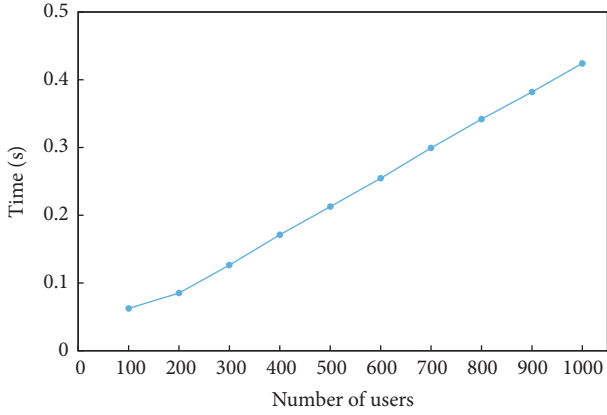


FIGURE 3: Computational cost of encryption at user side.

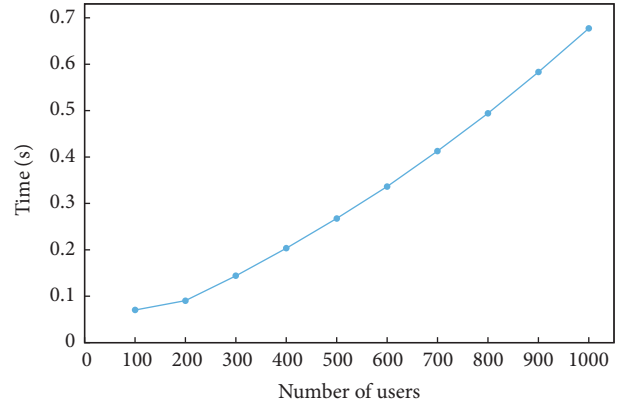


FIGURE 5: Computational cost of decryption at aggregator side.

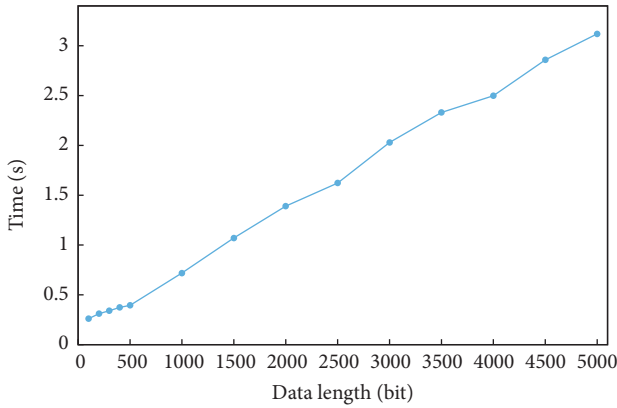


FIGURE 4: Computational cost of decryption at aggregator side.

takes 0.21 s for each user to take the encryption when the number of users increases to 500.

Here we compare our protocol with that in [11], to encrypt the raw data they have to compute 2 pseudorandom functions and take  $2n$  exclusive-OR operations on  $l$  bits' data. Recall that although our computational cost is five times as much as that in [11], we can achieve more security properties. When the data length is 100 bits, our protocol costs 0.24 s and the scheme proposed in [11] costs 0.05 s. When the data length is 1000 bits, 0.58 s and 0.12 s are needed, respectively.

We emphasize that we do not list the details of the comparison of signature time and verification time here, because the signature time depends much on the aggregate signature scheme. In fact, we only need to take 30 ms to sign the final ciphertext, when the data length is 5000 bits.

**7.3. Computational Cost at Aggregator Side.** Different from [11], the aggregator owns decryption keys to prevent others from obtaining all the raw data. In our scheme, the aggregator needs to compute  $nq$  pseudorandom functions and take  $n * (n + q)$  exclusive-OR operations on  $l$  bits' data. However, it takes  $n * (n + q)$  exclusive-OR operations on  $l$  bits' data in [11].

If we set  $n = 1000$ ,  $\gamma = 0.3$ ,  $p_u < 2^{-80}$ , and  $p_a < 2^{-80}$ , the result is shown in Figure 4. The aggregator only needs to take

several seconds to decrypt all the data when the data length is 5000 bits, and it only takes nearly one more second than [11] to resist the collusion attack. When the times of exclusive-OR operations increase, the time to compute pseudorandom functions does not dominate the whole running time any more. Experiment shows that it takes 0.26 s for decryption when  $l$  equals 100, while 0.031 s is needed in [11]. When  $l$  reaches 1000, our protocol and [11] consume 0.72 s and 0.23 s, respectively.

When the data length is set as 1000 bits, the computational cost at aggregator side is shown in Figure 5. The decryption time is proportional to the number of users. If the group includes 100 users, the decryption can be finished in 0.1 s. Even though the number of users grows up to 1000, the decryption time does not exceed one second.

## 8. Conclusion

In this paper, we propose a novel protocol to allow an untrusted aggregator to compute any aggregation functions based on all users' data. Collusion resistance is also achieved even if part of malicious users collude with the untrusted aggregator. The data collection can be finished in one-round communication, so bidirectional communication channel is not needed in our protocol. We also protect all users' data integrity by leveraging the aggregate signature. Security analysis shows that  $k$ -source anonymity is achieved. Through extensive performance evaluations, we have demonstrated that the proposed scheme is efficient at user/aggregator side. In our scheme, dynamic joining and exit of users have not been discussed. In the future, we will continue our efforts to address this issue.

## Notations

- $n$ : The number of users in the scheme
- $\dot{S}$ : The universal additive secret key set
- $\dot{S}_i$ : The user  $i$ 's additive secret key set
- $\hat{S}$ : The universal subtractive secret key set
- $\hat{S}_i$ : The user  $i$ 's subtractive secret key set
- $S_i$ : The user  $i$ 's encryption key set,  $S_i = \dot{S}_i \cup \hat{S}_i$

$S_a$ : The aggregator's decryption key set  
 $c$ : The size of  $\hat{S}_i$   
 $q$ : The size of  $S_a$   
 $d^i$ : The user  $i$ 's data  
 $l$ : The data length  
 $t$ : The time period  
 $H$ : A hash function,  $H : \{0, 1\}^* \rightarrow G_1$   
 $h_s(x)$ : A function indexed by  $s$  in a pseudorandom function family  $\mathcal{H}_{l, m + \lceil \log_2 n \rceil, l} = \{h_s : \{0, 1\}^{m + \lceil \log_2 n \rceil} \rightarrow \{0, 1\}^l\}_{s \in \{0, 1\}^l}$   
 $Uid_i$ : The user  $i$ 's pseudo ID  
 $pk_i$ : The user  $i$ 's public signing key for  $Uid_i$   
 $sk_i$ : The user  $i$ 's secret signing key for  $Uid_i$ .

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This research is supported by the National Natural Science Foundation of China (Grant nos. 61402037, 61272512, and 61300172), National Key Research and Development Program 2016YFB0800301, and DNSLAB, China Internet Network Information Center, Beijing 100190.

## References

- [1] A. Thiagarajan, L. Ravindranath, K. LaCurts et al., "VTrack: accurate, energy-aware road traffic delay estimation using mobile phones," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys '09)*, pp. 85–98, ACM, November 2009.
- [2] M. Mun, S. Reddy, K. Shilton et al., "PEIR, the personal environmental impact report, as a platform for participatory sensing systems research," in *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services (MobiSys '09)*, pp. 55–68, ACM, Kraków, Poland, June 2009.
- [3] S. Consolvo, D. W. McDonald, T. Toscos et al., "Activity sensing in the wild: a field trial of ubifit garden," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1797–1806, ACM, Florence, Italy, April 2008.
- [4] S. Gaonkar, J. Li, R. R. Choudhury, L. Cox, and A. Schmidt, "Micro-blog: sharing and querying content through mobile phones and social participation," in *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services*, pp. 174–186, ACM, Breckenridge, Colo, USA, June 2008.
- [5] T. Jung, X. Mao, X.-Y. Li, S.-J. Tang, W. Gong, and L. Zhang, "Privacy-preserving data aggregation without secure channel: multivariate polynomial evaluation," in *Proceedings of the IEEE INFOCOM*, pp. 2634–2642, IEEE, Turin, Italy, April 2013.
- [6] T. Jung, X.-Y. Li, and M. Wan, "Collusion-tolerable privacy-preserving sum and product calculation without secure channel," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 1, pp. 45–57, 2015.
- [7] M. Joye and B. Libert, "A scalable scheme for privacy-preserving aggregation of time-series data," in *Proceedings of the International Conference on Financial Cryptography and Data Security*, pp. 111–125, Springer, 2013.
- [8] F. Qiu, F. Wu, and G. Chen, "SLICER: a slicing-based K-anonymous privacy preserving scheme for participatory sensing," in *Proceedings of the 10th IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS '13)*, pp. 113–121, October 2013.
- [9] F. Qiu, F. Wu, and G. Chen, "Privacy and quality preserving multimedia data aggregation for participatory sensing systems," *IEEE Transactions on Mobile Computing*, vol. 14, no. 6, pp. 1287–1300, 2015.
- [10] Q. Li, G. Cao, and T. F. La Porta, "Efficient and privacy-aware data aggregation in mobile sensing," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 2, pp. 115–129, 2014.
- [11] Y. Zhang, Q. Chen, and S. Zhong, "Privacy-preserving data aggregation in mobile phone sensing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 5, pp. 980–992, 2016.
- [12] C. R. Perez, *Reputation-based resilient data aggregation in sensor network [M.S. thesis]*, Department of Electrical and Computer Engineering, Purdue University, 2007.
- [13] C. R. Perez-Toro, R. K. Panta, and S. Bagchi, "RDAS: reputation-based resilient data aggregation in sensor network," in *Proceedings of the 7th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '10)*, pp. 1–9, June 2010.
- [14] M. M. Groat, W. Hey, and S. Forrest, "KIPDA: K-indistinguishable privacy-preserving data aggregation in wireless sensor networks," in *Proceedings of the IEEE INFOCOM*, pp. 2024–2032, IEEE, April 2011.
- [15] C. Li and Y. Liu, "SRDA: smart reputation-based data aggregation protocol for wireless sensor network," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 105364, 10 pages, 2015.
- [16] X. Du, M. Guizani, Y. Xiao, and H.-H. Chen, "Defending DoS attacks on broadcast authentication in wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications (ICC '08)*, pp. 1653–1657, Beijing, China, May 2008.
- [17] M. Karaliopoulos, O. Telelis, and I. Koutsopoulos, "User recruitment for mobile crowdsensing over opportunistic networks," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '15)*, pp. 2254–2262, IEEE, 2015.
- [18] L. Gao, F. Hou, and J. Huang, "Providing long-term participation incentive in participatory sensing," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications and Networks (IEEE INFOCOM '15)*, pp. 2803–2811, Hong Kong, China, May 2015.
- [19] D. Boneh, E.-J. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts," in *Proceedings of the Theory of Cryptography Conference*, pp. 325–341, Springer, 2005.
- [20] Y. Yang, X. Wang, S. Zhu, and G. Cao, "SDAP: a secure hop-by-hop data aggregation protocol for sensor networks," *ACM Transactions on Information and System Security*, vol. 11, no. 4, article 18, 2008.
- [21] C. Cornelius, A. Kapadia, D. Kotz, D. Peebles, M. Shin, and N. Triandopoulos, "AnonySense: privacy-aware people-centric sensing," in *Proceedings of the 6th International Conference on*

*Mobile Systems, Applications, and Services*, pp. 211–224, ACM, June 2008.

- [22] I. Boutsis and V. Kalogeraki, “Privacy preservation for participatory sensing data,” in *Proceedings of the 11th IEEE International Conference on Pervasive Computing and Communications (PerCom '13)*, pp. 103–113, San Diego, Calif, USA, March 2013.
- [23] Y. Zhang, Q. Chen, and S. Zhong, “Efficient and privacy-preserving min and  $k$  th min computations in mobile sensing systems,” *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 1, pp. 9–21, 2017.
- [24] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, “Aggregate and verifiably encrypted signatures from bilinear maps,” in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 416–432, Springer, 2003.
- [25] C. Castelluccia, A. C.-F. Chan, E. Mykletun, and G. Tsudik, “Efficient and provably secure aggregation of encrypted data in wireless sensor networks,” *ACM Transactions on Sensor Networks*, vol. 5, no. 3, pp. 1–36, 2009.

## Research Article

# Performance Analysis of User Pairing Algorithm in Full-Duplex Cellular Networks

Wonjong Noh,<sup>1</sup> Wonjae Shin,<sup>2</sup> and Hyun-Ho Choi<sup>3</sup>

<sup>1</sup>Samsung Electronics Co., Ltd., Suwon 16677, Republic of Korea

<sup>2</sup>Department of Electrical and Computer Engineering, Seoul National University, Seoul 08826, Republic of Korea

<sup>3</sup>Department of Electrical, Electronic and Control Engineering, Hankyong National University, Anseong 17579, Republic of Korea

Correspondence should be addressed to Hyun-Ho Choi; hhchoi@hknu.ac.kr

Received 9 December 2016; Accepted 5 March 2017; Published 23 March 2017

Academic Editor: Hideyuki Takahashi

Copyright © 2017 Wonjong Noh et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In a full duplexing (FD) wireless cellular network, a base station operates in FD mode, while the downlink (DL) and uplink (UL) users operate in half duplexing (HD) mode. Thus, the downlink and uplink transmissions occur simultaneously so that interuser interference from a UL to a DL user occurs. In an FD network, the main challenge to minimize the interuser interference is *user pairing*, which determines a pair of DL and UL users who use the same radio resource simultaneously. We formulate a nonconvex optimization problem for user pairing to maximize the cell throughput. Then, we propose a heuristic user pairing algorithm with low complexity. This algorithm is designed such that the DL user having a better signal quality has higher priority to choose its paired UL user for throughput maximization. Thereafter, we conduct theoretical performance analysis of the FD cellular system based on stochastic geometry and analyze the impact of the user pairing algorithm on the performance of the FD cellular system. Results show that the FD system that uses the proposed user pairing algorithm effectively reduces the interuser interference and approaches optimal performance. It also considerably outperforms the FD system using a random user pairing and almost doubles the conventional HD system in terms of cell throughput.

## 1. Introduction

In traditional wireless communication, the duplexing mode for transmitting and receiving signals at a node commonly is half duplex (HD), such as frequency division duplexing (FDD) and time division duplexing (TDD). As another duplexing mode, full duplexing (FD) has recently received considerable attention because FD enables a node to transmit and receive signals simultaneously on the same frequency so that spectral efficiency can be improved twice in theory. However, the main challenge in applying the FD to wireless networks is the manner in which one can handle cochannel interference not only between the transmit (Tx) and receive (Rx) antennas at the base station (BS), but also between the uplink (UL) and downlink (DL) communication users. The cochannel interference between the Tx and Rx at the same station is called *self-interference*, whereas that between the different stations is called *interuser interference*.

To address self-interference, a smart code that divides the transmitting and receiving signals has been invented and a code division duplexing (CDD) system using this coding technique has been studied [1, 2]. Then, the CDD system has been shown to achieve twice the spectral efficiency of the HD system [3]. In addition, advanced antenna cancellation techniques that cancel the self-interference between Tx and Rx on the radio frequency (RF) band have been proposed and real performances by implementation have been demonstrated [4–6]. Unlike these physical layer solutions, scheduling on the medium access control (MAC) layer has been another research issue studied in order to tackle the self-interference problem in an FD system. The self-interference problems at the wireless relay node with the FD have been resolved by dynamic resource allocation schemes [7–9]. In a practical FD cellular system in which multiple Rx's of the BS are distributed at a distance with the Tx of BS, scheduling algorithms that determine both subcarrier allocation and the communication

direction between UL and DL users have been designed to maximize cell throughput [10].

To address interuser interference in FD systems, various algorithms including resource allocation, scheduling, user pairing, and their optimization have been studied to improve system performance with respect to cell throughput, cell coverage, and outage probability. Some initial studies were conducted to examine various resource allocation-related problems in FD cellular networks [11–13]. Resource block (RB) allocation was considered in [11] for an FD cellular network, and interference between users was reduced as a result. In [12], a two-user FD cellular network was considered and a noncooperative game was proposed for radio resource allocation in such networks. In [13], a power efficient resource allocation algorithm was designed to achieve total transmit power minimization in an FD cellular network.

In addition, various scheduling algorithms for an FD system that consider nonconvex and combinatorial optimization problems were studied in [14]. The Janus protocol in [15] also considers a scheduling algorithm for scheduling a transmission mode (HD or FD mode) to maximize in-band FD transmission opportunity, while maintaining fair channel access for all nodes. Recently, a MAC scheduling algorithm that employs cooperative FD relays (FDRs) was proposed in [16]. This study revealed that additional throughput can be achieved by using cooperative FDRs. In [17], joint user scheduling and channel allocation for cellular networks was considered and a suboptimal heuristic algorithm with low computational complexity was proposed. Furthermore, two heuristic user pairing algorithms were designed for throughput maximization and outage minimization, respectively [18]. Simulation results showed that these two algorithms approach optimal performance under the assumption of no residual self-interference at the FD BS. However, up-to-date studies and theoretical performance analysis on user pairing algorithms in FD cellular network environments have not been thoroughly conducted. Compared with [18], our study newly conducts theoretical analysis of the FD cellular system based on stochastic geometry after adopting the user pairing algorithm previously proposed and then investigates the theoretical performance of FD cellular networks according to various system parameters, which have not been considered in [18].

In this study, we consider FD wireless cellular networks in which the BS operates in FD mode while DL and UL users operate in HD mode. Because this FD system induces severe interuser interference from UL to DL users, the main concern in maximizing system performance is user pairing, which determines a pair of DL and UL users that use the same radio resource simultaneously. We formulate the optimization problems for user pairing to maximize cell throughput. Solving these optimization problems by means of exhaustive search is highly complex; thus we propose a suboptimal user pairing algorithm with low computation complexity for practical use. In a heuristic manner, the proposed user pairing algorithm considers the signal quality of users and the direct channel between users. Thereafter, we perform theoretical analysis of the FD cellular system based on stochastic geometry and investigate the effects of adopting

the user pairing algorithm. Analysis and simulation show that the FD system in conjunction with the proposed user pairing algorithm approaches optimal performance and greatly improves cell throughput compared to the conventional HD system.

The rest of this paper is organized as follows. Section 2 describes the system model of the considered FD wireless cellular network. In Section 3, user pairing algorithm is proposed for throughput maximization. Section 4 provides stochastic geometry-based analysis of the FD cellular network. In Section 5, analysis and simulation results are given and the performance gain of the proposed pairing algorithm is verified. Conclusion remarks are provided in Section 6.

## 2. System Description

We consider a cellular network composed of multiple cells in which each cell is surrounded by  $N$  neighbor cells, as shown in Figure 1. The BS works in FD mode while DL and UL users work in HD mode due to the limited processing ability of user terminals. We assume that a total of  $2M$  users are present in a cell and that both the number of UL users and the number of DL users are  $M$  equally. Moreover, one transmission frame is divided into  $M$  time slots and each user is allowed to use only one time slot within a frame to transmit or receive data. This time slot allocation corresponds to the manner of round-robin scheduling of radio resources and thus provides some degree of fairness for all users [19].

Because the DL and UL users transmit signals on the same resource simultaneously, the DL user experiences three different types of interferences from (i) the UL user scheduled in the same slot in the serving cell, (ii) the neighboring BSs, and (iii) the UL users using the same slot in neighboring cells. Therefore, the received signal at DL user  $i$  is expressed as

$$y_i = \sqrt{P_0}h_{d,i}x_i + \underbrace{\sqrt{P_j}\varphi_{ij}x_j}_{\text{interuser intf.}} + \underbrace{\sum_{n=1}^N \sqrt{P_0}g_{in}\tilde{x}_n}_{\text{intf. from neighbor BSs}} + \underbrace{\sum_{n=1}^N \sqrt{P_n}\varphi_{in}\tilde{x}_n}_{\text{intf. from neighbor UL users}} + n_i, \quad (1)$$

where  $P_0$  and  $P_j$  are the transmission power of the BS and UL user  $j$ , respectively. The variables  $x_i$ ,  $x_j$ , and  $x_n$  denote the signals sent from the serving BS<sub>0</sub>, the UL user  $j$  using the same time slot, and the  $n$ th neighboring cell, respectively. As illustrated in Figure 1, the channel coefficients  $h_{d,i}$  and  $\varphi_{ij}$  correspond to the links to DL user  $i$  from its serving BS and the UL user  $j$ , respectively. In addition, the channel coefficients  $g_{in}$  and  $\varphi_{in}$  correspond to the links to DL user  $i$  from the BS and UL user in the  $n$ th neighboring cell, respectively. Here, all the channel coefficients are assumed to be  $\mathcal{E}\mathcal{N}(0, \sigma^2/d_{ij}^\alpha)$ , where  $d_{ij}$  is the transmission distance of the corresponding  $ij$  link and  $\alpha$  denotes the path loss exponent. Finally,  $n_i$  is the additive white Gaussian noise with zero mean and  $N_0$  variance.

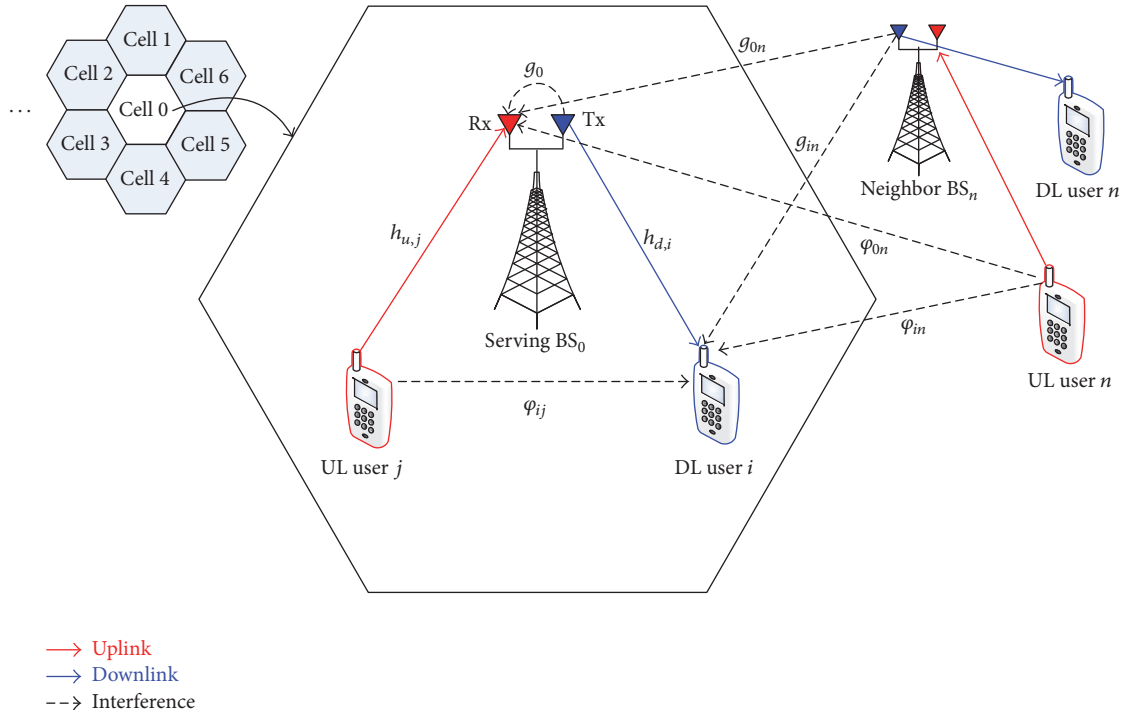


FIGURE 1: Considered full-duplex cellular networks.

In addition, the Rx of the BS experiences three kinds of interferences from (i) the Tx of the serving BS, (ii) the neighboring BSs, and (iii) the UL users using the same slot in neighboring cells. Therefore, the signal received from the UL user  $j$  at the Rx of the serving BS<sub>0</sub> is expressed as

$$\begin{aligned}
 y_j = & \sqrt{P_j} h_{u,j} x_j + \underbrace{\sqrt{P_0} g_0 x_i}_{\text{self-interference}} + \underbrace{\sum_{n=1}^N \sqrt{P_0} g_{0n} \hat{x}_n}_{\text{intf. from neighbor BSs}} \\
 & + \underbrace{\sum_{n=1}^N \sqrt{P_n} \varphi_{0n} \tilde{x}_n}_{\text{intf. from neighbor UL users}} + n_0,
 \end{aligned} \quad (2)$$

where  $h_{u,j}$ ,  $g_0$ ,  $g_{0n}$ , and  $\varphi_{0n}$  represent the channel coefficients corresponding to the links from the UL user  $j$ , the Tx of the serving BS<sub>0</sub>, the  $n$ th neighboring BS, and the  $n$ th neighboring UL user, respectively.

In the FD system, the Rx of the BS experiences strong interferences from the Txs of both the serving and neighboring BSs (relatively, the interferences from the Txs of UL users in the neighboring cells are weak because of their small transmission power and the long distance to the neighboring BS [10]). However, the BS with FD function can eliminate these kinds of interferences by using the antenna's cancellation techniques and channel estimation techniques [5, 10]. For the purpose of channel estimation, we assume that the Rx at the BS employs pilot symbols, which are transmitted from the Txs of both the serving BS<sub>0</sub> and neighboring BSs. Although the received interference power is

very strong compared to the receiving signal strength at the BS, it can be reduced to the same level as the receiver noise floor by applying analog and digital cancellation techniques sequentially [4–6]. Moreover, by adopting a proper channel estimation technique, such as least-square and minimum mean square error, we can estimate the channel coefficients  $g_0$  and  $g_{0n}$  [20] (details about channel estimation techniques are out of the scope of this study; the channel estimation at the BS is assumed to be perfect for the practicality of the FD). Based on this interference channel estimation and the knowledge of the signals sent from the Txs of the serving BS<sub>0</sub> and neighboring BSs through the wired backhaul, the Rx of the serving BS<sub>0</sub> can eliminate the corresponding interference parts from the received signal. After interference cancellation, the postprocessed signal of UL user  $j$  is expressed as

$$\begin{aligned}
 \hat{y}_j = & \sqrt{P_j} h_{u,j} x_j + \sqrt{P_0} (g_0 - \hat{g}_0) x_i \\
 & + \sum_{n=1}^N \sqrt{P_0} (g_{0n} - \hat{g}_{0n}) \hat{x}_n + \sum_{n=1}^N \sqrt{P_n} \varphi_{0n} \tilde{x}_n + n_0,
 \end{aligned} \quad (3)$$

where  $\hat{g}_0$  and  $\hat{g}_{0n}$  denote the estimated channel coefficients.

From (1) and (3), the signal-to-interference-plus-noise ratios (SINRs) of the received signals  $y_i$  and  $\hat{y}_j$  are, respectively, given by

$$\gamma_i = \frac{P_0 \|h_{d,i}\|^2}{P_j \|\varphi_{ij}\|^2 + \sum_{n=1}^N P_0 \|g_{in}\|^2 + \sum_{n=1}^N P_n \|\varphi_{in}\|^2 + N_0}, \quad (4)$$

$$\gamma_j = \frac{P_j \|h_{u,j}\|^2}{P_0 \|g_0 - \hat{g}_0\|^2 + \sum_{n=1}^N P_0 \|g_{0n} - \hat{g}_{0n}\|^2 + \sum_{n=1}^N P_n \|\varphi_{0n}\|^2 + N_0}. \quad (5)$$

Consequently, the rates (i.e., spectral efficiencies) of DL user  $i$  and UL user  $j$  are given by

$$\begin{aligned} R_i &= \log_2(1 + \gamma_i), \\ R_j &= \log_2(1 + \gamma_j). \end{aligned} \quad (6)$$

Finally, the cell throughput of FD system is expressed as

$$S_{\text{FD}} = \sum_{i=1}^M R_i + \sum_{j=1}^M R_j. \quad (7)$$

In the conventional HD system (e.g., TDD system) in which neither self-interference nor interuser interference exists, the SINRs of DL user  $i$  and UL user  $j$  are, respectively, expressed as

$$\begin{aligned} \gamma_i^{\text{HD}} &= \frac{P_0 \|h_{d,i}\|^2}{\sum_{n=1}^N P_0 \|g_{in}\|^2 + N_0}, \\ \gamma_j^{\text{HD}} &= \frac{P_j \|h_{u,j}\|^2}{\sum_{n=1}^N P_n \|\varphi_{0n}\|^2 + N_0}. \end{aligned} \quad (8)$$

Therefore, the cell throughput of HD system is expressed as

$$S_{\text{HD}} = \frac{1}{2} \left\{ \sum_{i=1}^M \log_2(1 + \gamma_i^{\text{HD}}) + \sum_{j=1}^M \log_2(1 + \gamma_j^{\text{HD}}) \right\}. \quad (9)$$

### 3. Proposed User Pairing Algorithm

Based on (4) and (5), the parameters that we can control in the FD system are  $P_j$  and  $\varphi_{ij}$ . The other parameters are given based on the node position or are generally fixed. Here,  $\varphi_{ij}$  is controlled by the user pairing. Thus, the user pairing influences the DL SINR and cell throughput. Our objective is to design a novel user pairing algorithm to optimize cell throughput.

We denote by  $(i, j)$  a user pair in which DL user  $i$  and UL user  $j$  are allocated to the same time slot. Possible user pairs are derived by choosing a number  $i$  in  $\{1, 2, \dots, M\}$  and a number  $j$  in  $\{1, 2, \dots, M\}$  under the constraint that the chosen number cannot be selected again. Let  $\Pi$  denote the set of all possible user pairs. For example, for  $M = 3$ ,  $\Pi$  becomes

$$\begin{aligned} \Pi &= \{(1, 1), (2, 2), (3, 3)\}, \{(1, 1), (2, 3), (3, 2)\}, \\ &\{(1, 2), (2, 1), (3, 3)\}, \{(1, 2), (2, 3), (3, 1)\}, \\ &\{(1, 3), (2, 1), (3, 2)\}, \{(1, 3), (2, 2), (3, 1)\}, \end{aligned} \quad (10)$$

where six configurations exist. For a general  $M$ , the number of possible configurations for user pairing is given by  $M!$ . We denote by  $\pi_k$  the  $k$ th element of  $\Pi$ , where  $k = 1, 2, \dots, M!$ .

For the purpose of maximizing cell throughput, the optimization problem is formulated as

$$\begin{aligned} \pi_k &= \arg \max \sum_{(i,j) \in \pi_k} R_i + R_j \\ \text{s. t. } &P_j = P_{\max} \end{aligned} \quad (11)$$

$$i, j \in \{1, 2, \dots, M\},$$

where  $P_{\max}$  is the maximum transmission power of the UL user. In this problem, every UL user should set its transmission power  $P_j$  to  $P_{\max}$  to maximize its UL throughput. To solve this nonconvex and combinatorial optimization problem, we must search the total  $M!$  configurations for the user pairs. This computation complexity is  $O(M!)$ , which is surprisingly high with a large  $M$  (i.e., a large number of users). Therefore, we propose a suboptimal pairing algorithm with low complexity for practical use.

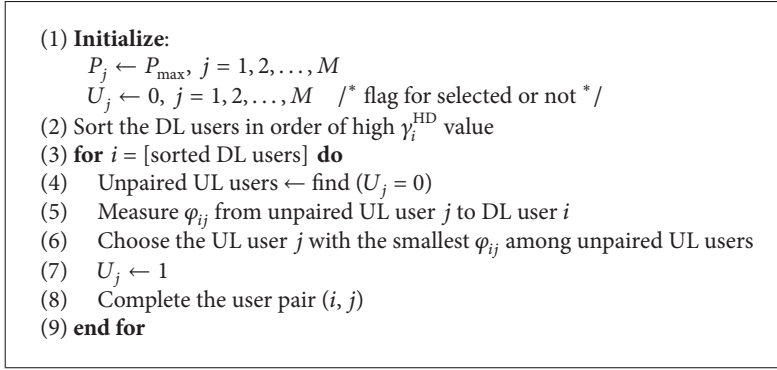
To design a user pairing algorithm, we can obtain insight from (4) and (5). As shown in (5), the SINR of UL user  $\gamma_j$  is not affected by DL users. If we assume that the channel estimation and interference cancellation are perfect and the interference from neighbor UL users is negligible,  $\gamma_j$  depends on mainly UL channel  $h_{u,j}$ . By contrast, the SINR of DL user  $\gamma_i$  depends not only on the DL channel,  $h_{d,i}$ , but also on the direct channel from UL user  $j$  to DL user  $i$ ,  $\varphi_{ij}$ . Therefore, the user pairing influences only the SINR of DL user  $\gamma_i$ . To maximize the sum rate of all users, having the user with a better signal quality receive less interference is reasonable [21]. Therefore, we have the DL user with a better signal quality (i.e., a higher value of  $\gamma_i^{\text{HD}}$ ) choose first the UL user with a smaller  $\varphi_{ij}$  as its partner. This approach improves the SINR of the DL user with a better signal quality more by reducing more interference from the UL user, and it eventually increases the sum rate of all users.

The proposed user pairing algorithm is given as Algorithm 1. First, the transmit power of all UL users is initialized to their available maximum power,  $P_{\max}$ , in order to maximize the UL SINR at the BS. Moreover, the flag indicates whether the UL user is selected for user pairing and is initialized to 0 for all  $j$ . Thereafter, the algorithm sorts the DL user in order from the highest to lowest DL SINR value and then runs based on these sorted DL users. For DL user  $i$ , the pairing algorithm measures the direct channel  $\varphi_{ij}$  from the unpaired UL user  $j$  to the DL user  $i$ . Then, the UL user having the smallest  $\varphi_{ij}$  value among the unpaired UL users is selected for pairing. Accordingly, the flag of this UL user  $j$  is set to 1; thus, the DL user  $i$  and selected UL user  $j$  are paired with each other. This pairing operation is repeated until all sorted DL users are paired with the remaining UL users. Note that the proposed algorithm depends on only one iteration and one search within each iteration for  $M$  DL/UL users so that its computation complexity becomes  $O(M^2)$ .

### 4. Performance Analysis

**4.1. Assumptions and Notations.** For the stochastic geometry-based analysis, we assume a homogeneous cellular network that consists of macro-cell BSs and users. The BSs are





ALGORITHM 1: Proposed user pairing algorithm.

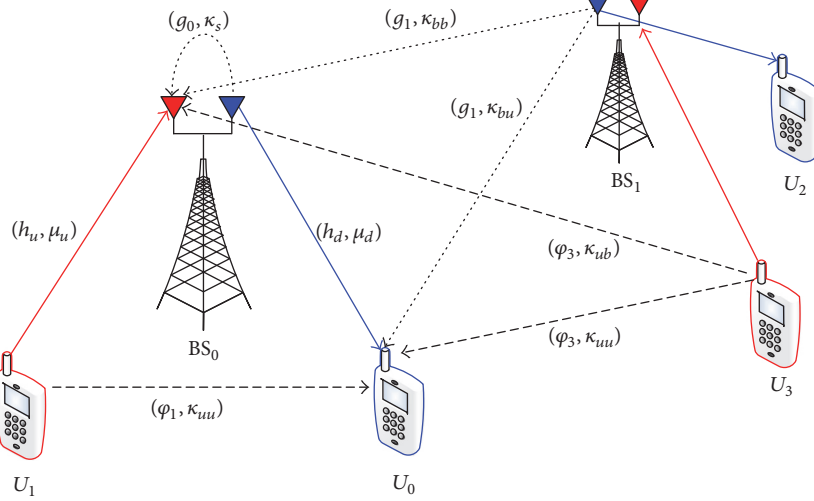


FIGURE 2: System model and notations.

arranged according to a homogeneous Poisson point process (PPP)  $\Phi$  with intensity  $\lambda$  in the Euclidean plane [22, 23]. We also assume that DL and UL users are distributed according to an independent homogeneous PPP  $\Omega$  with the same density  $\lambda$  so that the FD BS schedules one UL user and one DL user concurrently during a given time/frequency resource unit [23, 24]. Each user is connected to its closest BS.

Figure 2 shows the system model where a BS<sub>0</sub> simultaneously serves its DL user  $u_0$  and UL user  $u_1$  through FD. While  $u_0$  receives and  $u_1$  transmits a signal from and to BS<sub>0</sub> using FD,  $u_0$  receives interferences from  $u_1$ ,  $u_3$ , and BS<sub>1</sub>, and  $u_1$  receives interferences from  $u_3$ , BS<sub>0</sub>, and BS<sub>1</sub>. Normal line with  $(h, \cdot)$  denotes the signal channel. Also, dotted line with  $(g, \cdot)$  and dashed line with  $(\varphi, \cdot)$  denote interference channels by the neighboring BSs and users, respectively. Here, we assume that a tagged BS and tagged user experience Rayleigh fading with unit mean and employ a constant transmit power of  $1/\mu$ . In this case, the received power at a typical node at a distance  $r$  from its BS is  $hr^{-\alpha}$ , where the random variable  $h$  follows an exponential distribution with mean  $1/\mu$  (i.e.,  $h \sim \exp(\mu)$ ). It is denoted as  $(h, \mu)$ . Furthermore, all BSs and users experience a standard power loss propagation with path loss exponent

$\alpha > 2$  and additive and constant noise power of  $\sigma^2$ . All results are related to a single transmit antenna and single receive antenna, although future extension to multiple antennas is clearly desirable.

**4.2. Downlink Rate in Full-Duplex.** When a typical DL user,  $u_0$ , has a distance  $r$  and fading strength  $h_d \sim \exp(\mu_d)$  from its serving BS, the user  $u_0$  receives a signal with a power of  $h_d r^{-\alpha}$ . However, the user  $u_0$  experiences cumulative interferences  $I_b$  and  $I_u$  from neighbors, where  $I_b$  is the sum of the received powers from all other BSs, each with distance  $r_i$  and fading strength  $g_i \sim \exp(\kappa_{bu})$  to the user  $u_0$  such that

$$I_b = \sum_{i \in \Phi_b} g_i r_i^{-\alpha}, \quad (12)$$

where  $\Phi_b$  is the set of all interfering BSs. In addition,  $I_u$  is the sum of the received powers from all other UL users, each with distance  $r_i$  and fading strength  $\varphi_i \sim \exp(\kappa_{uu})$  to the user  $u_0$  such that

$$I_u = \sum_{i \in \Omega_u} \varphi_i r_i^{-\alpha}, \quad (13)$$

where  $\Omega_u$  is the set of all interfering users. The SINR of the typical DL user  $u_0$  can then be expressed as

$$\gamma_{\text{DL}} = \frac{h_d r^{-\alpha}}{I_b + I_u + \sigma^2}. \quad (14)$$

The average rate of the typical DL user  $u_0$  is

$$R_{\text{DL}}(\lambda, \alpha) = \mathbf{E}_{I_b, I_u, r} [\ln(1 + \gamma_{\text{DL}}(I_b, I_u, r))]. \quad (15)$$

Because the probability that the closest BS has a distance  $r$  is  $e^{-\lambda\pi r^2} 2\pi\lambda r$ , (15) is calculated as

$$\begin{aligned} R_{\text{DL}}(\lambda, \alpha) &= \int_{r>0} e^{-\lambda\pi r^2} \mathbf{E}_{I_b, I_u} \left[ \ln \left( 1 + \frac{h_d r^{-\alpha}}{I_b + I_u + \sigma^2} \right) \mid r \right] \\ &\quad \cdot 2\pi\lambda r \, dr, \end{aligned} \quad (16)$$

where

$$\begin{aligned} &\mathbf{E}_{I_b, I_u} \left[ \ln \left( 1 + \frac{h_d r^{-\alpha}}{I_b + I_u + \sigma^2} \right) \mid r \right] \\ &= \int_{t>0} \mathbf{P}_{I_b, I_u} \left[ \ln \left( 1 + \frac{h_d r^{-\alpha}}{I_b + I_u + \sigma^2} \right) > t \mid r \right] dt \\ &= \int_{t>0} \mathbf{P}_{I_b, I_u} [h_d > r^\alpha (\sigma^2 + I_b + I_u) (e^t - 1) \mid r] dt \\ &= \int_{t>0} \mathbf{E}_{I_b, I_u} [\mathbf{P} [h_d > r^\alpha (\sigma^2 + I_b + I_u) (e^t - 1) \mid r, I_b, I_u]] dt \\ &\quad \text{since } h_d \sim \exp(\mu_d), \\ &\mathbf{E}_{I_b, I_u} \left[ \ln \left( 1 + \frac{h_d r^{-\alpha}}{I_b + I_u + \sigma^2} \right) \mid r \right] \\ &= \int_{t>0} \mathbf{E}_{I_b, I_u} [e^{-\mu_d r^\alpha (\sigma^2 + I_b + I_u) (e^t - 1)}] dt \\ &= \int_{t>0} \mathbf{E}_{I_b, I_u} [e^{-\mu_d r^\alpha \sigma^2 (e^t - 1)} e^{-\mu_d r^\alpha (I_b + I_u) (e^t - 1)}] dt \\ &= \int_{t>0} e^{-\mu_d r^\alpha \sigma^2 (e^t - 1)} \mathbf{E}_{I_b} [e^{-\mu_d r^\alpha I_b (e^t - 1)}] \\ &\quad \cdot \mathbf{E}_{I_u} [e^{-\mu_d r^\alpha I_u (e^t - 1)}] dt \\ &= \int_{t>0} e^{-\mu_d r^\alpha \sigma^2 (e^t - 1)} \mathcal{L}_{I_b} (\mu_d r^\alpha (e^t - 1)) \\ &\quad \cdot \mathcal{L}_{I_u} (\mu_d r^\alpha (e^t - 1)) dt. \end{aligned} \quad (17)$$

In (17), the Laplace transforms of  $I_b$  and  $I_u$  are obtained as the following two lemmas, respectively.

**Lemma 1.** Laplace transform of  $I_b$

$$\begin{aligned} \mathcal{L}_{I_b} (\mu_d r^\alpha (e^t - 1)) &= \exp \left( -\pi\lambda r^2 (\beta_{d,bu} (e^t - 1))^{2/\alpha} \right. \\ &\quad \cdot \left. \int_{(\beta_{d,bu}(e^t-1))^{-2/\alpha}}^{\infty} \frac{1}{1+x^{\alpha/2}} dx \right). \end{aligned} \quad (18)$$

*Proof.*

$$\begin{aligned} \mathcal{L}_{I_b} (s) &= \mathbf{E}_{I_b} [e^{-sI_b}] \\ &= \mathbf{E}_{\Phi} [\Pi_{i \in \Phi \setminus \{b_0\}} \mathbf{E}_{g_i} [\exp(-sg_i R_i^{-\alpha})]] \\ &= \mathbf{E}_{\Phi} \left[ \Pi_{i \in \Phi \setminus \{b_0\}} \frac{\kappa_{bu}}{\kappa_{bu} + sR_i^{-\alpha}} \right] \\ &= \exp \left( -\lambda \int_{\mathbb{R}^2} \left( 1 - \frac{\kappa_{bu}}{\kappa_{bu} + sx^{-\alpha}} \right) dx \right) \end{aligned} \quad (19)$$

where we employ 1-dimension integral by radius  $v$ ,

$$\mathcal{L}_{I_b} (s) = \exp \left( -2\pi\lambda \int_r^{\infty} \left( 1 - \frac{\kappa_{bu}}{\kappa_{bu} + sv^{-\alpha}} \right) v \, dv \right).$$

Therefore,

$$\begin{aligned} &\mathcal{L}_{I_b} (\mu_d r^\alpha (e^t - 1)) \\ &= \exp \left( -2\pi\lambda \int_r^{\infty} \left( 1 - \frac{1}{1 + \mu_d / \kappa_{bu} (e^t - 1) (r/v)^\alpha} \right) \right. \\ &\quad \cdot v \, dv \Big) \quad \text{where we put } \frac{\mu_d}{\kappa_{bu}} = \beta_{d,bu} \\ &\mathcal{L}_{I_b} (\mu_d r^\alpha (e^t - 1)) \\ &= \exp \left( -2\pi\lambda \int_r^{\infty} \left( 1 - \frac{1}{1 + \beta_{d,bu} (e^t - 1) (r/v)^\alpha} \right) v \, dv \right) \\ &= \exp \left( -2\pi\lambda \int_r^{\infty} \frac{1 + \beta_{d,bu} (e^t - 1) (r/v)^\alpha - 1}{1 + \beta_{d,bu} (e^t - 1) (r/v)^\alpha} v \, dv \right) \\ &= \exp \left( -2\pi\lambda \int_r^{\infty} \frac{\beta_{d,bu} (e^t - 1) (r/v)^\alpha}{1 + \beta_{d,bu} (e^t - 1) (r/v)^\alpha} v \, dv \right) \\ &= \exp \left( -2\pi\lambda \int_r^{\infty} \frac{\beta_{d,bu} (e^t - 1)}{\beta_{d,bu} (e^t - 1) + (v/r)^\alpha} v \, dv \right) \\ &\quad \text{where we put } \frac{v^2}{r^2} = u, \\ &\mathcal{L}_{I_b} (\mu_d r^\alpha (e^t - 1)) \\ &= \exp \left( -\pi\lambda r^2 \int_1^{\infty} \frac{\beta_{d,bu} (e^t - 1)}{\beta_{d,bu} (e^t - 1) + u^{\alpha/2}} du \right) \end{aligned}$$

$$\begin{aligned}
 &= \exp\left(-\pi\lambda r^2 \int_1^\infty \frac{1}{1 + ((\beta_{d,bu}(e^t - 1))^{-2/\alpha} u)^{\alpha/2}} du\right) \\
 &\quad \text{where we put } (\beta_{d,bu}(e^t - 1))^{-2/\alpha} u = x, \\
 \mathcal{L}_{I_b}(\mu_d r^\alpha (e^t - 1)) &= \exp\left(-\pi\lambda r^2 (\beta_{d,bu}(e^t - 1))^{2/\alpha}\right. \\
 &\quad \cdot \left. \int_{(\beta_{d,bu}(e^t - 1))^{-2/\alpha}}^\infty \frac{1}{1 + x^{\alpha/2}} dx\right). \tag{20}
 \end{aligned}$$

**Lemma 2.** Laplace transform of  $I_u$

$$\begin{aligned}
 &\mathcal{L}_{I_u}(\mu_d r^\alpha (e^t - 1)) \\
 &= \exp\left(-\pi\lambda r^2 (\beta_{d,uu}(e^t - 1))^{2/\alpha} \int_0^\infty \frac{1}{1 + x^{\alpha/2}} dx\right). \tag{21}
 \end{aligned}$$

*Proof.*

$$\begin{aligned}
 \mathcal{L}_{I_u}(s) &= \mathbf{E}_{I_u}[e^{-sI_u}] \\
 &= \mathbf{E}_\Omega[\Pi_{i \in \Omega/\{u_o\}} \mathbf{E}_{\varphi_i}[\exp(-s\varphi_i R_i^{-\alpha})]] \\
 &= \mathbf{E}_\Omega\left[\Pi_{i \in \Omega/\{u_o\}} \frac{\kappa_{uu}}{\kappa_{uu} + sR_i^{-\alpha}}\right] \tag{22} \\
 &= \exp\left(-2\pi\lambda \int_0^\infty \left(1 - \frac{\kappa_{uu}}{\kappa_{uu} + sv^{-\alpha}}\right) v dv\right).
 \end{aligned}$$

Therefore,

$$\begin{aligned}
 &\mathcal{L}_{I_u}(\mu_d r^\alpha (e^t - 1)) \\
 &= \exp\left(-2\pi\lambda \int_0^\infty \left(1 - \frac{1}{1 + \mu_d/\kappa_{uu}(e^t - 1)(r/v)^\alpha}\right) v dv\right) \\
 &\quad \text{where we put } \frac{\mu_d}{\kappa_{uu}} = \beta_{d,uu},
 \end{aligned}$$

$$\begin{aligned}
 &\mathcal{L}_{I_u}(\mu_d r^\alpha (e^t - 1)) \\
 &= \exp\left(-2\pi\lambda \int_0^\infty \left(1 - \frac{1}{1 + \beta_{d,uu}(e^t - 1)(r/v)^\alpha}\right) v dv\right) \\
 &= \exp\left(-2\pi\lambda \int_0^\infty \frac{1 + \beta_{d,uu}(e^t - 1)(r/v)^\alpha - 1}{1 + \beta_{d,uu}(e^t - 1)(r/v)^\alpha} v dv\right) \\
 &= \exp\left(-2\pi\lambda \int_0^\infty \frac{\beta_{d,uu}(e^t - 1)(r/v)^\alpha}{1 + \beta_{d,uu}(e^t - 1)(r/v)^\alpha} v dv\right) \\
 &= \exp\left(-2\pi\lambda \int_0^\infty \frac{\beta_{d,uu}(e^t - 1)}{\beta_{d,uu}(e^t - 1) + (v/r)^\alpha} v dv\right) \\
 &\quad \text{where we put } \frac{v^2}{r^2} = u,
 \end{aligned}$$

$$\begin{aligned}
 &\mathcal{L}_{I_u}(\mu_d r^\alpha (e^t - 1)) \\
 &= \exp\left(-\pi\lambda r^2 \int_0^\infty \frac{\beta_{d,uu}(e^t - 1)}{\beta_{d,uu}(e^t - 1) + u^{\alpha/2}} du\right) \\
 &= \exp\left(-\pi\lambda r^2 \int_0^\infty \frac{1}{1 + ((\beta_{d,uu}(e^t - 1))^{-2/\alpha} u)^{\alpha/2}} du\right) \\
 &\quad \text{where we put } (\beta_{d,uu}(e^t - 1))^{-2/\alpha} u = x, \\
 &\mathcal{L}_{I_u}(\mu_d r^\alpha (e^t - 1)) \\
 &= \exp\left(-\pi\lambda r^2 (\beta_{d,uu}(e^t - 1))^{2/\alpha} \int_0^\infty \frac{1}{1 + x^{\alpha/2}} dx\right). \tag{23}
 \end{aligned}$$

**4.3. Uplink Rate in Full-Duplex.** When a typical UL user  $u_1$  has a distance  $r$  and fading strength  $h_u \sim \exp(\mu_u)$  to its serving BS, user  $u_1$ 's signal power at the serving BS is  $h_u r^{-\alpha}$ . Moreover, user  $u_1$  experiences cumulative interferences  $I_b$ ,  $I_u$ , and  $I_s$ . Here,  $I_b$  is the sum of the received powers from all other BSs, each with distance  $r_i$  and fading strength  $g_i \sim \exp(\kappa_{bb})$  to the serving BS of user  $u_1$ , such that

$$I_b = \sum_{i \in \Phi_b} g_i r_i^{-\alpha}, \tag{24}$$

where  $\Phi_b$  is the set of all interfering BSs. Also,  $I_u$  is the sum of the received powers from all other UL users, each with distance  $r_i$  and fading strength  $\varphi_i \sim \exp(\kappa_{ub})$  to the serving BS of user  $u_1$ , such that

$$I_u = \sum_{i \in \Omega_u} \varphi_i r_i^{-\alpha}, \tag{25}$$

where  $\Omega_u$  is the set of all interfering users. Moreover,  $I_s$  denotes the self-interference by the serving BS of user  $u_1$ . Therefore, the SINR of typical UL user  $u_1$  can be expressed as

$$\gamma_{\text{UL}} = \frac{h_d r^{-\alpha}}{I_b + I_u + I_s + \sigma^2}. \tag{26}$$

Here, we assume that the self-interference  $I_s$  amounts to  $\zeta\sigma^2$  (i.e.,  $\zeta$  times noise power  $\sigma^2$ ). Then, the average rate of a typical UL user  $u_1$  is

$$\begin{aligned}
 R_{\text{UL}}(\lambda, \alpha) &= \mathbf{E}_{I_b, I_u, I_s, r}[\ln(1 + \gamma_{\text{UL}}(I_b, I_u, I_s, r))] \\
 &= \int_{r>0} e^{-\pi\lambda r^2} \\
 &\quad \cdot \int_{t>0} e^{-\mu_u r^\alpha (\sigma^2 + \zeta\sigma^2)(e^t - 1)} \mathcal{L}_{I_b}(\mu_u r^\alpha (e^t - 1)) \\
 &\quad \cdot \mathcal{L}_{I_u}(\mu_u r^\alpha (e^t - 1)) dt 2\pi\lambda r dr. \tag{27}
 \end{aligned}$$

Similar to the downlink analysis, we can obtain the Laplace transform of  $I_b$  and  $I_u$  in this uplink case, as follows:

$$\begin{aligned}
& \mathcal{L}_{I_b}(\mu_u r^\alpha (e^t - 1)) \\
&= \exp\left(-2\pi\lambda \int_r^\infty \left(1 - \frac{1}{1 + \mu_u/\kappa_{bb}(e^t - 1)(r/v)^\alpha}\right) \cdot v dv\right) \quad \text{where we put } \frac{\mu_u}{\kappa_{bb}} = \beta_{u,bb}, \\
& \mathcal{L}_{I_b}(\mu_u r^\alpha (e^t - 1)) \\
&= \exp\left(-2\pi\lambda \int_r^\infty \frac{\beta_{u,bb}(e^t - 1)}{\beta_{u,bb}(e^t - 1) + (v/r)^\alpha} v dv\right) \\
& \quad \text{where we put } \frac{v^2}{r^2} = u, \\
& \mathcal{L}_{I_b}(\mu_u r^\alpha (e^t - 1)) \\
&= \exp\left(-\pi\lambda r^2 \int_1^\infty \frac{1}{1 + ((\beta_{u,bb}(e^t - 1))^{-2/\alpha} u)^{\alpha/2}} du\right) \\
& \quad \text{where we put } (\beta_{u,bb}(e^t - 1))^{-2/\alpha} u = x, \\
& \mathcal{L}_{I_b}(\mu_u r^\alpha (e^t - 1)) = \exp\left(-\pi\lambda r^2 (\beta_{u,bb}(e^t - 1))^{2/\alpha} \cdot \int_{(\beta_{u,bb}(e^t - 1))^{-2/\alpha}}^\infty \frac{1}{1 + x^{\alpha/2}} dx\right), \\
& \mathcal{L}_{I_u}(\mu_u r^\alpha (e^t - 1)) \\
&= \exp\left(-2\pi\lambda \int_r^\infty \left(1 - \frac{1}{1 + \mu_u/\kappa_{ub}(e^t - 1)(r/v)^\alpha}\right) \cdot v dv\right) \quad \text{where we put } \frac{\mu_u}{\kappa_{ub}} = \beta_{u,ub}, \\
& \mathcal{L}_{I_u}(\mu_u r^\alpha (e^t - 1)) \\
&= \exp\left(-2\pi\lambda \int_r^\infty \frac{\beta_{u,ub}(e^t - 1)}{\beta_{u,ub}(e^t - 1) + (v/r)^\alpha} v dv\right) \\
& \quad \text{where we put } \frac{v^2}{r^2} = u, \\
& \mathcal{L}_{I_u}(\mu_u r^\alpha (e^t - 1)) \\
&= \exp\left(-\pi\lambda r^2 \int_1^\infty \frac{1}{1 + ((\beta_{u,ub}(e^t - 1))^{-2/\alpha} u)^{\alpha/2}} du\right) \\
& \quad \text{where we put } (\beta_{u,ub}(e^t - 1))^{-2/\alpha} u = x, \\
& \mathcal{L}_{I_u}(\mu_u r^\alpha (e^t - 1)) = \exp\left(-\pi\lambda r^2 (\beta_{u,ub}(e^t - 1))^{2/\alpha} \cdot \int_{(\beta_{u,ub}(e^t - 1))^{-2/\alpha}}^\infty \frac{1}{1 + x^{\alpha/2}} dx\right).
\end{aligned} \tag{28}$$

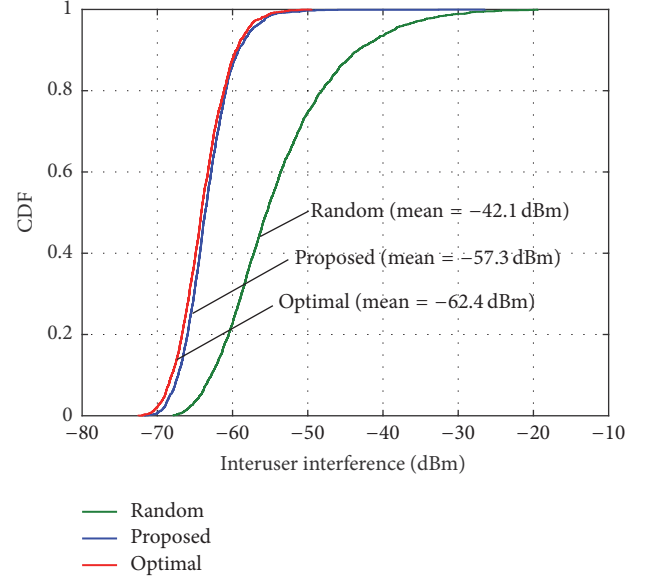


FIGURE 3: Cumulative density function of interuser interference when  $\lambda = 0.5$ .

4.4. *Sum Rate in Full-Duplex.* The average sum rate,  $R_{FD}(\lambda, \alpha)$ , with density  $\lambda$  and path loss exponent  $\alpha$  is summarized as

$$R_{FD}(\lambda, \alpha) = R_{DL}(\lambda, \alpha) + R_{UL}(\lambda, \alpha). \tag{29}$$

## 5. Results and Discussions

For a performance comparison with the proposed algorithms, we consider (i) the conventional *HD mode* in which the BS does not use the FD technique, (ii) the FD with the *random pairing* that determines the user pair randomly, and (iii) the FD with the *optimal pairing* that determines the optimal user pairs through exhaustive search. We observe the system rate by changing the user density  $\lambda$  from 0.1 to 1. For example, the users with  $\lambda = 1$  denote the density of one user per 100 m<sup>2</sup> on average [24]. For simulation, we consider the same network environment as the one considered at the analysis [10]. The same number of DL and UL users is randomly distributed according to the density  $\lambda$  in the multicell network areas. The detailed parameters used for evaluation are listed in Table 1. Here, we obtain the channel parameter from neighbor UL user to UL user,  $\kappa_{uu}$ , by the simulation after applying each of the considered user pairing algorithms.

Figure 3 plots the cumulative distribution function (CDF) of the interuser interference power when  $\lambda = 0.5$ . This interuser interference level is measured by the simulation when each user pairing algorithm is applied. The distribution of interuser interference is an important measure to verify whether the user pairing algorithm works well. Compared to the random pairing, the proposed pairing remarkably reduces the interuser interference. Moreover, there is no significant difference between the proposed pairing and the optimal pairing.

TABLE 1: Parameter Setup.

Parameter	Description	Value
$\lambda$	User density	0.1~1 (default = 0.5)
$\sigma^2$	Noise power	1
$\alpha$	Path loss exponent	4, 3, 2 (default = 4)
$\zeta$	Self-interference ratio	0, 10, 100 (default = 0)
$\mu_d$	Channel parameter from serving BS to DL user	0.1
$\mu_u$	Channel parameter from UL user to serving BS	1
$\kappa_{bu}$	Channel parameter from neighbor BS to DL user	0.1
$\kappa_{ub}$	Channel parameter from neighbor UL user to serving BS	1
$\kappa_{bb}$	Channel parameter from neighbor BS to serving BS	100, 1, 0.1 (default = 100)
$\kappa_{uu}$	Channel parameter from neighbor UL user to UL user	Variable (obtained by simulation)

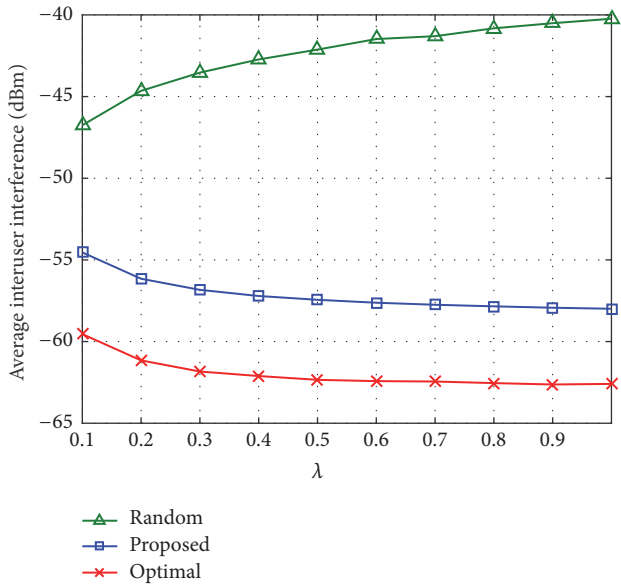
FIGURE 4: Average interuser interference versus user density  $\lambda$ .

Figure 4 shows the average interuser interference level versus the user density  $\lambda$ . In the case of the random pairing, the interuser interference increases gradually as the user density increases. This is because the random pairing algorithm often generates uncontrolled stronger interferences between the randomly selected DL and UL users as the number of users increases. On the other hand, in the proposed pairing algorithm, the interuser interference decreases gradually as the user density increases because the user diversity increases as the number of users increases so that the proposed algorithm has more opportunity to select a DL and UL user pair with less interuser interference. The optimal pairing shows the similar tendency as the proposed one and has a slightly lower interuser interference level.

Figure 5 plots the CDF of the uplink rate, downlink rate, and sum rate when  $\lambda = 0.5$ . In the case of the UL rate, the FD mode has the same performance regardless of pairing algorithms because the user pairing algorithm affects only

the SINR of DL user. However, the UL rate of FD mode is considerably improved compared with that of HD mode. On the other hand, the DL rate is different according to the pairing algorithm. The DL rate is improved in order of the random, proposed, and optimal pairing because their interuser interference level is reduced in this order, as shown in Figure 4. In the case of the sum rate (i.e., sum of UL and DL rates), the similar performance is shown. We can observe that the rate of the proposed pairing algorithm approaches that of the optimal pairing.

Figure 6 shows the average rate versus the user density  $\lambda$ . As the user density increases, the sum rate increases because of the effect of user diversity. In each  $\lambda$ , the UL rates of FD mode are equal regardless of the pairing algorithm. However, the DL rate is different according to the pairing algorithm. The proposed pairing algorithm improves the DL rate greater than the random pairing and approaches the optimal pairing. This verifies the proposed pairing algorithm reduces the interuser interference effectively and so contributes to improve the cell throughput. The performance difference between the proposed pairing and the random pairing increases as  $\lambda$  increases because the proposed pairing algorithm has more opportunities to employ the user diversity but the random pairing does not. Compared with the HD mode, the average gains of FD using the random pairing and proposed pairing algorithms are approximately 180 and 195%, respectively, when  $\lambda = 0.5$ .

Figure 7 shows the average sum rate versus the user density  $\lambda$  according to the variations of related parameters: (a) self-interference ratio ( $\zeta$ ), (b) channel parameter from neighbor BS to serving BS ( $\kappa_{bb}$ ), and (c) path loss exponent ( $\alpha$ ).  $\zeta$  greater than zero means that there remains the self-interference at the BS using FD. Thus, the sum rates of FD mode decrease as  $\zeta$  increases. In addition, the smaller  $\kappa_{bb}$  means that there exists more interference from neighboring BSs because of the imperfect channel estimation. This effect also deteriorates the sum rates of FD mode. When we change the path loss exponent  $\alpha$ , the sum rate decreases as  $\alpha$  decreases and the performance gap between the proposed and random pairing algorithms becomes smaller. This is because the interuser interference increases as the path loss

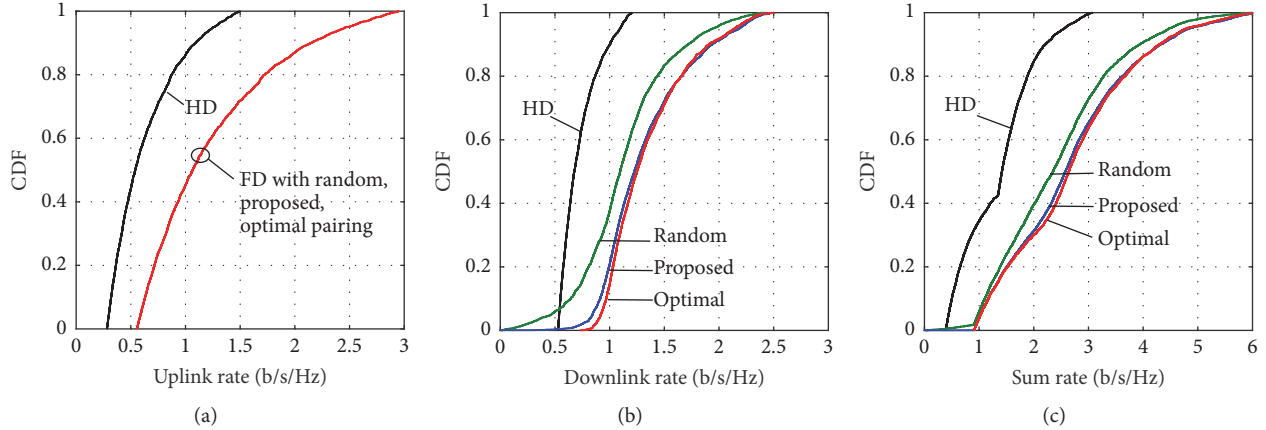


FIGURE 5: Cumulative density function of (a) uplink rate, (b) downlink rate, and (c) sum rate when  $\lambda = 0.5$ .

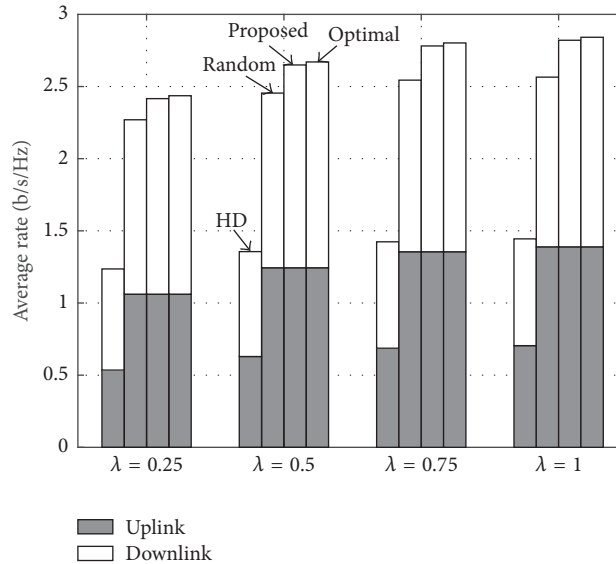


FIGURE 6: Average rate versus user density  $\lambda$ .

between users weakens. These results show that the FD system depends heavily on the interference cancellation capability at the BS and it is appropriate to apply in the channel environment with high attenuation in order to maximize its performance.

## 6. Conclusion

In this study, we investigated the issue of user pairing in FD wireless cellular networks. We proposed a heuristic user pairing algorithm with low complexity to maximize cell throughput. The proposed algorithm was designed such that the DL user having a better signal quality has higher priority to choose its UL user for throughput maximization. Then, we analyzed the performance of the FD cellular network theoretically. The results showed that the proposed user pairing algorithm effectively reduced the interuser interference and achieved near-optimal performance. Thus, the FD system

that uses it considerably outperforms the FD system using a random user pairing and almost doubles the conventional HD system in terms of cell throughput. For further study, we will extend the proposed user pairing algorithm to more general user scheduling algorithms that dynamically allocate radio resources and the transmission direction in an FD cellular network.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea

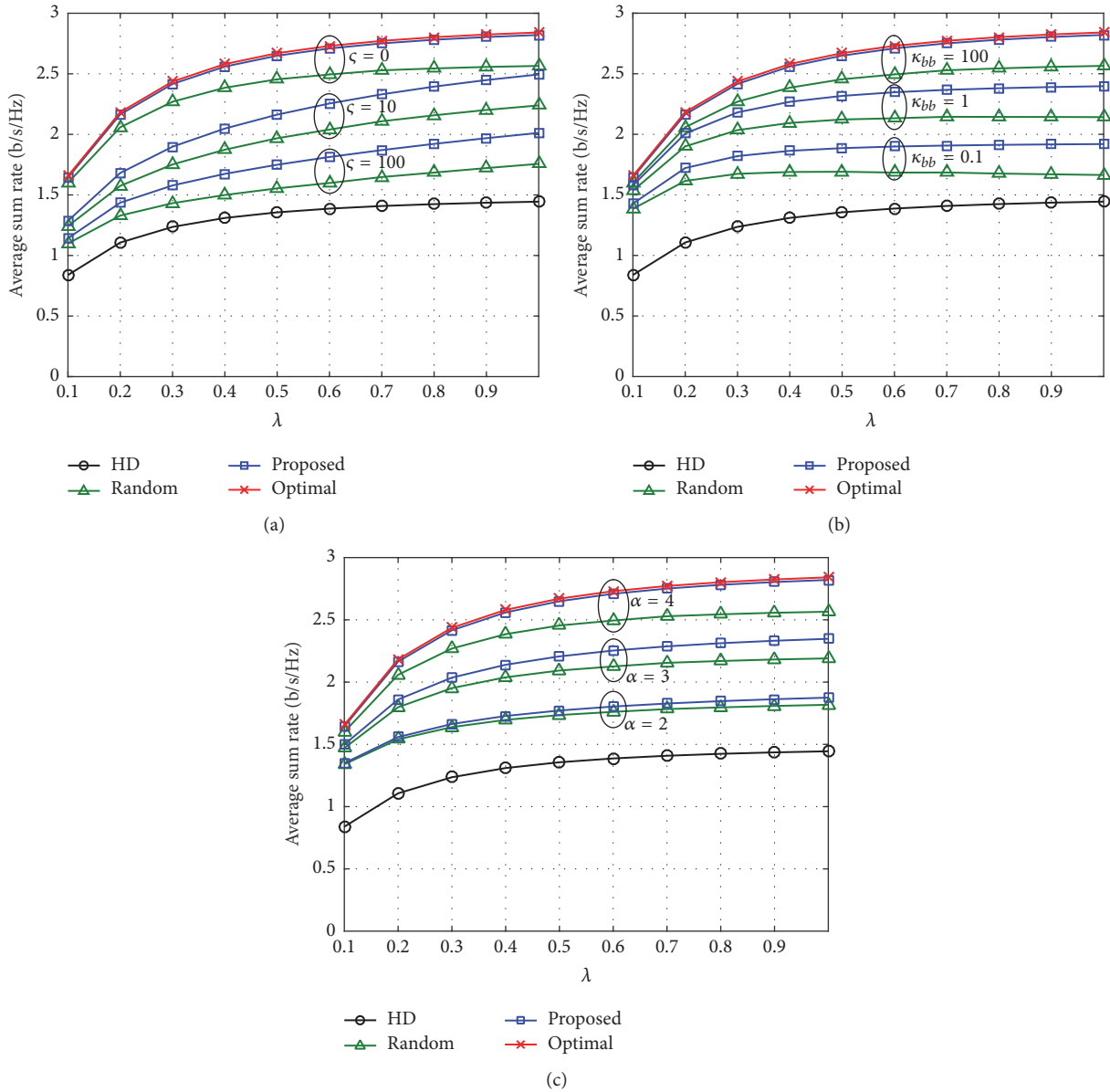


FIGURE 7: Average sum rate versus user density  $\lambda$  according to the variations of (a)  $\zeta$ , (b)  $\kappa_{bb}$ , and (c)  $\alpha$ .

(NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2016R1C1B1016261).

## References

- [1] W. C. Y. Lee, "The most spectrum-efficient duplexing system: CDD," *IEEE Communications Magazine*, vol. 40, no. 3, pp. 163–166, 2002.
- [2] W. C. Y. Lee, "CS-OFDMA: a new wireless CDD physical layer scheme," *IEEE Communications Magazine*, vol. 43, no. 2, pp. 74–79, 2005.
- [3] X. Jin, M. Ma, B. Jiao, and W. C. Lee, "Studies on spectral efficiency of the CDD system," in *Proceedings of the IEEE Vehicular Technology Conference (VTC '09)*, pp. 1–5, Anchorage, Alaska, USA, September 2009.
- [4] J. I. Choi, M. Jain, K. Srinivasan, P. Levis, and S. Katti, "Achieving single channel, full duplex wireless communication," in *Proceedings of the 16th Annual International Conference on Mobile Computing and Networking (MobiCom '10)*, pp. 1–12, IEEE, Chicago, Ill, USA, September 2010.
- [5] D. Bharadia, E. McMillin, and S. Katti, "Full duplex radios," in *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM '13)*, pp. 375–386, Hong Kong, China, August 2013.
- [6] S. Hong, J. Brand, J. Choi II et al., "Applications of self-interference cancellation in 5G and beyond," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 114–121, 2014.
- [7] D. W. K. Ng and R. Schober, "Dynamic resource allocation in OFDMA systems with full-duplex and hybrid relaying," in

- Proceedings of the IEEE International Conference on Communications (ICC '11)*, pp. 1–6, Kyoto, Japan, June 2011.
- [8] D. W. K. Ng, E. S. Lo, and R. Schober, “Dynamic resource allocation in MIMO-OFDMA systems with full-duplex and hybrid relaying,” *IEEE Transactions on Communications*, vol. 60, no. 5, pp. 1291–1304, 2012.
- [9] W. Cheng, X. Zhang, and H. Zhang, “Full/half duplex based resource allocations for statistical quality of service provisioning in wireless relay networks,” in *Proceedings of the IEEE INFOCOM*, pp. 864–872, IEEE, Orlando, Fla, USA, March 2012.
- [10] X. Shen, X. Cheng, L. Yang, M. Ma, and B. Jiao, “On the design of the scheduling algorithm for the full duplexing wireless cellular network,” in *Proceedings of the IEEE Global Communications Conference (GLOBECOM '13)*, pp. 4970–4975, IEEE, Atlanta, Ga, USA, December 2013.
- [11] Z. Liu, Y. Liu, and F. Liu, “Joint resource scheduling for full-duplex cellular system,” in *Proceedings of the 22nd International Conference on Telecommunications (ICT '15)*, pp. 85–90, Sydney, Australia, April 2015.
- [12] M. Al-Imari, M. Ghorraishi, P. Xiao, and R. Tafazolli, “Game theory based radio resource allocation for full-duplex systems,” in *Proceedings of the 81st IEEE Vehicular Technology Conference (VTC Spring '15)*, pp. 1–5, Scotland, UK, May 2015.
- [13] Y. Sun, D. W. K. Ng, and R. Schober, “Multi-objective optimization for power efficient full-duplex wireless communication systems,” in *Proceedings of the 58th IEEE Global Communications Conference (GLOBECOM '15)*, pp. 1–6, San Diego, Calif, USA, December 2015.
- [14] D. Kim, H. Lee, and D. Hong, “A survey of in-band full-duplex transmission: from the perspective of PHY and MAC layers,” *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2017–2046, 2015.
- [15] J. Y. Kim, O. Mashayekhi, H. Qu, M. Kazandjieva, and P. Levis, “Janus: a novel MAC protocol for full duplex radio,” Tech. Rep., Stanford University, 2013.
- [16] Q. Gao, G. Chen, L. Liao, and Y. Hua, “Full-duplex cooperative transmission scheduling in fast-fading MIMO relaying wireless networks,” in *Proceedings of the International Conference on Computing, Networking and Communications (ICNC '14)*, pp. 771–775, Honolulu, Hawaii, USA, February 2014.
- [17] G. Yu, D. Wen, and F. Qu, “Joint user scheduling and channel allocation for cellular networks with full duplex base stations,” *IET Communications*, vol. 10, no. 5, pp. 479–486, 2016.
- [18] H.-H. Choi, “On the design of user pairing algorithms in full duplexing wireless cellular networks,” in *Proceedings of the 5th International Conference on Information and Communication Technology Convergence (ICTC '14)*, pp. 490–495, Busan, Korea, October 2014.
- [19] A. Sarkar, P. P. Chakrabarti, and R. Kumar, “Frame-based proportional round-robin,” *IEEE Transactions on Computers*, vol. 55, no. 9, pp. 1121–1129, 2006.
- [20] M. Jain, J. Choi, T. Kim et al., “Practical, real-time, full duplex wireless,” in *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking (ACM MobiCom '11)*, pp. 301–312, Las Vegas, Nev, USA, September 2011.
- [21] H.-H. Choi and J.-R. Lee, “Distributed transmit power control for maximizing end-to-end throughput in wireless multi-hop networks,” *Wireless Personal Communications*, vol. 74, no. 3, pp. 1033–1044, 2014.
- [22] J. G. Andrews, F. Baccelli, and R. K. Ganti, “A tractable approach to coverage and rate in cellular networks,” *IEEE Transactions on Communications*, vol. 59, no. 11, pp. 3122–3134, 2011.
- [23] T. D. Novlan, H. S. Dhillon, and J. G. Andrews, “Analytical modeling of uplink cellular networks,” *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2669–2679, 2013.
- [24] S. Goyal, P. Liu, S. Hua, and S. Panwar, “Analyzing a full-duplex cellular system,” in *Proceedings of the 47th Annual Conference on Information Sciences and Systems (CISS '13)*, pp. 1–6, IEEE, Baltimore, Md, USA, March 2013.



## Research Article

# Network Intelligence Based on Network State Information for Connected Vehicles Utilizing Fog Computing

**Seongjin Park and Younghwan Yoo**

*Department of Electrical and Computer Engineering, Pusan National University, Busandaehak-ro 63beon-gil, Geumjeong-gu, Busan 46241, Republic of Korea*

Correspondence should be addressed to Younghwan Yoo; ymomo@pusan.ac.kr

Received 9 December 2016; Accepted 30 January 2017; Published 20 February 2017

Academic Editor: Yujin Lim

Copyright © 2017 Seongjin Park and Younghwan Yoo. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a method to take advantage of fog computing and SDN in the connected vehicle environment, where communication channels are unstable and the topology changes frequently. A controller knows the current state of the network by maintaining the most recent network topology. Of all the information collected by the controller in the mobile environment, node mobility information is particularly important. Thus, we divide nodes into three classes according to their mobility types and use their related attributes to efficiently manage the mobile connections. Our approach utilizes mobility information to reduce control message overhead by adjusting the period of beacon messages and to support efficient failure recovery. One is to recover the connection failures using only mobility information, and the other is to suggest a real-time scheduling algorithm to recover the services for the vehicles that lost connection in the case of a fog server failure. A real-time scheduling method is first described and then evaluated. The results show that our scheme is effective in the connected vehicle environment. We then demonstrate the reduction of control overhead and the connection recovery by using a network simulator. The simulation results show that control message overhead and failure recovery time are decreased by approximately 55% and 5%, respectively.

## 1. Introduction

New IT technologies have recently been converged for a new platform, due to the prevalence of smartphones, formidable growth in mobile traffic, and the emergence of IoT services. Recent research has several common themes. First, it considers large numbers of wireless mobile devices, of many types (e.g., portable smart devices, cars, drones, and mobile sensors). Second, besides data collection in the heterogeneous communication environment, it emphasizes intelligent management and utilization of data in accordance with specific services. This work encompasses new technologies including fog computing, Software Defined Networks (SDN), and Fifth-Generation (5G) networks.

The term “fog computing” was coined by Cisco Systems in 2012. Cloud computing is a technology where users can take IT resources (a remote server, communication systems, storage, services, etc.) anywhere. However, cloud computing cannot provide real-time services, which require less than

1 ms delay of a 5G requirement because of the physical distance between users and a remote server. On the other hand, fog computing can provide more practical 5G services, by deploying servers nearer to the end user. The fog metaphor comes from the idea that it utilizes small “close to the ground” servers, as opposed to more distant servers in cloud computing [1]. In fog computing, cloud servers are located near the end user.

Fog computing adds a fog layer to the cloud computing between the cloud server and the end user. Access points, base stations, routers, and mobile devices can serve as fog servers. Mobile devices can overcome their lack of adequate IT resources by using fog computing. Besides, it is possible to take advantage of mobility pattern of mobile devices. Examples of applications and services that can utilize fog computing include connected vehicles (CV), smart grid, wireless sensor and actuator networks (WSAN), augmented reality (AR), content delivery networks (CDN), and others that require real-time and big data [2]. Communication

between a fog server and a mobile device can use various wireless technologies (WiFi, Bluetooth, LTE, and 5G) depending on the service type.

Efficient resource management is essential but difficult, because of the inflexibility of the current network structure. One switch or router has to perform all functions, including forwarding, routing, and management of network resources. Each switch or router performs these tasks consecutively. These hardware-dependent features cause inefficient resource utilization with time-varying network state. In order to modify some network functions, it takes too much time to correct all the switch or router which is incompatible with each vendor-specific device. In addition, this can result in temporary outages.

SDN was introduced in order to overcome these drawbacks of the current network architecture [3]. It represents a paradigm shift from hardware-dependent to software defined architectures. Its main benefit is to decouple the control plane from the data plane. In the past, the global Internet had to be realized as a distributed system. High data rates in the backbone network now make it possible to centralize the networks control system. Thus, a central controller can manage the overall network state and install forwarding rules in each network device. To achieve this, it is necessary to separate hardware from software functions in vendor-specific devices. Network devices simply forward incoming packets according to a flow table installed by the controller. Other functions, such as routing path calculation and global resource management, are performed by the controller. In this respect, fog computing applies the SDN architecture as a network perspective.

The SDN is one of the indispensable components of the 5G network, which has to handle high volumes of mobile traffic and support low latency communication. Because one technology cannot satisfy all user demands, it is important for emerging technologies to cooperate with existing infrastructure for seamless services. From this perspective, the centralized resource management of SDN brings about efficiency benefits in multiservice and multitechnology environments [4]. Then, the fog computing may enhance the benefits of SDN since the fog layer reduces the distance between mobile devices and servers, resulting in the improvement of communication latency.

In this paper, we apply our ideas to the field of connected vehicles which has attracted much attention these days. Cars are now smart devices, with numerous sensors and communication capabilities. Vehicles are less energy-constrained and can have higher communication capacity than smartphones. Moreover, there are many challenges to find new applications concerning the devices mobility. The connected vehicle is an emerging representative application. A vehicle is not only a mobile node but also a forwarding device which can act as infrastructure, like a base station or a RSU (Roadside Unit) [5].

Services in connected vehicles are classified as being safety-related or non-safety-related. A safety-related service is critical for safe driving. A non-safety-related service is not critical but requires a high data rate to ensure a seamless user experience. For example, a self-driving car must promptly

and accurately collect and analyze data from its own embedded sensors, from neighboring cars, and from infrastructure, in order to maintain safety. In this paper, we consider the connected vehicle as an application that takes advantage of fog computing and SDN. However, the proposed ideas are not limited to only the connected vehicle.

In this paper, we categorize mobility information into three classes and measure the signal strength of the link between the controller and each switch in order to help the controller supervise global network resources. We further propose an intelligent maintenance method utilizing network information and give some practical use cases for connected vehicles.

## 2. Related Work

*2.1. Mobile SDN.* First, we introduce recent research related to SDN in the mobile environment. Software Defined Wireless Network (SDWN) [6] is the first SDN study using the IEEE 802.15.4 (Low Rate Wireless Personal Area Networks) standard. This architecture decouples a sink node from a general node. This work consider only wireless channel but not node mobility.

Software Defined Mobile Networks (SDMN) [7] consider the mobile environment in the Radio Access Network (RAN) and the core network. Ku et al. [8] applied SDN to Mobile Ad Hoc Networks (MANETs) and Vehicular Ad Hoc Networks (VANETs). The major difference from the precedent researches is the wireless communication between the controller and switches. In addition, moving nodes add complexity. The LTE link is used for control plane communications, and the WiFi link is used by the data plane. An SDN wireless node simultaneously plays the roles of a router and a host. To obtain the current topology, each SDN wireless node sends periodic beacon messages. The nodes respond with connection information, which the controller maintains in a table. Because of the nature of wireless communications, each switch must prepare for link failure between the controller and switches. In the general case, SDN switches forward incoming packets to a corresponding output port in accordance with an installed flow table; similarly, an SDN wireless node forwards packets to an appropriate wireless interface or channel in MANET. In other words, the role of the port in the SDN switch is replaced by the wireless interface or the frequency in the SDN wireless node.

*2.2. Fog Computing.* Fog SDN (FSDN) [9] is an architecture that combines fog computing and SDN in VANETs. In a VANET, mobile nodes move faster than those in a MANET, and their mobility pattern tends to follow fixed roadways. In addition to the SDN controller and the SDN wireless node used in an SDN-based MANET, an FSDN includes an SDN Roadside Unit (RSU), an SDN Roadside Unit Controller (RSUC), and a base station as a fog device. All fog devices are controlled by the SDN controller. The SDN controller knows the global state of the network and assigns tasks at a service level to efficiently utilize limited resources. Acting as a fog device, an SDN RSU, an SDN RSUC, or a base station can supply low latency services to mobile devices, by

using attributes of each device location and mobility pattern. Examples are given for a safety-related lane-change service and a non-safety-related data streaming service.

Vehicular Fog Computing (VFC) [5] is an architecture which treats vehicles as infrastructure. The main difference between VFC and Vehicular Cloud Computing (VCC) is that the former can support real-time services and geolocation-based distributed services, due to the proximity of mobile users to fog servers. Using vehicles as infrastructure in VFC avoids the additional cost of dedicated infrastructure. This work distinguishes between moving cars and parking cars in the urban environment and divides computational and communication resources in accordance with their usage. For example, rapidly moving cars are preferred for relaying data to a distant location, while slowly moving cars, passing through a congested area, are favorable as localized computational resources. Parked cars are used as infrastructure (e.g., RSU, RSUC, or base station).

Previous research has proposed architectures applying SDN and fog computing to MANET or VANET. However, these works do not sufficiently utilize mobility information. SDN wireless nodes simply inform the fog server or the SDN controller periodically about their current connection to neighboring nodes. We propose a method of managing and utilizing mobility information in the connected vehicle.

### 3. The Proposed Idea

The concept of SDN can facilitate easy and flexible management of a network. It is essential in the network layer of the fog computing. The SDN architecture consists of three layers: application layer, network controller layer, and infrastructure layer. The infrastructure layer is responsible for packet forwarding; and the network controller layer collects network state information and provides the API to control forwarding devices in the infrastructure layer. Finally, the application layer decides the forwarding policy using network intelligence. The layers communicate with each other through open interfaces. A northbound API provides the interface between application layer and network controller layer. A southbound API is utilized for the communication between network controller and infrastructure layer. OpenFlow [10] is a common example of a southbound API. Several northbound APIs are being standardized.

Then, there are some challenges to run an SDN controller on fog servers. Since the fog servers maintain mobile nodes as the clients, the SDN controller should also consider the nodes mobility information and the vulnerable nature of communications in the mobile environment. To do this, we propose a management method of the network state information at first and then suggest some use cases.

**3.1. Management of Network State Information.** The controller maintains network state information (NSI) to manage the network. Node mobility information is especially important in unpredictable environments, such as connected vehicles. Previous research [8] only utilizes the nodes connection state (i.e., up or down) in a mobility table. We augment this with the Received Signal Strength Indicator (RSSI) to

measure the channel quality. It is also desirable to exploit node mobility patterns. This paper identifies three types of mobility pattern: nearly stationary nodes, nodes that move in arbitrary patterns, and nodes which move in predictable patterns. By categorizing mobility patterns, the controller can coordinate the overall network from a global perspective.

Figure 1 shows an example of the system in a campus scenario. A school bus is an example of a moving node which has a predictable mobility pattern. Similarly, a moving car and a parked car represent an unpredictable moving node and a quasi-stationary node, respectively. The controller manages the parameters shown in the following part.

#### Structure of NSI

$V$ : Set of vehicles,  $V = \{v_1, v_2, \dots, v_{|V|}\}$

$N_i$ : A list of neighbor nodes,  $N_i \subset V$

$N_p$ : The current position of the node,  $N_p = (x, y, z)$

$N_m$ : A class of the node's mobility,  $N_m = \{r, s, p\}$

$N_r$ : RSSI information of the node,  $N_{r,th}$ : threshold value

$N_f$ : Failure type of the node,  $N_f = \{t, s\}$

$N_d$ : Distance from the controller,  $N_c$ : max distance

The parameter  $N_m$  denotes the mobility class of vehicles, where the value  $r$  indicates a vehicle which has random movement,  $s$  is a quasi-stationary vehicle, and  $p$  is a vehicle which moves in a predictable pattern.

**3.2. Adaptive Control Messages.** Mobility information is used to reduce control message overhead. Each wireless node sends and receives beacon messages periodically, in order to maintain recent connection information. In mobile networks with dynamic topologies, the number of control messages increases with the number of nodes. This scalability problem can be addressed by leveraging node mobility patterns. If nodes mobility pattern is quasi-stationary or predictable, we can reduce the reporting frequency, because the controller can predict the nodes position. On the other hand, nodes that have an unpredictable mobility pattern must report their mobility information frequently in order to maintain the current network topology.

**3.3. Connection Recovery Process.** NSI is also useful for recovery. First, we consider connection failure between the fog server and the wireless nodes. Since the link between the nodes and the connected vehicle is wireless, the networks suffer from frequent disconnection, which diminishes overall network performance. Prior research [8] suggests recovery by reverting to routing policies such as Ad Hoc On-Demand Distance Vector Routing (AODV), or Dynamic Source Routing (DSR). Although this is a very easy and simple solution, it is not sufficient for the intelligent network. Due to the unstable communication environment in the connected vehicle, networks must predict and cope with the communication interruption. We propose that the controller predict whether the connection between a fog server and a

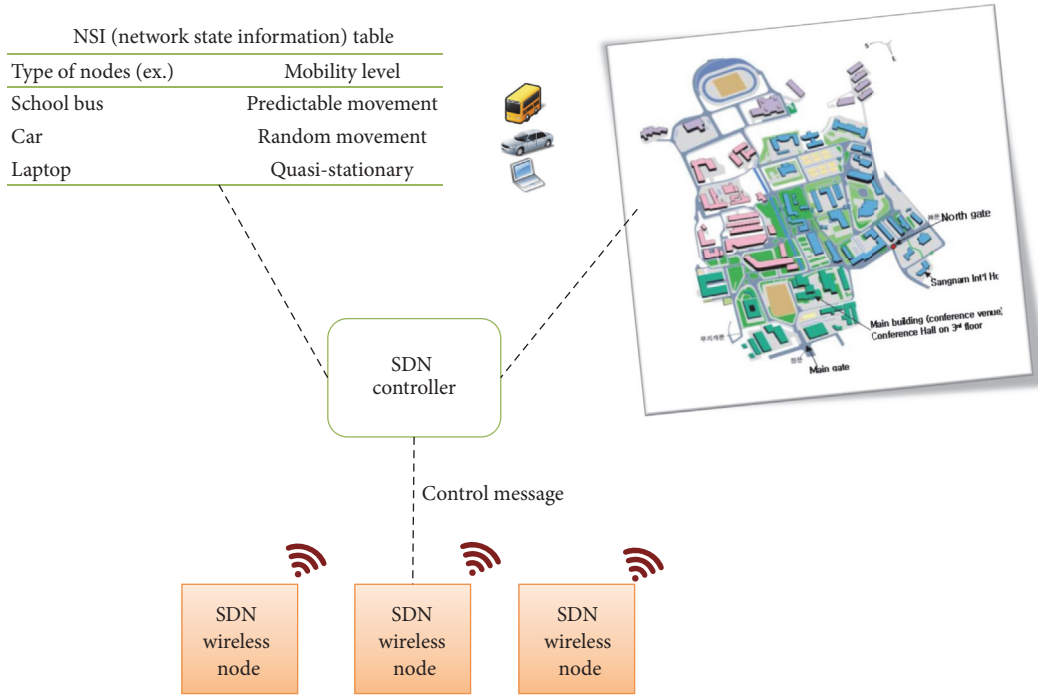


FIGURE 1: Overall system overview.

```

Begin
while  $N_r \leq N_{r,th}$  do
  if  $N_d \leq N_c$  then
     $N_f == t$                                 ▷ Temporal Failure
  else
    if  $N_m == r$  then
       $N_f == s$                                 ▷ Severe Failure
    else if  $N_m == s \parallel N_m == p$  then
      Predicts the node's upcoming position
      if  $N_d \leq N_c$  then
         $N_f == t$                                 ▷ Temporal Failure
      else
         $N_f == s$                                 ▷ Severe Failure
      end if
    end if
  end if
end while
End

```

ALGORITHM 1: Failure decision process.

wireless node will be lost, by using the nodes mobility pattern and the link quality. The controller also decides whether a disconnected nodes link will recover soon or not. Algorithm 1 shows the controllers decision process. When the controller forecasts a connection loss, it is classified as either a temporal or a severe failure. The recovery process then proceeds as shown in Algorithm 2. In a predicted temporal failure, the disconnected node simply waits until the link recovers. If a severe connection loss is predicted, the node ceases to follow the existing forwarding table and performs its own routing

policy in advance of the disruption. For example, when the controller predicts that a lost connection node will undergo a failure for a short time, returning to the existing routing protocol is inefficient from the perspective of the network resource utilization.

*3.4. Server Failure Recovery Process.* We further propose a recovery process from fog server failure. The vehicle that was connected to the failed fog server cannot continue to use its services until the connection is resumed. To offer

```

Begin
if  $N_f == t$  then
    Stay until recover the connection
else if  $N_f == s$  then
    Delete the current flow table & Run an existing routing protocol
end if
End

```

ALGORITHM 2: Recovery process.

seamless services, a fast fail-over scheme is needed. The fast fail-over scheme can be regarded as an optimization problem. Its objective is to find the maximum number of services that can be recovered while meeting the delay requirement of each service. We define this problem as a real-time scheduling optimization problem in Section 4. Then, we suggest our scheduling algorithm for the mobile environment like Figure 1. The simulation result of the control overhead and the connection recovery process is shown in Section 5.

#### 4. Analysis

We formulate the connected vehicle environment using set notation (see the following part).

##### Summary of Symbols

- $V$ : Set of vehicles,  $V = \{v_1, v_2, \dots, v_{|V|}\}$
- $S$ : Set of services,  $S = \{s_1, s_2, \dots, s_{|S|}\}$
- $F$ : Set of fog servers,  $F = \{f_1, f_2, \dots, f_{|F|}\}$
- $R_{f_j}$ : Set of vehicles which are located in coverage of a fog server  $f_j$ ,  $R_{f_j} \subset V$
- $SQ_{f_j}$ : Set of services that will be treated by a fog server  $f_j$ ,  $SQ_{f_j} = \{sq_{f_j,1}, sq_{f_j,2}, \dots, sq_{f_j,|SQ_{f_j}|}\}, \forall sq_{f_j,k} \in S$
- $L_{v_i}$ : A timer of a vehicle  $v_i$  for the current service,  $L_{v_i}(t) = \alpha_{v_i} A_{v_i}, L_{v_i}(t+1) = L_{v_i}(t) - 1$
- $E_{v_i}$ : The expected delay of a current service for vehicle  $v_i$ ,  $E_{v_i} = H_{v_i} A_{v_i}$

Because of the dynamic nature of the mobile environment, elements of some sets change over time. For example, the elements of  $R_{f_i}(t_m)$  are different from  $R_{f_i}(t_n)$ ; that is,  $R_{f_i}(t_m) \neq R_{f_i}(t_n)$ , where  $m \neq n$ . Each vehicle requires services out of  $S$ , according to a Poisson process. The services provided by fog servers are typically location sensitive, for example, driving information such as local traffic congestion information or accident reports. On the other hand, cloud servers typically support entertainment services like multimedia streaming. Fog servers maintain a service queue to provide services to vehicles which are located in their local communication area.

When a fog server has unexpectedly failed, the controller recognizes the failure situation and performs the recovery process. The controller informs the neighboring fog servers, providing them with the service queued at the failed fog

server at the time of failure. The neighboring fog servers then reschedule their service queues to recover from the failure. If vehicles are not located in the coverage area of any neighboring fog servers, boundary vehicles between the two neighboring fog servers serve as relay nodes.

*4.1. Problem Description.* The objective function of the recovery process is shown in (1). Its purpose is to maximize the number of services provided to vehicles, while satisfying the delay requirement of each service, until the end of recovery time  $t_e$ .

$$\underset{V,S,F}{\text{minimize}} \quad \sum_{t=t_0}^{t_e} \sum_{i=0}^{|V|} I_{v_i}(t). \quad (1)$$

$I_{v_i}(t)$  is a binary variable which can have one value out of 1 and 0. It denotes whether the delay requirement of a service for vehicle  $v_i$  is met or not at time  $t$ . A vehicle sets a timer  $L_{v_i}$  at the time at which it requests a service. As in (2),  $I_{v_i}(t)$  is changed to 1 when the timer expires. Therefore, the minimization of the total sum of  $I_{v_i}(t)$  in (1) is equal to the maximization of the number of services vehicles can use.

$$I_{v_i}(t) = \begin{cases} 1, & \text{If } L_{v_i}(t) = -1, \\ 0, & \text{Otherwise.} \end{cases} \quad (2)$$

Equation (3) is a constraint to decide the candidate services to be recovered from the fog server failure. This means the service whose timer should be greater than the expected delay  $E_{v_i}$  at time  $t$ . The notations  $E_{v_i}$  and  $L_{v_i}$  vary with a specific environment and the service type for each vehicle. The values of these two notations are decided by several determinants. First,  $E_{v_i}$  is decided by two factors,  $H_{v_i}$  and  $A_{v_i}$ , hop counts and delay of a service in one hop distance.  $H_{v_i}$  is set to 1 if vehicles are in the coverage of the fog server. Otherwise, vehicles need a hop count greater than 1 to communicate with the available fog server via multihop. In addition, the hop count is influenced by the origin of the service, which can be a fog server or a cloud server. In other words, if a current service for a vehicle is not cached at the fog server, the service is retrieved from the cloud server, resulting in the increase of  $H_{v_i}$ . Because  $A_{v_i}$  is a services delay in one hop distance, we multiply  $A_{v_i}$  by  $H_{v_i}$ . Second, an initial value of the timer  $L_{v_i}$  is a threshold of a services delay that users can allow. Thus, this timer is initialized by the multiplication of the one hop delay  $A_{v_i}$  and the allowable hop count  $\alpha_{v_i}$ .

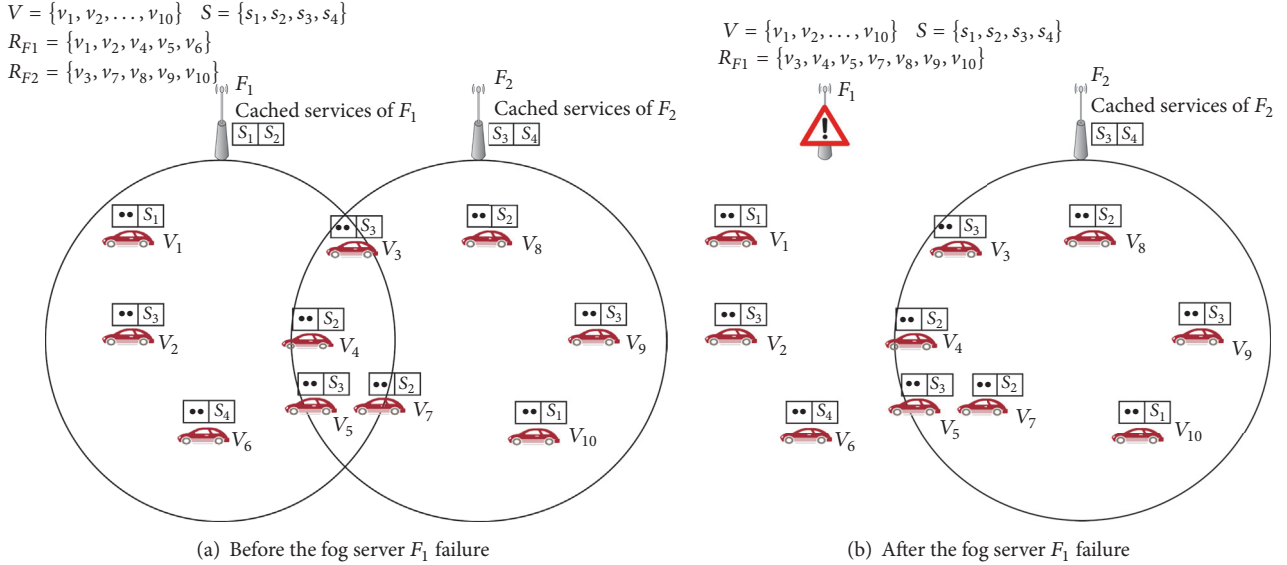


FIGURE 2: An example of the recovery process.

For the performance comparison in Section 5,  $\alpha_{v_i}$  is randomly selected.

$$\text{subject to } L_{v_i}(t) - E_{v_i}(t) > 0, \quad E_{v_i}(t) > 0. \quad (3)$$

**4.2. Example of Recovery Process.** Figure 2 depicts an example of the recovery process. There are ten vehicles, four available services, and two fog servers. Each fog server is providing services to vehicles which are located in its communication coverage, as shown in Figure 2(a). Figure 2(b) depicts the situation when fog server  $F_1$  fails. We identify two recovery cases. Any lost connections that can be recovered by connecting to a neighboring fog server are taken over by that fog server. In the example, when  $F_1$  fails, vehicles  $v_4$  and  $v_5$  are served instead by fog server  $F_2$ . Any vehicles which cannot connect directly to another fog server must depend on neighboring vehicles to relay the connection to a neighboring fog server. In the example, vehicles  $v_1$ ,  $v_2$ , and  $v_6$  are needed to receive relayed data from boundary vehicles  $V_3$ ,  $V_4$ , and  $v_5$ .

Each fog server makes a table to decide the priority of the services in the recovery situation. When the controller informs a certain fog server failure, neighboring fog servers start the recovery process by rescheduling queues for all the services including the broken fog servers services. Table 1 represents the service queues of fog server  $F_2$  just after fog server  $F_1$  fails. To make effective scheduling, the fog server maintains several attributes such as service ID, vehicle ID, service type, timer, and expected delay. First, the service types are divided into two cases according to which servers are responsible for offering a service to a vehicle. If a cloud server is responsible for providing a certain service, the service type is marked as  $C$ . On the other hand, the service type is marked as  $F$  when a fog server offers a service. Second, each of these two cases,  $C$  and  $F$ , is divided into two subcases, depending on whether the service is relayed by other vehicles or not. For instance, if a service is originated from a cloud server

TABLE 1: An example of the fog server's service queue.

Vehicle ID	Service ID	Service type	Timer	Expected delay
$v_1$	$s_1$	$C, V$	9	7
$v_2$	$s_3$	$C, V$	18	6
$v_3$	$s_3$	$F$	10	2
$v_4$	$s_2$	$C$	18	10
$v_5$	$s_3$	$F$	16	2
$v_6$	$s_4$	$F, V$	15	15
$v_7$	$s_2$	$C$	12	10
$v_8$	$s_2$	$C$	14	10
$v_9$	$s_3$	$F$	18	2
$v_{10}$	$s_1$	$C$	9	5

and it is relayed by not any infrastructure node but another vehicle, the service is marked as  $C$  and  $V$  at the same time. By utilizing these attributes that the fog server maintains, we apply a real-time scheduling scheme to find the optimal ordering of services.

**4.3. The Proposed Real-Time Scheduling Algorithms.** Real-time scheduling has been studied for many fields of application, especially in MANET. Because the connected vehicle environment is similar to MANET in terms of dynamic features of its mobile environment, the proposed method is modified from the precedence real-time scheduling algorithms [11, 12]. One of such algorithms is Earliest Deadline First (EDF), which is a dynamic priority algorithm. When the fog server must decide the next order of service, it uses EDF to select the service for which the deadline is soonest. Another reasonable algorithm is Most Requested First (MRF). When applying the MRF, the fog server schedules services in the order of frequency of their request. Schedules are maintained before the start of the recovery process.

In order to make optimal scheduling, we propose a real-time scheduling algorithm that exploits precedence algorithms and attributes in the fog servers service queues. By doing this, we can obtain near-optimal results with the various requirements of delay. First, we adopt the EDF algorithm to serve as many services as possible until the recovery process is completed. We then add a decision method which selects a service according to its service type and one hop delay of a service,  $A_{v_i}$ . If the service type is marked as  $F$  and the value of  $A_{v_i}$  is small, the service can be provided to users in a short time. When the delay requirement of most services is tight, giving a high priority to a service which has the shorter service time is rational since it can maximize the total number of services meeting the deadline. Generally, the services provided by a fog server have shorter service time than the services from a cloud server; thus, the services from a fog server have higher priority in the recovery process. This policy makes sense in the respect that the service by the fog server is more critical than the service by the cloud server. The information for safe driving is provided from the fog server.

### 5. Simulation

We performed the simulation to evaluate the proposed idea in two cases. First, the control overhead and the connection recovery are evaluated in the connected vehicle environment. Then, the recovery from the fog server failure is evaluated as a numerical result in Table 3.

We constructed the campus network shown in Figure 3, using the Simulation of Urban Mobility (SUMO) [13] simulator. The mobile nodes drive along the black line according to their routing path. We generated nodes that have one mobility pattern out of the three patterns in Figure 1: random, predictable, and quasi-stationary. The beacon message period of each node is based on its mobility pattern; that is, nodes having random movement patterns send beacon messages more frequently than the others. We assume that the controller knows all the nodes mobility pattern, while Ku et al. [8] make all SDN wireless nodes send periodic beacon messages. Because our fog servers are controlled by an SDN controller, we utilized the Mininet-WiFi [14] simulator which considers both the SDN and the mobile environment.

**5.1. Control Message Overhead.** The result for the control overhead is shown in Figure 4. The control overhead consists of two types of messages: first the beacon messages that each mobile node periodically informs to an SDN controller and second the messages exchanged between mobile nodes whenever they meet each other, in order to collect neighboring information. This neighboring information is included in the NSI of each node, and this NSI is contained in the beacon message reported from each node to the SDN controller. These two types of control messages were considered in the simulation. The ratio of nodes having the random, predictable, and quasi-stationary mobility pattern is 1:1:1. Our method reduces the number of control messages by approximately 55% as compared with Ku et al.'s method [8], while still maintaining fresh topology. This is because the

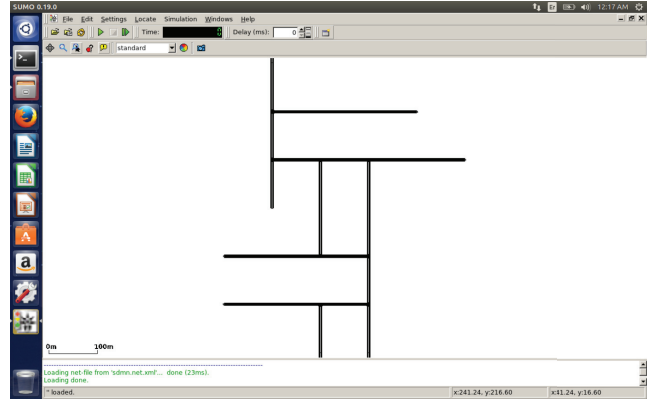


FIGURE 3: Building a campus network.

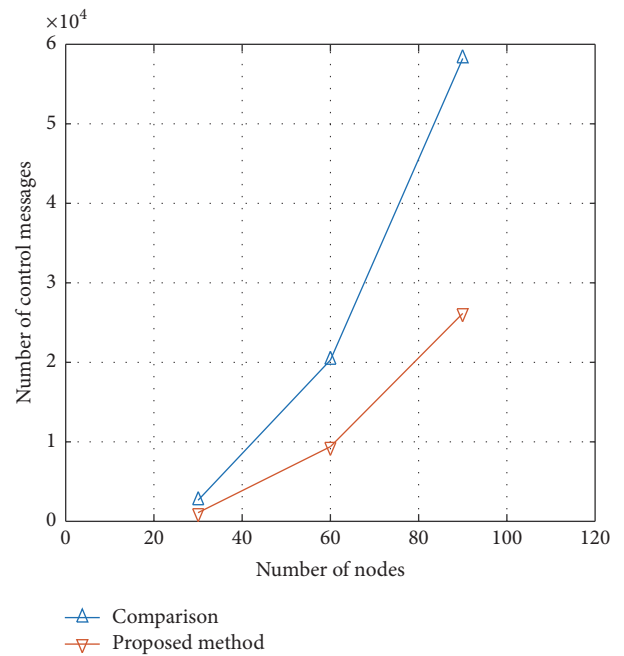


FIGURE 4: Control overhead.

controller can reduce each node control message period if it can predict its next location.

Figure 5 depicts the control overhead with different ratios of three mobility pattern nodes: random, predictable, and quasi-stationary. The ratios are set to 2:1:1, 1:2:1, and 1:1:2, respectively. The number of control messages is the greatest when the ratio is 1:2:1. This is because the nodes with predictable movement have a higher probability of encountering other nodes while driving along the regular routing path. In our simulation, the nodes with predictable movement are generally public buses. They move along the main road in the campus, so they can meet more vehicles than other types of nodes. On the other hand, because stationary nodes meet the least number of other nodes, the number of control messages is the lowest when the ratio is 1:1:2. Relative to the comparison, our method reduced control message

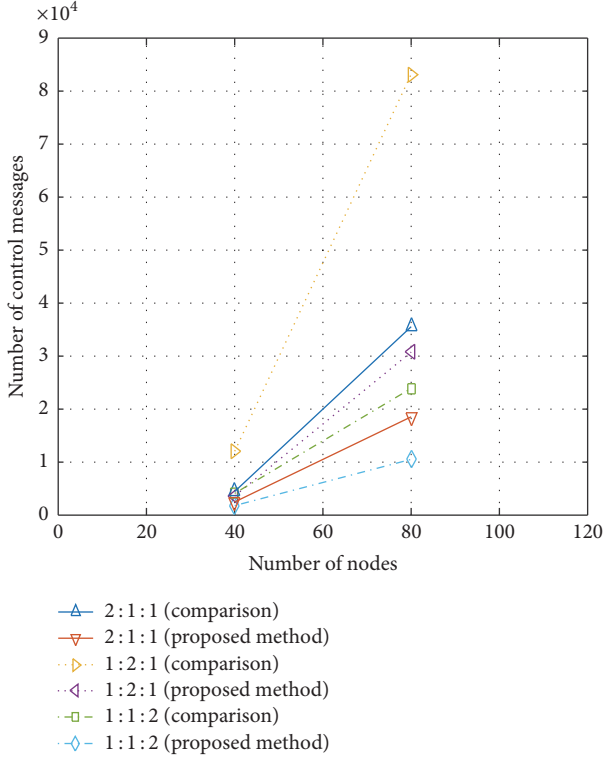


FIGURE 5: Control overhead in various environments.

TABLE 2: Evaluation parameters.

Parameter	Value
Hop count from the fog server	1
Hop count from the cloud	5
Expected increasing hop count (random mobility)	4
Expected increasing hop count (predictable mobility)	2
Size of services of $S$	{1, 2, 2, 3}

TABLE 3: The number of successfully delivered services.

	EDF	MRF	Proposed algorithm	Optimal
Rigid delay	5	7	7	7
Sufficient delay	9	8	9	9

overhead by 64%, 56%, and 47%, for mobility pattern ratios of 1:2:1, 1:1:2, and 2:1:1.

**5.2. Connection Lost Time.** The duration while the connection is lost is compared in Figure 6. As the number of nodes grows, the total failure time also increases. The result shows that the connection lost time is reduced by about 5% relative to the comparison method. Furthermore, the proposed method avoids wasting resources by just waiting for a while until the connection is automatically recovered, not starting a new routing process. It can predict when the connection will be automatically recovered using the information of the node moving pattern.

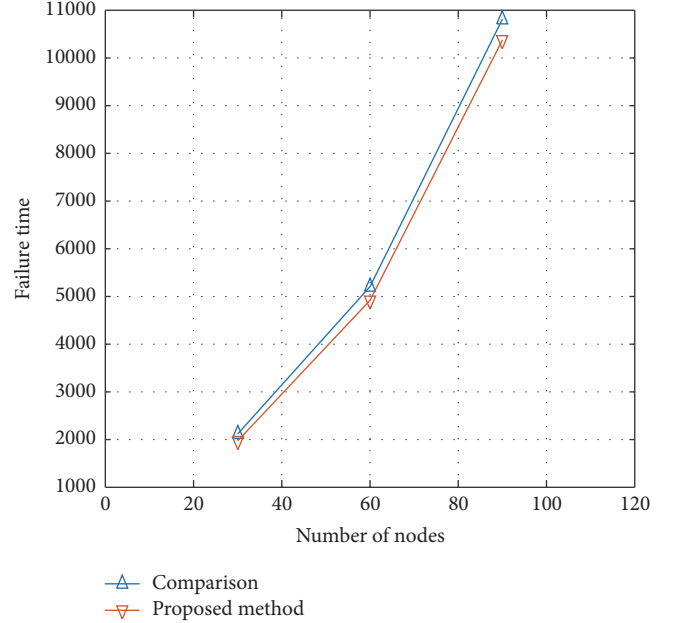


FIGURE 6: Connection lost time.

**5.3. Service Deadline Miss Ratio.** To evaluate our real-time scheduling algorithm, we set several parameters to the values shown in Table 2. After a fog server failure, the vehicles residing out of the fog server suffer from the long delay due to the multihop communication. To reflect this problem in the evaluation, we set the expected increasing hop count depending on the vehicles mobility pattern. The fog server which is responsible for recovery cannot find out the exact increased hop count, especially when the vehicles drive with random mobility. Therefore, the expected increasing hop count of vehicles which have random mobility is set higher than that of vehicles having predictable mobility. Depending on an initial value of the timer which is affected by  $\alpha_{v_i}$ , the evaluation result is differentiated. We classify two cases which have different requirements of delay by assigning the value of  $\alpha_{v_i}$ , rigid and sufficient delay cases. In the rigid delay case,  $\alpha_{v_i}$  is randomly generated from integers over the range [5...9]. Because users want to enjoy services in time, the timer is decided to the value which is close to the expected delay. This means that  $\alpha_{v_i}$  is set close to  $H_{v_i}$ . The minimum value that users require is bounded to the expected hop count where the service type is  $C$ . The upper bound is decided to the worst case where the service type is  $C$  and  $V$  with the randomly moving vehicle. On the other hand,  $\alpha_{v_i}$  is set to 10 in the sufficient delay case.

We compare our algorithm with EDF and MRF. Table 3 represents the total number of services which are provided successfully during the recovery. Since the number of services is 4 in the example, the possible scheduling cases are 24. Among these cases, there are some cases which have the maximum number of services. The optimal scheduling is one of them and their number of succeeded services is the optimal result.



The optimal result is 7 in the rigid delay environment. The proposed algorithm and the MRF show an optimal result, while EDF does not. This is because EDF only considers the deadline of the service. If many vehicles require tight delay of services, the strategy of offering many services at once is efficient like MRF.

In the sufficient delay environment, the EDF scheme outperforms the MRF, because the EDF is better than the MRF in maximizing the service throughput. Our algorithm shows the optimal result in both cases, because it considers services which have tight delay requirements while maximizing service throughput.

Based on the evaluation result, we observe two important comprehensions for rescheduling the service during the fog server recovery. First, we have to maximize service throughput while keeping delay requirements. Second, due to the services requiring low latency, it is reasonable to give a high priority to the service which will be quickly completed.

## 6. Conclusion

This paper proposed a method utilizing the network state information (i.e., the nodes mobility patterns and link quality) to overcome unstable communication in the fog computing and SDN-based connected vehicle environment. We built a campus network using the SUMO simulator and implemented the mobile SDN environment by modifying the Mininet-WiFi. Our simulation substantiated the fact that our method reduced the number of control messages by approximately 55% over the existing method. The proposed connection recovery scheme reduced failure time by about 5%. We further suggested a real-time scheduling algorithm to maximize the number of services successfully delivered to users even in a fog server failure situation.

As a future work, we will simulate a more complex situation of the fog server failure by using SUMO simulator, considering dynamic features in the connected vehicle environment such as congestion and mobility pattern in multihop communication.

## Competing Interests

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

## Acknowledgments

This work was supported by Mid-Career Researcher Program through NRF grant funded by the MEST (2016R1A2B4016588).

## References

- [1] T. H. Luan, L. Gao, Z. Li, Y. Xiang, G. We, and L. Sun, "Fog computing: focusing on mobile users at the edge," <https://arxiv.org/abs/1502.01815v3>.
- [2] S. Yi, C. Li, and Q. Li, "A survey of fog computing: concepts, applications and issues," in *Proceedings of the ACM Workshop on Mobile Big Data (Mobidata '15)*, pp. 37–42, Hangzhou, China, June 2015.
- [3] D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: a comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2015.
- [4] J. G. Andrews, S. Buzzi, W. Choi et al., "What will 5G be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, 2014.
- [5] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen, "Vehicular fog computing: a viewpoint of vehicles as the infrastructures," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3860–3873, 2016.
- [6] S. Costanzo, L. Galluccio, G. Morabito, and S. Palazzo, "Software defined wireless networks: unbridling SDNs," in *Proceedings of the 1st European Workshop on Software Defined Networks (EWSDN '12)*, pp. 1–6, October 2012.
- [7] T. Chen, M. Matinmikko, X. Chen, X. Zhou, and P. Ahokangas, "Software defined mobile networks: concept, survey, and research directions," *IEEE Communications Magazine*, vol. 53, no. 11, pp. 126–133, 2015.
- [8] I. Ku, Y. Lu, and M. Gerla, "Software-defined mobile cloud: architecture, services and use cases," in *Proceedings of the 10th International Wireless Communications and Mobile Computing Conference (IWCMC '14)*, Nicosia, Cyprus, August 2014.
- [9] N. B. Truong, G. M. Lee, and Y. Ghamri-Doudane, "Software defined networking-based vehicular Adhoc Network with Fog Computing," in *Proceedings of the 14th IFIP/IEEE International Symposium on Integrated Network Management (IM '15)*, pp. 1202–1207, Ottawa, Canada, May 2015.
- [10] OpenFlow Switch Specification, Version 1.1.0 Implemented (Wire Protocol 0x02), 2011.
- [11] K. Liu, J. K. Y. Ng, V. C. S. Lee, S. H. Son, and I. Stojmenovic, "Cooperative data scheduling in hybrid vehicular ad hoc networks: VANET as a software defined network," *IEEE/ACM Transactions on Networking*, vol. 24, no. 3, pp. 1759–1773, 2016.
- [12] B. Dezfouli, M. Radi, and O. Chipara, "Mobility-aware real-time scheduling for low-power wireless networks," in *Proceedings of the 35th Annual IEEE International Conference on Computer Communications (INFOCOM '16)*, San Francisco, Calif, USA, April 2016.
- [13] Simulation of Urban Mobility (SUMO), <http://sumo-sim.org/>.
- [14] R. R. Fontes, S. Afzal, S. H. B. Brito, M. A. S. Santos, and C. E. Rothenberg, "Mininet-WiFi: emulating software-defined wireless networks," in *Proceedings of the 11th International Conference on Network and Service Management (CNSM '15)*, pp. 384–389, Barcelona, Spain, November 2015.

## Research Article

# Distributed Association Method Assisted by Cell for Efficiency Enhancement of Wireless Networks

**Jaesung Park**

*Department of Information Security, University of Suwon, San 2-2, Wau-ri, Bongdam-eup, Hwaseong-si, Gyeonggi-do 445-743, Republic of Korea*

Correspondence should be addressed to Jaesung Park; [jaesungpark@suwon.ac.kr](mailto:jaesungpark@suwon.ac.kr)

Received 22 November 2016; Accepted 24 January 2017; Published 16 February 2017

Academic Editor: Hideyuki Takahashi

Copyright © 2017 Jaesung Park. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we propose a distributed cell association scheme called cell-guided association method (CGAM) to improve the efficiency of a wireless network. In CGAM, MSs attempt to associate with their best cells. However, unlike the conventional methods, cells do not passively accept the association requests of MSs. Instead, a cell determines whether to accept an association request or not by considering the performance of MSs already associated with it and that of the requesting MS. If a cell cannot provide a certain level of service to them, it rejects the association request and guides the requesting MS to select another cell that gives the next maximum performance metric to the MS. Since our method takes the cell resource usage into consideration, it can increase the resource efficiency of a wireless network while enhancing the overall data rate provided to MSs by balancing the number of MSs in a cell. Through performance comparisons by simulation studies, we verify that CGAM outperforms maximum SINR-based method and QoS-based method in terms of the total data rate provided by a system and outage probabilities of MSs.

## 1. Introduction

As the number of handheld devices proliferates fast, mobile data traffic has increased tremendously over a few decades. The traffic demand is expected to grow over the future because of the advent of Internet of things [1, 2]. To cope with the traffic demands, wireless networks become more and more complex and the importance of distributed resource management methods that increase both the resource efficiency of a network and quality of service (QoS) experiences by users attracts much attention. Among those, since a cell association policy that matches a cell and a mobile station (MS) affects the distribution of MSs, it influences system efficiency and QoS of users.

Traditionally, MSs attempt to associate with a cell that give them the strongest signal-to-interference and noise ratio (SINR). The rationale behind this is that data rate provided to a MS enhances as the signal quality between a MS and a cell increases [3]. However, the data rate provided to a MS is determined not only by the SINR but also by the amount of radio resources allocated to it. A MS can measure SINRs from its adjacent cells. However, a MS does not know the amount of

resources that it can obtain from a cell before it associates with the cell. Since cell resources are shared by the MSs associated with a cell, the fraction of resources allocated to each MS decreases as the number of MSs in a cell increases. Therefore, if a MS selects a cell to associate with based on the SINR, it may not receive the highest data rate. In addition, SINR between a cell and a MS is determined by the transmit power of a cell, the level of interference, and the random fading. Because of the randomness, it becomes highly probable that the number of MSs in a cell is not evenly distributed among cells even if cells are carefully deployed considering the traffic demands of areas covered by them. If the load unbalance among cells occurs, MSs in a lightly populated cell enjoy high data rate while MSs in a densely populated cell suffer from low data rate because of the resource contention among MSs.

To overcome this problem, cell load-aware or QoS-oriented cell association methods have been proposed. In [4], joint optimization of user association policies and antenna tilts settings is proposed by predicting cell load accurately. The authors in [5] propose two distributed association methods that enable MSs to make association decisions based on the information gathered by probing neighboring cells. In [6],

a user association policy is proposed to minimize the energy consumption in LTE access networks which are composed of small cells deployed densely. However, cell load-aware association methods require that cells broadcast the cell loads periodically or on-demand. In addition, MSs select a cell to associate with to maximize its benefit selfishly while cells passively accept the association request from MSs. Therefore, system resources are not used in an optimal way because of the selfishness.

To address this issue, we devise a distributed cell-guided association method (CGAM). In CGAM, cells do not passively accept the association requests from MSs. On the contrary, cells determine whether or not to accept the association request based on their resource availability. Thus, even if MSs operate selfishly to maximize its payoff, system can guide MSs to achieve its own objective such as increasing resource utilization and cell load balance. In addition, since CGAM does not require cells to broadcast cell load, conventional cell operations need not be changed.

The rest of the paper is organized as follows. In Section 2, we review related works on association management. In Section 3, we describe the CGAM in detail after introducing system model. In Section 4, we present and discuss simulation results to verify CGAM by comparing the performances of CGAM and SINR-based association method and cell load-aware association method.

## 2. Related Works

Since the cell association problem is to find a set of MSs and cells pairs that optimize a given objective function of a wireless network, optimization methods are widely used to design an association rule. Various objective functions are devised for the cell association problem. In [7], authors propose an algorithm that maximizes the system revenue while associating MSs with the minimum total transmission power. They use Benders' decomposition to solve nonconvex optimization problem optimally. Throughput maximization is also used as an objective function. In [8], sum rate is maximized and authors in [9, 10] maximize the log-utility of a network under the proportional fairness. In [11], MSs are associated with BSs in a way that global outage probability is minimized. However, it has been shown that the optimization problem is an NP-hard problem.

To overcome the NP-hardness, different kinds of optimization techniques are used to solve the cell association problem. Markov decision process is used in [12, 13]. However, it is hard to define state transition probability and the complexity increases substantially as the number of states increases. In [14], the structure of the network utility maximization problem is used to solve the problem directly. Dual decomposition is often employed to design a distributed algorithm by relaxing the optimization constraints [10, 15]. However, since the computational cost of these methods is high, it is not clear whether these methods can be applied at a small time scale when the values of input variables change very fast.

Game theory is also applied to solve the cell association problem. In [16], an evolutionary game theory is used

to devise an algorithm for cell association and antenna allocation in 5G networks with massive MIMO. Authors in [17] show that the game that users selfishly select base stations giving them the best throughput and BSs allocate the same time to their users has one Nash equilibrium point which achieves proportional fairness system-wide. However, the convergence of the algorithms based on a game theory is not guaranteed generally. Additionally, in terms of the implementation, they usually cause large overhead, which deteriorates resource utilization.

Stochastic geometry has been used to analyze the cell association problem. In [18], authors propose a distributed belief propagation algorithm to resolve user association problem and analyze the average sparsity and degree distribution using stochastic geometry. Stochastic geometry is also used in [19] to analyze the service success probability. Then, they derive the impact of cell association and user scheduling on the service success probability. Unlike the optimization methods that maximize the utility function for the current network configuration, stochastic geometry performs optimization over the average utility. Therefore, even though the complexity and overhead of a stochastic geometry approach are lower than those of repeating optimization process whenever network configuration changes, the results will be suboptimal.

Unlike the association methods based on a theoretical static interference model, a practical measurement based interference-aware association policy is proposed in [20].

There have been research works that optimize jointly cell association and other performance metrics. In [21], a tractable framework is proposed to analyze the performance of eICIC by jointly considering cell association, resource partitioning, and transmit power reduction. An energy efficient user association scheme is proposed in [22, 23]. Joint BS association and power control algorithm that updates iteratively BS association solution then the transmit power of each user is proposed in [24].

## 3. Cell-Guided Association Method

*3.1. System Model.* We consider a downlink of a wireless network that provides best effort data service to MSs. We denote by  $N_c$  the set of cells in a network and by  $|N_c|$  the cardinality of  $N_c$ . We assume that the system fully reuses frequency.

Since radio resources of a cell are allocated to MSs in the unit of radio resource block (RB) in modern wireless networks such as LTE, we assume that each cell  $x$  has  $R_x$  RBs. We consider long term SINR and throughput. In other words, we assume that the timescale measuring these values is much larger than that of fast channel variation. Thus, if we denote the transmit power of a cell  $x$  as  $P_x$ , the SINR between a MS  $y$  and a cell  $x$  is given by

$$S_{x,y} = \frac{P_x G_{x,y}}{A_N + \sum_{c \in N_c - \{x\}} P_c G_{c,y}}, \quad (1)$$

where  $A_N$  is the noise power and  $G_{x,y}$  denotes the channel gain between a cell  $x$  and a MS  $y$  which includes the system

parameter such as antenna gain and channel parameters such as path loss and shadowing.

The spectral efficiency is an increasing function of  $S_{x,y}$ . A variety of functions have been devised to reflect the effect of operational frequency and modulation and coding schemes [25–27]. However, to avoid system dependency, we use Shannon's formula to obtain the data rate for given SINR. If a cell  $x$  allocates  $R_{x,y}$  RBs to a MS  $y$ , the data rate of  $y$  is obtained as

$$D_{x,y} = R_{x,y} W_R \log_2(1 + S_{x,y}), \quad (2)$$

where  $W_R$  is the bandwidth of a RB. The number of RBs allocated to a MS is determined by the type of scheduler a cell uses. A scheduler considers the number of MSs sharing cell resources, the channel quality between a cell, and a MS when it allocates resources at each frame. In case of a round-robin scheduler, the long term RBs allocated to each MS becomes  $R_x/|M_x|$ , where  $M_x$  is the set of MSs that a cell  $x$  serves.

However, it is shown that there is a multiuser diversity gain when a cell exploits a proportional fair scheduler [28–30]. In this paper, we assume a cell uses a proportional fair scheduler. Then according to the results of [30],  $R_{x,y}$  is determined as follows.

$$R_{x,y} = \frac{R_x}{|M_x|} g(M_x), \quad (3)$$

where  $g(M_x) = \sum_{i=1}^{|M_x|} (1/i)$  is the multiuser diversity gain.

**3.2. QoS Metric.** When a MS requests for a guaranteed service such as a voice call, the number of RBs to provide the service is guaranteed by a cell. Moreover, no more RBs are allocated to the request even if a cell has surplus RBs. However, when a cell provides data service, the number of RBs allocated to each MS varies according to the number of MSs served by a cell simultaneously. For example, even when there is only one MS associated with a cell, the cell allocates all its RBs to the MS. We also note that even if a MS uses data service, users tend to give up using a network when they did not receive a minimum rate from a network. Thus, we use the outage probability [31–33] as the QoS metric for a MS. The outage probability is defined as the probability that a data rate provided to a MS is less than a given threshold  $\theta_d$ . We use the outage probability model derived in a network whose frequency reuse factor is one and where rayleigh fading channel is assumed [33]. When a MS  $y$  associates with a cell  $x$ , the outage probability of  $y$  is modeled as

$$\begin{aligned} O_{x,y} &= \text{Prob}(D_{x,y} < \theta_d) \\ &= 1 \\ &\quad - e^{-\frac{A_N B_{x,y}}{G_{x,y} P_x} \prod_{i \in N_c - \{x\}} \frac{1/(G_{i,y} P_i)}{B_{x,y}/(G_{x,y} P_x) + 1/(G_{i,y} P_i)}}, \end{aligned} \quad (4)$$

where  $B_{x,y} = 2^{\theta_d/D_{x,y}} - 1$ . Since  $O_{x,y}$  is a decreasing function of  $D_{x,y}$ ,  $O_{x,y}$  decreases if  $R_{x,y}$  or  $S_{x,y}$  increases. Therefore  $O_{x,y}$  can be regarded as a metric that represents the degree of cell load.

**3.3. CGAM Algorithm.** CGAM is composed of two algorithms, a MS algorithm and a cell algorithm, that operate independently of each other. Algorithm 1 shows each algorithm.

A MS that needs to associate with a cell searches neighboring cells by hearing signals sent by them. Then, a MS  $y$  constructs a set of adjacent cells ( $C_y$ ) that contains the cell identification numbers and SINR received from each cell in  $C_y$ . Then,  $y$  initializes a potential cell list  $C'_y = C_y$  and attempts to associate with a cell  $x$  in  $C'_y$  that gives it the maximum SINR by sending an association request message and waits for the response from  $x$ . If  $y$  receives an association response message from  $x$  saying that  $x$  does not accept the request,  $y$  removes the cell  $x$  in  $C'_y$  and attempts to associate with a cell in  $C'_y$  giving the highest SINR to it (i.e., the second best cell in  $C_y$  in terms of SINR). A MS  $y$  repeats the same process until it receives a positive association response from a cell. If  $y$  does not receive a positive response from any cell in  $C_y$ , it means that  $y$  cannot be provided with the minimum data rate from its neighboring cells. Thus, it suspend the association attempt.

Upon receiving an association request message from a MS  $y$ , a cell  $x$  calculates the outage probabilities of  $y$  and  $j$  in  $M_x$  by assuming that it accepts the request. If none of the outage probabilities are below a threshold value  $\theta_o$ ,  $x$  accepts the association request of  $y$  by sending a positive association response message to  $y$ . Otherwise,  $x$  guides  $y$  to associate with other cells that may satisfy the outage probability constraint by sending a negative association response to  $y$ .

## 4. Performance Evaluation

In this section, we evaluate the performance of the proposed cell association methods. Specifically we compare the performance of the following three methods in terms of the outage probability of a MS and data rates provided by cells.

- (i) SNM: a MS associates with a cell according to the maximum signal strength and a cell does not guide a MS to associate with better cell.
- (ii) ONM: a MS associates with a cell based on the minimum outage probability and a cell does not guide a MS to associate with better cell.
- (iii) SOM: a MS associates with a cell based on the maximum signal strength and a cell guide a MS to associate with better cell with the outage probability it can provide to an MS.

We construct urban macrocell topology following the scenario specified in 3GPP [27]. We deployed 18 base stations. Each base station has three sectors. We assume a hexagonal cell and each cell has 6 neighboring cells. The system bandwidth is set to 5 MHz, and the bandwidth of a RB is configured as 180 KHz. The intersite distance is set to 500 m. Transmit power of a base station is configured to be 46 dBm. Antenna gain of a base station is 45 dBi and that of a MS is 2 dBi. The noise power is configured as  $-111.45$  dBm and frequency reuse factor is set to 1.

```

MS Algorithm
(0) cell search
(1) construct  $C_y$ 
(2)  $C'_y \leftarrow C_y$ 
(3) While  $C'_y \neq \emptyset$ 
(4)  $x \leftarrow \operatorname{argmax} i \in C'_y (S_{i,y})$ 
(5) send association request message to  $x$ 
(6) receive association response message from  $x$ 
(7) if (response  $\neq$  OK)
(8)  $C'_y \leftarrow C'_y - \{x\}$ 
Cell Algorithm
(0) Upon receiving association request from a MS  $y$ 
(1)  $M'_x \leftarrow M_x \cup \{y\}$ 
(2) flag  $\leftarrow 0$ 
(3) while  $M'_x \neq \emptyset$ 
(4) calculate  $O_{x,j}$ , ( $j \in M'_x$ )
(5) if ( $O_{x,j} > \theta_o$ )
(6) send association response with NOK
(7) flag  $\leftarrow 1$ 
(8) break
(9) else
(10)  $M'_x \leftarrow M_x - \{i\}$ 
(11) if (flag == 0)
(12) send association response with OK

```

ALGORITHM 1: Cell-guided association method.

The path loss model of  $128.1 + 37.6 \log_{10}(\max(d, 0.035))$  is applied, where  $d$  is the distance in km between a sender and a receiver. We assume log-normal shadowing channel that has zero mean and standard deviation of  $\sigma_{\text{dB}} = 8$  dB.

We locate a MS in a network randomly following the uniform distribution and apply each method for a MS to determine a cell to associate with. All the statistics are gathered when the number of MSs reach 5,000.

**4.1. MS Performance.** To quantify the performance of MS, we use two metrics. One metric is the data rate received by a MS and the other is the outage probability of a MS. According to the locations of MSs, we further classify MSs into edge MSs which are defined as the MSs whose Euclidean distance from its serving base station is larger than intersite distance over three.

Figure 1 shows the cumulative distribution of data rates received by MSs and Figure 2 shows the cumulative distribution of outage probabilities experienced by MSs. Comparing SNM with ONM, performances of MSs increase by changing cell selection metric from SINR to outage probability. For example, 70% MSs receive less than 575.8 Kbps when SINR is used as a cell selection metric. On the contrary 70% MSs receive 607.9 Kbps when MSs use outage probability to select a cell. In terms of the outage probability, the proportion of MSs whose outage probabilities are less than 0.2 is 0.794 when SINR is used. On the contrary, the proportion increases to 0.817 when MS selects a cell with an outage probability.

The performances of MSs increase dramatically when a cell guides MSs when they select a cell to associate with. When

TABLE 1: Cell statistics in terms of the number of MSs in a cell.

	SNM	ONM	SOM
FI	0.88	0.96	0.92
Avg.	215	208	137
Std.	80	41	43

the association method is changed from SNM to SOM, the data rate received by 70% MSs increases from 575.8 Kbps to 904.2 Kbps. In addition, the proportion of MSs whose outage probabilities are less than 0.2 increases 1.06 times from 0.794 to 0.842.

The performance gain obtained by guiding MSs with outage probability is more significant for the edge MSs. Ninety percentage of edge MSs receive less than 793.3 Kbps when SNM is used while it increases to 1301.9 Kbps when SOM is used. The outage probabilities of 90% edge MSs are less than 0.5 when SNM is applied while the number decreases to 0.37 when MSs use SOM to associate with a cell.

**4.2. System Performance.** Figure 3 shows the total data rate provided by each cell called cell rate. We can observe that the cell rate increase dramatically if there is a guide by a cell. This is attributed to the fact that if MSs choose cells selfishly to maximize their own performance metric, some cells are crowded with MSs while the others are loosely populated. To examine the distribution of MSs in a cell, in Table 1, we show Jain's fairness index (FI) in terms of the number of MSs in a cell. In the table, we also show the average and the standard

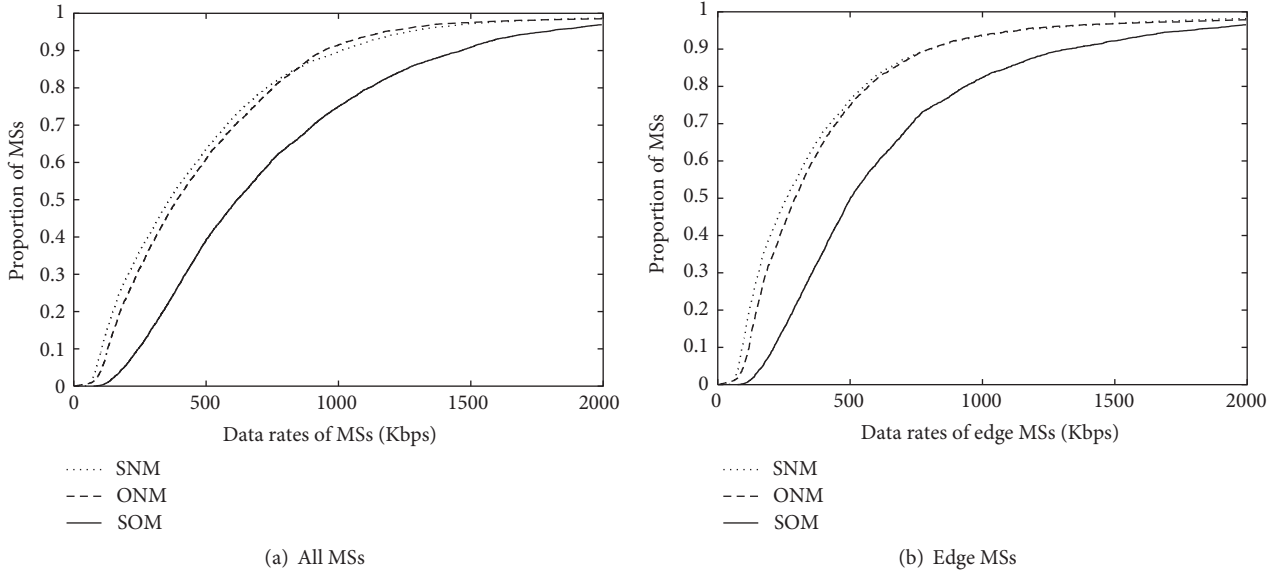


FIGURE 1: Cumulative distribution of data rates received by MSs.

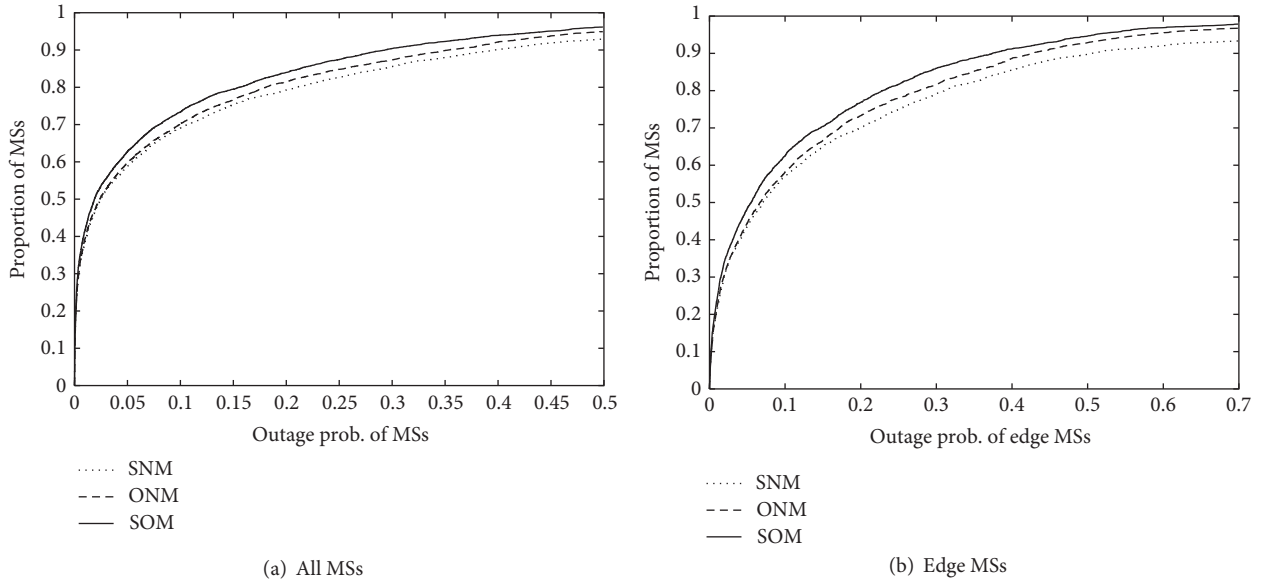


FIGURE 2: Cumulative distribution of MS outage probability.

deviation of the number of MSs in a cell. We can find that FI is 0.88 and the standard deviation is 80 when SNM is used while FI and the standard deviation of SOM are 0.92 and 43, respectively, which means that MSs are more evenly populated when SOM is used than when SNM is used. ONM also distribute MSs evenly. However, the average number of MSs in a cell of ONM is higher than those of SOM. Since cell resources are shared among MSs, the number of RBs that is allocated to each MS in a cell becomes smaller as the number of MSs in a cell increases. In addition, since SINR is inversely proportional to the distance from a base station and a MS and

we deployed MSs according to the uniform distribution, the average data rate per RB decreases as the number of MSs in a cell increases. Therefore, even though cell rate of ONM is higher than that of SNM, it is smaller than the cell rate of SOM.

To further investigate the benefit of cell guidance, we show the minimum data rate provided to a MS by each cell in Figure 4 and the maximum outage probability among the MSs in each cell experience in Figure 5. We can observe that the minimum data rate provided to MSs in a cell increases and the maximum outage probability in each cell decreases.

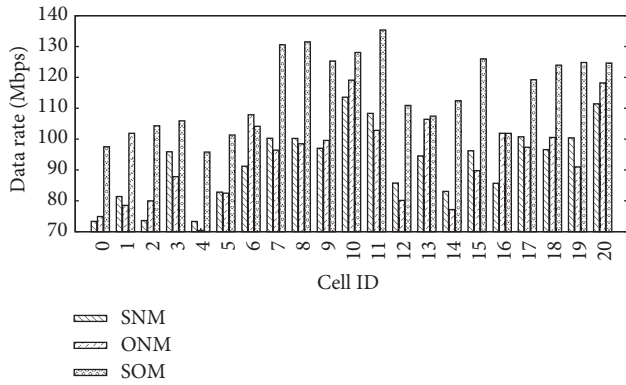


FIGURE 3: Total data rate provided by a cell.

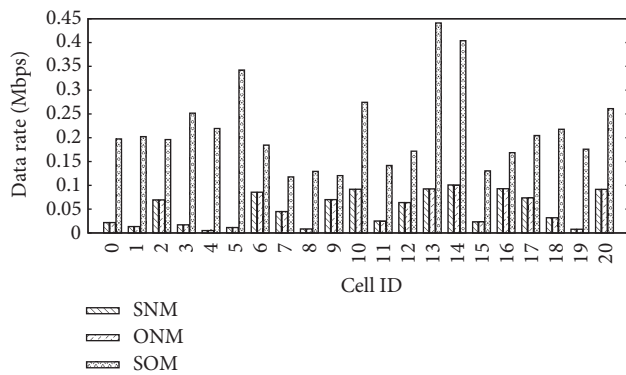


FIGURE 4: Minimum data rate provided to a MS associated with a cell.

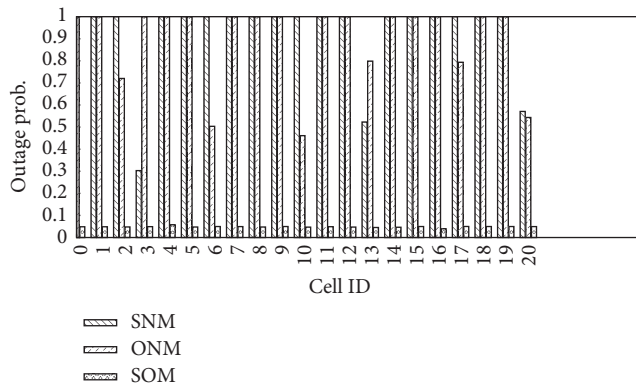


FIGURE 5: Maximum outage probability provided to a MS associated with a cell.

## 5. Conclusions

In this paper, we propose a cell-guided association method. Unlike other cell load aware methods where MSs estimate the load of neighboring cells before attempting to make associations with a cell, in CGAM, a cell decides whether or not to accept an association request from a MS. Considering the amount of available resources, a cell rejects the association request if it cannot provide the minimum data rate to a MS. The rejection guides a MS to try to associate with the second

best cell. Thus, even if a MS selfishly selects a cell to maximize its benefit, they are not densely populated in some cells while the other cells are sparsely populated. Therefore, CGAM increases not only the resource utilization of a system but also the quality of service perceived by MSs. In addition, CGAM does not require additional information in a cell broadcast message and conventional cell selection metric of a MS.

## Competing Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work was partly supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2015RID1A1A01060117) and by the GRRRC program of Gyeonggi Province [(GRRRC SUWON2016-B2), Study on Object Recognition System for Industrial Security and Intelligent Control System for Establishing Social Safety Net Based on Advanced Neuro-Fuzzy Technologies].

## References

- [1] D. Evans, "The Internet of Things: How the Next Evolution of the Internet is Changing Everything," white paper, April 2011.
- [2] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2014–2019," White Paper, February 2015.
- [3] J. G. Andrews, S. Singh, Q. Ye, X. Lin, and H. S. Dhillon, "An overview of load balancing in HetNets: old myths and open problems," *IEEE Wireless Communications*, vol. 21, no. 2, pp. 18–25, 2014.
- [4] A. J. Fehske, H. Klessig, J. Voigt, and G. P. Fettweis, "Concurrent load-aware adjustment of user association and antenna tilts in self-organizing radio networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 5, pp. 1974–1988, 2013.
- [5] K. C. Garikipati and K. G. Shin, "Distributed association control in shared wireless networks," in *Proceedings of the 10th Annual IEEE Communications Society Conference on Sensing and Communication in Wireless Networks (SECON '13)*, pp. 362–370, New Orleans, La, USA, June 2013.
- [6] C. Bottai, C. Ciconetti, A. Morelli, M. Rosellini, and C. Vitale, "Energy-efficient user association in extremely dense small cell networks," in *Proceedings of the European Conference on Networks and Communications (EuCNC '14)*, pp. 1–5, Bologna, Italy, June 2014.
- [7] L. P. Qian, Y. J. A. Zhang, Y. Wu, and J. Chen, "Joint base station association and power control via Benders' decomposition," *IEEE Transactions on Wireless Communications*, vol. 12, no. 4, pp. 1651–1665, 2013.
- [8] S. Corroy, L. Falconetti, and R. Mathar, "Cell association in small heterogeneous networks: downlink sum rate and min rate maximization," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '12)*, pp. 888–892, Paris, France, April 2012.
- [9] K. Shen and W. Yu, "Distributed pricing-based user association for downlink heterogeneous cellular networks," *IEEE Journal on*

- Selected Areas in Communications*, vol. 32, no. 6, pp. 1100–1113, 2014.
- [10] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis, and J. G. Andrews, “User association for load balancing in heterogeneous cellular networks,” *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2706–2716, 2013.
- [11] H. Boostanimehr and V. K. Bhargava, “Unified and distributed QoS-driven cell association algorithms in heterogeneous networks,” *IEEE Transactions on Wireless Communications*, vol. 14, no. 3, pp. 1650–1662, 2015.
- [12] D. Kumar, E. Altman, and J.-M. Keli, “Globally optimal user-Network Association in an 802.11 WLAN & 3G UMTS hybrid cell,” in *Proceedings of the 20th International Teletraffic Conference on Managing Traffic Performance in Converged Networks*, pp. 1173–1187, Ottawa, Canada, June 2007.
- [13] C. Sun, E. Stevens-Navarro, and V. W. S. Wong, “A constrained MDP-based vertical handoff decision algorithm for 4G wireless networks,” in *Proceedings of the IEEE International Conference on Communications (ICC '08)*, pp. 2169–2174, IEEE, Beijing, China, May 2008.
- [14] K. Son, S. Chong, and G. De Veciana, “Dynamic association for load balancing and interference avoidance in multi-cell networks,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 7, pp. 3566–3576, 2009.
- [15] K. Chitti, Q. Kuang, and J. Speidel, “Joint base station association and power allocation for uplink sum-rate maximization,” in *Proceedings of the IEEE 14th Workshop on Signal Processing Advances in Wireless Communications (SPAWC '13)*, pp. 6–10, Darmstadt, Germany, June 2013.
- [16] P. Wang, W. Song, D. Niyato, and Y. Xiao, “QoS-aware cell association in 5G heterogeneous networks with massive MIMO,” *IEEE Network*, vol. 29, no. 6, pp. 76–82, 2015.
- [17] L. Jiang, S. Parekh, and J. Walrand, “Base station association game in multi-cell wireless networks,” in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '08)*, pp. 1616–1621, Las Vegas, Nev, USA, March 2008.
- [18] Y. Chen, J. Li, Z. Lin, G. Mao, and B. Vucetic, “User association with unequal user priorities in heterogeneous cellular networks,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7374–7388, 2016.
- [19] S. Chen, M. Peng, H. Zhang, and C. Wang, “Investigation of service success probability for downlink heterogeneous cellular networks with cell association and user scheduling,” in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '15)*, pp. 1434–1439, IEEE, New Orleans, La, USA, March 2015.
- [20] B. Rengarajan and G. De Veciana, “Practical adaptive user association policies for wireless systems with dynamic interference,” *IEEE/ACM Transactions on Networking*, vol. 19, no. 6, pp. 1690–1703, 2011.
- [21] W. Tang, S. Feng, Y. Liu, and M. C. Reed, “Joint low-power transmit and cell association in heterogeneous networks,” in *Proceedings of the 58th IEEE Global Communications Conference (GLOBECOM '15)*, pp. 1–6, IEEE, San Diego, Calif, USA, December 2015.
- [22] Y. Zhu, Z. Zeng, T. Zhang, L. An, and L. Xiao, “An energy efficient user association scheme based on cell sleeping in LTE heterogeneous networks,” in *Proceedings of the IEEE International Symposium on Wireless Personal Multimedia Communications (WPMC '14)*, pp. 75–79, September 2014.
- [23] H. Pervaiz, L. Musavian, and Q. Ni, “Joint user association and energy-efficient resource allocation with minimum-rate constraints in two-tier HetNets,” in *Proceedings of the IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC '13)*, pp. 1634–1639, London, UK, September 2013.
- [24] V. N. Ha and L. B. Le, “Distributed base station association and power control for heterogeneous cellular networks,” *IEEE Transactions on Vehicular Technology*, vol. 63, no. 1, pp. 282–296, 2014.
- [25] A. M. Rao, A. Weber, S. Gollamudi, and R. Soni, “LTE and HSPA+: revolutionary and evolutionary solutions for global mobile broadband,” *Bell Labs Technical Journal*, vol. 13, no. 4, pp. 7–34, 2009.
- [26] S. S. Mwanje and A. Mitschele-Thiel, “Minimizing handover performance degradation due to LTE self organized mobility load balancing,” in *Proceedings of the IEEE 77th Vehicular Technology Conference (VTC Spring '13)*, June 2013.
- [27] 3GPP TR 36.942 Ver. 11.0.0 Rel. 11, “Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (EUTRA); Radio Frequency (RF) system scenarios,” Tech. Rep., 2012.
- [28] R. Combes, Z. Altman, and E. Altman, “Scheduling gain for frequency-selective Rayleigh-fading channels with application to self-organizing packet scheduling,” *Performance Evaluation*, vol. 68, no. 8, pp. 690–709, 2011.
- [29] R. Combes, S. E. Elayoubi, and Z. Altman, “Cross-layer analysis of scheduling gains: application to LMMSE receivers in frequency-selective Rayleigh-fading channels,” in *Proceedings of the IEEE International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt '11)*, pp. 133–139, Princeton, NJ, USA, May 2011.
- [30] H. J. Kushner and P. A. Whiting, “Convergence of proportional-fair sharing algorithms under general conditions,” *IEEE Transactions on Wireless Communications*, vol. 3, no. 4, pp. 1250–1259, 2004.
- [31] C. Tellambura, “Computing the outage probability in mobile radio networks using the sampling theorem,” *IEEE Transactions on Communications*, vol. 47, no. 8, pp. 1125–1128, 1999.
- [32] S. Tang, B. L. Mark, and A. E. Leu, “An exact solution for outage probability in cellular networks,” in *Proceedings of the Military Communications Conference (MILCOM '07)*, October 2007.
- [33] H. Boostanimehr and V. K. Bhargava, “Distributed and QoS-driven cell association in HetNets to minimize global outage probability,” in *Proceedings of the IEEE Global Communications Conference (GLOBECOM '14)*, pp. 3665–3671, IEEE, Austin, Tex, USA, December 2014.



## Review Article

# Principles, Applications, and Challenges of Synchronization in Nature for Future Mobile Communication Systems

Hyun-Ho Choi<sup>1</sup> and Jung-Ryun Lee<sup>2</sup>

<sup>1</sup>*Department of Electrical, Electronic and Control Engineering and the Institute for Information Technology Convergence, Hankyong National University, Anseong 17579, Republic of Korea*

<sup>2</sup>*School of the Electrical Engineering, Chung-Ang University, Seoul 06974, Republic of Korea*

Correspondence should be addressed to Jung-Ryun Lee; jrlee@cau.ac.kr

Received 18 November 2016; Revised 13 December 2016; Accepted 22 December 2016; Published 22 January 2017

Academic Editor: Yujin Lim

Copyright © 2017 Hyun-Ho Choi and Jung-Ryun Lee. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In dynamic and complex natural environments, a number of individuals such as fireflies, flocks of birds, schools of fish, and pacemaker cells show various synchronization phenomena, with the purpose of achieving their certain goals in an efficient and distributed manner. So far, there has been considerable research attention on the working principles behind synchronization phenomena in nature and, as a result, various models and theoretical investigation have been developed to apply synchronization principles to various mobile communication systems. In this article, we present an exploration of synchronization phenomena in nature. Some representative models on synchronization are investigated and its working principles are analyzed. In addition, we survey some key applications inspired by synchronization principles for the future mobile communication systems. The characteristics and limitations of the applications inspired by synchronization in nature are evaluated in the context of the use of nature-inspired technologies. Finally, we provide the discussion of further research challenges for developing the advanced application of natural synchronization phenomena in the future mobile communication systems.

## 1. Introduction

The inclination of living entities, ranging from animals to humans, to synchronize with each other is the most common tendency in the universe [1]. Thousands of fireflies synchronously illuminate, while geese fly at the same speed in formation. Applause at concert halls merges to produce a harmonized sound in time, and the menstrual periods of women who closely interact for a long time also synchronize. Thousands of cardiac pacemaker cells in the heart fire in synchronization to sustain life. Inanimate objects, such as particles and planets, synchronize as well. Laser beams are created when trillions of atoms oscillating in sync emit photons of the same phase and frequency. Moreover, only one side of the moon can be viewed because the orbital and rotational periods of the moon are synchronized by the gravitational pull between the earth and moon.

As shown in the above examples, synchronization occurs with both conscious living organisms and inanimate objects

without consciousness. A key aspect of this phenomenon is that it does not involve a leader who directs the behavior, nor does it require the obtaining of clues from the surrounding environment. Rather, the entities synchronize in a certain rhythm. In other words, this order of synchronization emerges—it naturally occurs out of “nothing”—in all cases [2].

Fireflies, flocks of birds, schools of fish, and pacemaker cells are all modeled as populations of oscillators [3]. An oscillator is an entity that continues to repeat itself in a regular time interval. If more than one oscillator influences another oscillator through a physical or chemical process, they are considered to be connected. Fireflies communicate through light, birds in a flock identify their respective position using their sight, and pacemaker cells transmit electric currents to each other. Nature utilizes all available communication channels for the connecting of oscillators. This kind of communication often results in synchronization.

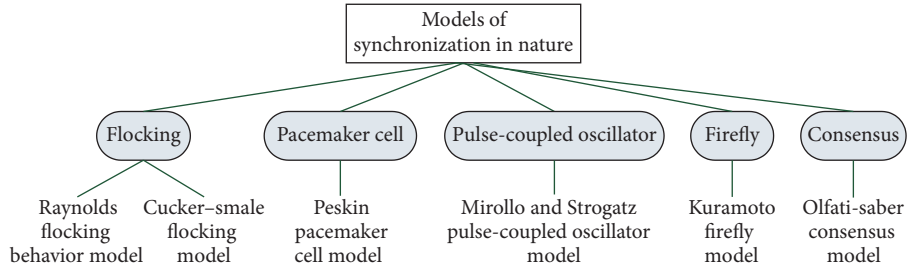


FIGURE 1: A taxonomy of models of synchronization in nature.

Synchronization in nature has been attracting considerable research attention and has been applied to theoretical investigations and a variety of mobile communication and networking systems [4–6]. Some examples of its applications include time synchronization and scheduling in large-scale distributed networks; distributed fusion in sensor networks; rapid consensus in various network topologies; distributed formation control of multiple vehicles; and solutions of nonlinear optimization problems. Due to the advancement of communication technologies in the last few years, the number of nodes available for networking is dramatically increasing. Consequently, applications that are intended to achieve collective goals through cooperation among nodes are increasing. However, the conventional approach of controlling numerous nodes in a centralized system has limitations in terms of cost, complexity, resource availability, and performance. Therefore, approaches inspired by the phenomena of natural orders, such as synchronization—which efficiently achieve certain goals in complex environments with various constraints—are being investigated to address complex engineering problems.

In this article, we examine synchronization phenomena in nature, elucidate representative mathematical models concerning synchronization, and investigate the principles of synchronization. We then introduce some applications that apply synchronization principles to communication network systems. We thereby analyze the attributes and limitations of nature-inspired synchronization methods. Finally, we discuss research challenges for developing the advanced applications based on synchronization phenomena in the future mobile communication systems.

The remainder of this article is organized as follows. Section 2 overviews representative theoretical studies on synchronization, and the principles of synchronization are investigated. Section 3 examines major applications based on these principles. Section 4 describes challenges involved in applying synchronization phenomena. Section 5 discusses the similarities between synchronization in nature and certain engineering problems, while highlighting the use of nature-inspired technologies. Section 6 presents our conclusions.

## 2. Principles of Synchronization

In this section, representative theoretical synchronization models are outlined to elucidate synchronization principles and related conditions. The taxonomy of synchronization

model is provided in Figure 1, which is classified by the objects in nature that inspire to make each synchronization model.

**2.1. Reynolds Flocking Behavior Model.** Reynolds created the first behavioral model of animal groups, such as those of birds and fish, and he demonstrated their behaviors through computer simulation [7]. In this “flocking” model, animal group behavior aligns with three rules—separation, alignment, and cohesion—as shown in Figure 2. Each animal in a group independently controls its own position and speed based on these rules. According to the separation rule, each animal tends to maintain a certain distance from its neighboring group members. In the alignment rule, each animal tends to determine its heading direction in accordance with the average direction of its neighboring group members. According to the cohesion rule, each animal tends to steer itself toward the average position of its neighboring group members to avoid being separated from them. An animal’s neighbors in this context are determined according to the physical distance between the respective animal and the surrounding animals, as well as their respective visual fields. The other group members that are not recognized in this way are excluded from the calculation. Accordingly, each animal in the group adjusts its position and speed based on the average position and speed of its neighbors in accordance with the three rules. After a period of time, the animals move at the same speed in the same distance. That is, synchronization occurs in terms of distance and speed.

**2.2. Peskin Pacemaker Cell Model.** Peskin modeled cardiac pacemaker cells as a parallel circuit with capacitance and resistance [8]. This electrical circuit is charged along a gradually increasing curve and fires when the voltage reaches a certain threshold, as shown in Figure 3. Thereafter, the voltage resets and a new cycle begins. That is, this behaves like an oscillator. The pacemaker cell model mimics the periodic firing of cardiac cells and decreasing of voltage to the bottom of a curve.

Then, Peskin modeled the cardiac cellular phenomenon as a network of electrical oscillators. Among  $N$  oscillators, the voltage of the  $i$ th oscillator,  $x_i$ , is modeled as

$$\frac{dx_i}{dt} = S_0 - \gamma x_i, \quad 0 \leq x_i \leq 1, \quad i = 1, 2, \dots, N, \quad (1)$$

where  $S_0$  is the initial value and  $\gamma$  indicates the rate of dissipation. The  $i$ th oscillator fires when  $x_i = 1$  and it regresses

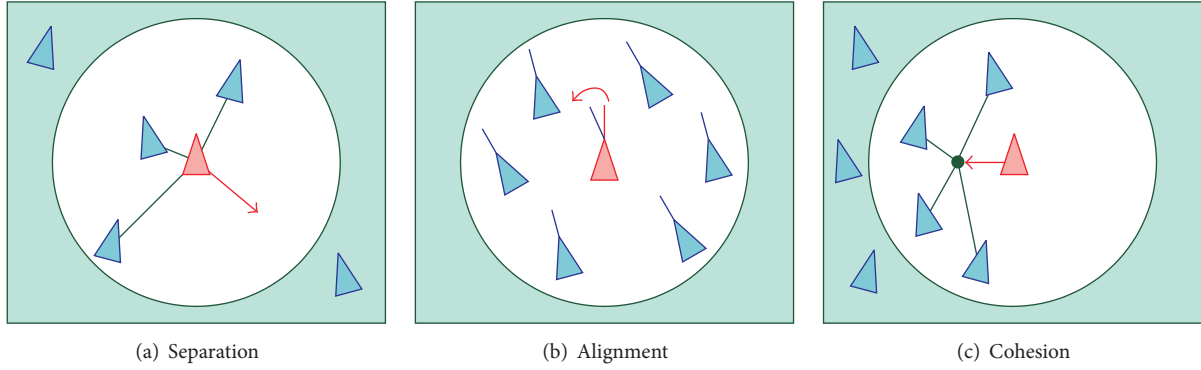


FIGURE 2: Three rules of the flocking behavior model.

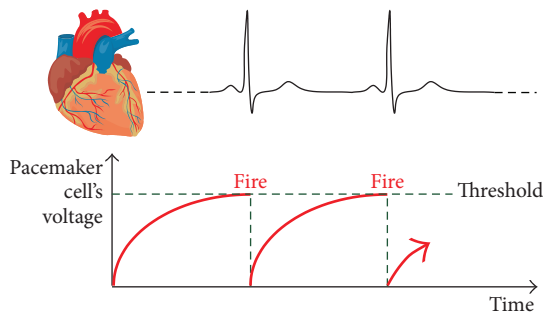


FIGURE 3: Cardiac pacemaker cell model.

to  $x_i = 0$ . Oscillators influence each other through firing. If an oscillator fires, either the voltages of the other oscillators that detect it increase by coupling strength  $\epsilon$  or they fire when their voltages increase beyond the given threshold. This rule is expressed as

$$x_i(t) = 1 \rightarrow x_j(t^+) = \min(1, x_j(t) + \epsilon), \quad \forall j \neq i. \quad (2)$$

Assuming that all oscillators are identical (i.e., they have the same charge curve) and are coupled by the same strength, Peskin proved that two oscillators always simultaneously converge and fire, even when starting activity in random initial conditions.

**2.3. Strogatz Pulse-Coupled Oscillator Model.** Mirollo and Strogatz modeled the synchronization phenomenon of entities into a pulse-coupled oscillator (PCO) [9]. PCO has an integrated clock. It emits pulses along a certain defined cycle and receives the pulses of other oscillators through the media existing between them. When receiving these pulses, each oscillator adjusts its internal clock according to appropriate clock adjustment rules. The synchronization then occurs after a period of time.

Figure 4 illustrates the PCO phase synchronization process. In the PCO model, each node acts as an oscillator with a fixed time cycle,  $T$ . The oscillator has an internal time phase,  $t$ , and increases at a certain rate from  $t = 0$  to  $t = T$ . When  $t = T$ , the nodes fire and the phase is reset to  $t = 0$ . At that point, the surrounding nodes that detect the firing signal

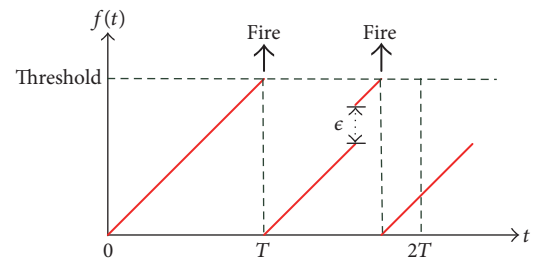


FIGURE 4: PCO phase synchronization process.

adjust their phases of  $t$  to the upper point to shorten the time until firing. The new phase value,  $t_{\text{update}}$ , is determined as

$$t_{\text{update}} = f^{-1}(f(t) + \epsilon), \quad (3)$$

where firing function  $f$  should be an asymptotically increasing concave function, and the increased amount  $\epsilon$  is a real number smaller than 1. When the value of  $t_{\text{update}}$  is greater than  $T$ , the node immediately fires and  $t_{\text{update}}$  is reset to 0. The nodes initiate activity with different  $t$  values; nonetheless, in accordance with the same rules, their phases of  $t$ , respectively, converge to the same value in the course of time.

Strogatz generalized the coupled oscillator model of Peskin into  $N$  oscillators. According to Strogatz model, under the assumption that all oscillators are identical and connected, it is proved that  $N$  oscillators with random initial values always synchronize. However, it was found that PCO model synchronization is not realized in actuality when noise or a pulse transmission delay exists [10].

**2.4. Kuramoto Firefly Model.** Assuming that identical oscillators are weakly coupled and the interactions between the oscillators are dependent on the sine function of the phase difference, Kuramoto modeled the synchronization of fireflies as

$$\frac{d\theta_i}{dt} = \omega_i - \frac{K}{N} \sum_{j=1}^N \sin(\theta_i - \theta_j), \quad i = 1, 2, \dots, N, \quad (4)$$

where  $\theta_i$  is the phase of each individual,  $\omega_i$  is the frequency of each individual, and  $K$  is the coupling strength [11–13].

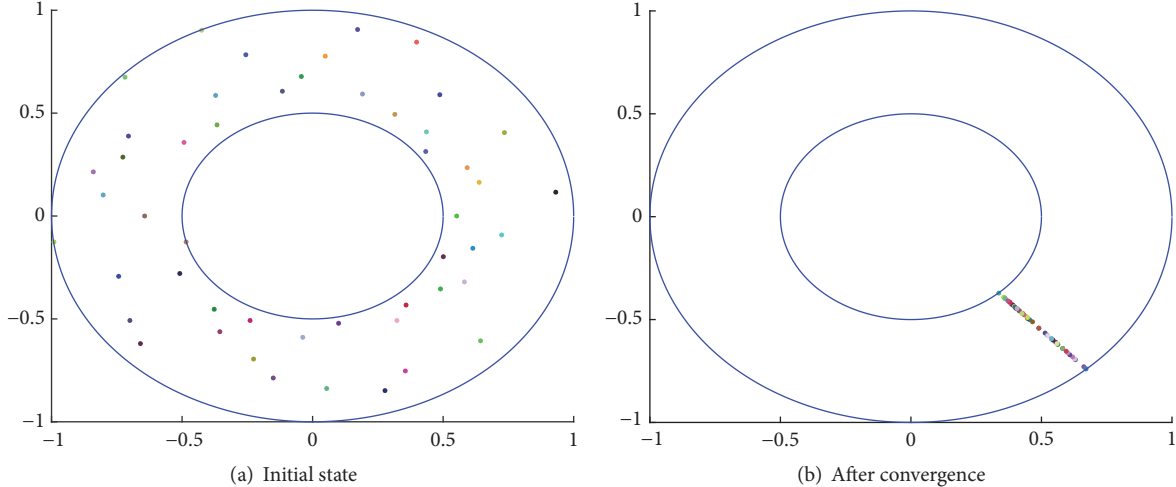


FIGURE 5: Phase synchronization phenomenon according to the Kuramoto model.

In the initial state before synchronization, each individual oscillates at its own frequency, and its phase differs from those of the others. However, the phase of each individual changes in time through interactions by  $\sin(\theta_i - \theta_j)$ . Ultimately, synchronization occurs when the phases of all the individuals converge, as shown in Figure 5.

Unlike the Strogatz model, the Kuramoto model considers oscillators that have different natural frequencies and assumes that the interactions of the oscillators are limited to the sine function. Therefore, the Kuramoto model can be exactly solved in mathematical terms, even though it is a nonlinear model. This model achieves synchronization with a random initial value on the condition that  $K$  value is sufficiently large in a fully connected network [14]. However, synchronization is not always guaranteed when the differences of the natural frequencies of the oscillators transcend a specific range or when a transmission delay exists [15].

**2.5. Cucker–Smale Flocking Model.** Cucker and Smale mathematically modeled the flocking behavior of birds in flight as

$$\begin{aligned} v_i(t+1) - v_i(t) &= \frac{\lambda}{N} \sum_{j=1}^N \Psi(|x_j(t) - x_i(t)|) (v_j(t) - v_i(t)), \end{aligned} \quad (5)$$

where  $x_i(t)$  and  $v_i(t)$  are the position and velocity of the  $i$ th bird at time  $t$ , respectively [16].  $N$  is the total number of birds,  $\lambda$  is the coupling strength among the birds, and  $\Psi$  is the communication range function with input of the distance between two birds. For example, if each bird in a flock has only information of the adjacent birds within a certain distance,  $r$ , then  $\Psi = 1$  when  $|x_j - x_i| \leq r$  and  $\Psi = 0$  otherwise. As expressed in (5), the Cucker–Smale flocking model shows that each bird adjusts its velocity by averaging the velocity of its perceived neighbors. Therefore, this model can be a mathematical representation of the Reynolds flocking behavior model.

Cucker and Smale considered various  $\Psi$  functions and proved the synchronization conditions for the general  $\Psi$  functions in which path loss is proportional to distance [16, 17]. They proved that the convergence value by (5) becomes the average of the initial velocities of the birds. Additionally, they showed that the position of each bird does not diverge; rather, it remains within a certain range of space when converging. The convergence characteristics of the Cucker–Smale model are expressed as

$$v_c := \lim_{t \rightarrow \infty} v_i(t) = \frac{1}{N} \sum_{i=1}^N v_i(0), \quad \text{for } \forall i, \quad (6)$$

$$\sup_{0 \leq t < \infty} |x_i(t) - x_j(t)| < \infty, \quad \text{for } \forall i \neq j.$$

Figure 6 shows the simulated synchronization phenomenon of bird flocking according to this model.

**2.6. Olfati-Saber Consensus Model.** In dynamic systems, such as multiagent networks, the term “consensus” is used to mean “synchronization” [18]. Consensus connotes the reaching of an agreement regarding a certain quantity of interest that depends on the state of all agents. The consensus algorithm refers to an interaction rule that specifies the information exchange between an agent and all of its neighbors in the network in order to reach a consensus.

In accordance with continuous time and discrete time intervals in a linear system, a generalized consensus algorithm is, respectively, expressed as

$$\begin{aligned} x'_i(t) &= \sum_{j \in N_i} a_{ij} (x_j(t) - x_i(t)), \\ x_i(k+1) &= x_i(k) + \epsilon \sum_{j \in N_i} a_{ij} (x_j(k) - x_i(k)), \end{aligned} \quad (7)$$

where  $x_i(t)$  is the status value of node  $i$  at time  $t$ ,  $N_i$  is the set of neighboring nodes of node  $i$ ,  $a_{ij}$  is the coupling strength

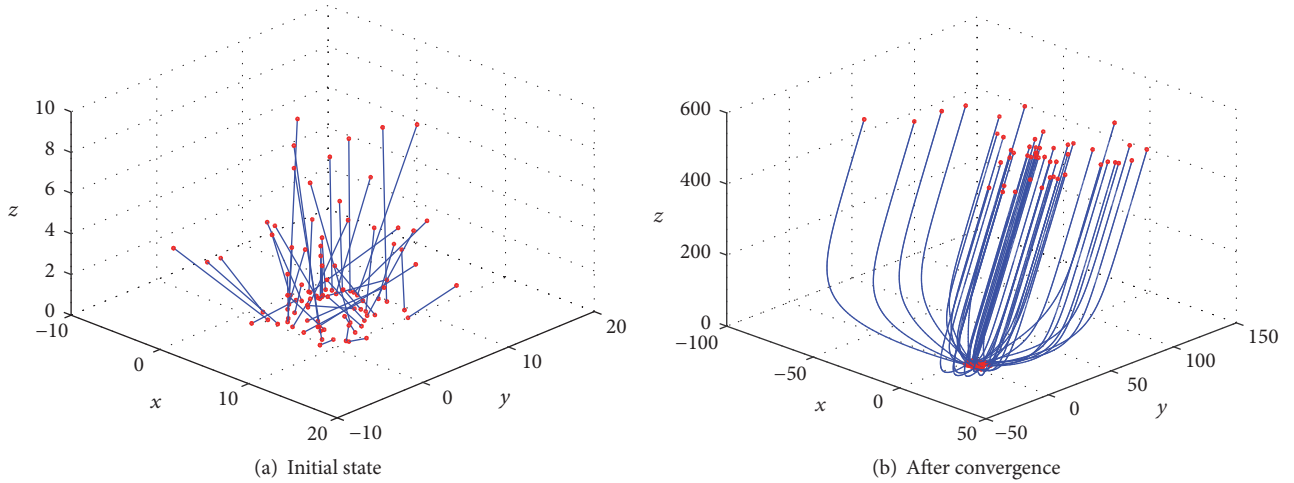


FIGURE 6: Synchronization phenomenon of bird flocking in three-dimensional space according to the Cucker–Smale flocking model.

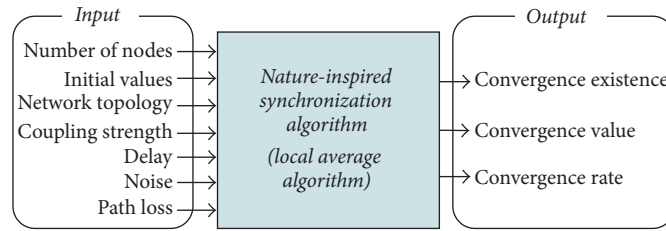


FIGURE 7: System model of synchronization.

between the two nodes,  $i$  and  $j$ , and  $\epsilon > 0$  is the step size. According to (7), the change of the status value of each node is determined by the sum of the differences of the neighboring nodes' state values. Based on these algorithms, the status value of each node converges in time to the average of the initial values of all the nodes. It is given as

$$\alpha = \frac{1}{N} \sum_{i=1}^N x_i(0), \quad (8)$$

where  $N$  is the total number of nodes. The expressions and convergence result of this kind of consensus algorithm correspond to those of the Cucker–Smale model.

Various theoretical studies have been conducted on the consensus model and their convergence rate has been analyzed. Equations (7) can be simply converted into the matrix forms as follows:

$$\begin{aligned} \mathbf{x}' &= -\mathbf{L}\mathbf{x}, \\ \mathbf{x}(k+1) &= \mathbf{P}\mathbf{x}(k), \end{aligned} \quad (9)$$

where  $\mathbf{L}$  is a Laplacian matrix and  $\mathbf{P}$  is a Perron matrix. These matrices are determined according to the given network topology. The convergence rate of each consensus algorithm is determined depending on the second smallest eigenvalue of Laplacian  $\mathbf{L}$  and the second largest eigenvalue of Perron  $\mathbf{P}$ , respectively [18].

**2.7. Synchronization System Model.** In synchronization studies to date, the action of each individual is mathematically modeled. Then, the existence of convergence, the convergence value, and the convergence rate are theoretically analyzed based on the various conditions, such as the number of participating nodes, initial values, network topology, coupling strength between nodes, delay, noise, and path loss. The principle of synchronization algorithms is that each node repeats the process of updating its own information using only the information of the neighboring nodes. This update approach can thus be represented as the *local average* algorithm by which the convergence value is decided solely based on the average of the initial values. This synchronization algorithm can be modeled into a generalized system with inputs and outputs, as depicted in Figure 7.

### 3. Applications of Synchronization

In this section, we introduce some major applications based on synchronization principles, focusing on communication and networking systems. The taxonomy of synchronization applications is provided in Figure 8, which is classified by the major applications used in communication and networking systems.

**3.1. Distributed Time Synchronization.** In an environment in which many nodes exist and the network dynamically changes according to the movements, entering, and exiting

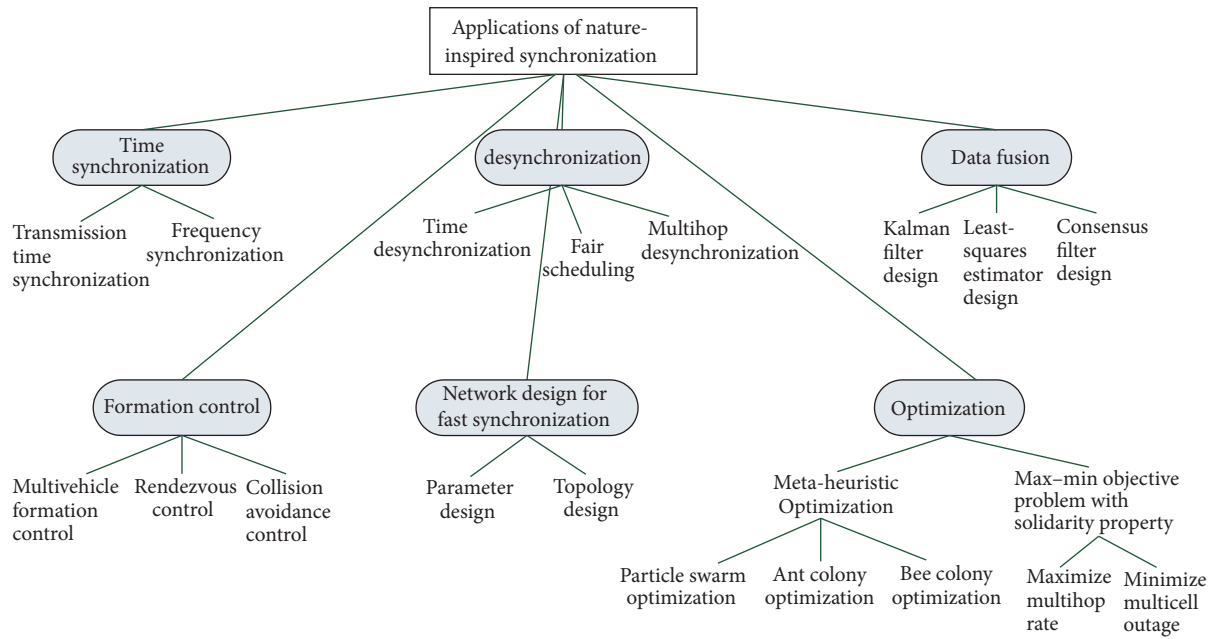


FIGURE 8: A taxonomy of applications of nature-inspired synchronization.

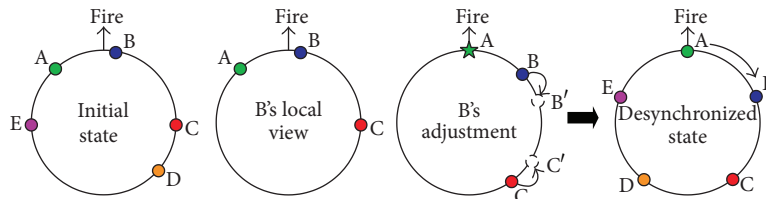


FIGURE 9: DESYNC algorithm.

of nodes—such as in mobile ad hoc, wireless mesh, and sensor networks—a nature-inspired distributed synchronization method is more suitable than the conventional centralized synchronization methods. Various studies on this premise have been conducted. A synchronization method for node transmission time was suggested by applying the synchronization principle of fireflies to a large-scale network [19, 20]. Therein, the Strogatz PCO model was applied to a large-scale sensor network. Synchronization performance was then analyzed according to the number of nodes and optimized according to the tradeoff of the energy efficiency and convergence rate [21]. A technique for synchronizing the timing of separate networks was suggested by using the positions and mobility of the nodes in mobile ad hoc networks [22]. A distributed frequency synchronization technique using a bioinspired algorithm was suggested to solve the problem of multiple frequency offsets in mesh networks wherein multiple nodes have different carrier frequencies [23]. Lastly, a synchronization method was suggested for realistic situations in which a time delay occurs across nodes and not all of them are connected [15, 24].

**3.2. Desynchronization.** Unlike synchronization, by which oscillators converge to the same time phase, desynchronization (DESYNC) separates the oscillators as far as possible

from each other. As a result, the phase differences of all the nodes become the same. In DESYNC, the synchronization target is not the phase value itself; instead, it is the phase difference between the nodes. A distributed DESYNC algorithm based on the PCO model was suggested for the desynchronizing of time event [25, 26]. As shown in Figure 9, each oscillator detects the firing of another oscillator immediately before and immediately after its own firing. It adjusts its own phase to the medium position of the firing time of its two neighbors. This process is repeated and DESYNC is completed.

A round-robin scheduling method that fairly allocates resources in a distributed manner without access collision was suggested by applying the DESYNC algorithm to time-division multiple access (TDMA) scheduling [25, 26]. DESYNC-based TDMA scheduling enables high throughput and low control overhead, even in a heavy load, because it does not induce signaling messages for scheduling. Furthermore, a distributed scheduling method was proposed by applying DESYNC to multihop networks. This approach flexibly controls the transmission intervals between nodes in a situation wherein the nodes randomly enter or exit [27]. Lastly, proportional fair scheduling was achieved by enabling each node to operate two oscillators through the expansion of the DESYNC algorithm [28].

**3.3. Distributed Fusion in Sensor Networks.** In a sensor network, it is necessary to reduce the amount of transmitted information through the fusion of sensed information in intermediate nodes, rather than to concentrate all information sensed from each node to a center and simultaneously compute it. In particular, as the number of nodes increases, the amount of sensed information to transmit rapidly increases. Thus, the fusion of sensed information is more necessary in various parts of the network. The fusion of sensed information is a process in which multiple sensor nodes cooperatively make a decision. This process is used to extract a useful piece of information by reaching a consensus among the nodes.

Various methods that compute the average in a distributed manner in sensor networks were suggested to realize the Kalman filter in this context [29, 30]. Additionally, the linear least-squares estimator (LSE) method was proposed to compute an average based on the consensus among network nodes [31]. Lastly, to more efficiently compute the average of sensed information, low and high pass consensus filters were suggested [32].

**3.4. Network Design for Fast Synchronization.** Nature-inspired synchronization algorithms require iterations and thus require time to obtain the desired convergence value. To overcome this constraint, the development of a network design that achieves fast synchronization is an active research topic. In [33], for example, a study on shortening the convergence time of a synchronization algorithm was presented. The convergence time was reduced by varying the weight, which is multiplied for each operation when the average is calculated.

In another study, the weight was fixed and the network topology was varied to stimulate faster synchronization [34, 35]. When a small-world network was created by rewiring random nodes in a given network topology, it was confirmed that the synchronization speed is dramatically faster. Figure 10 shows the difference in convergence times between a regular network, in which 100 nodes are wired within three hops, and a small-world network, wherein an additional connection is made between random 300 nodes on the regular network.

**3.5. Distributed Formation Control.** The multivehicle system is an important networking system for commercial and military applications. Controlling the number of vehicles to be automatically driven in a certain formation requires cooperation between nodes. It is thus necessary for the vehicles to synchronize to achieve their collective mission. It was proven that distributed multivehicle formation control is a matter of synchronization and a related theoretical framework was developed [36].

In addition, the matter of a space rendezvous is regarded as a synchronization problem concerning the position of various individuals that interact [37, 38]. In the rendezvous problem, the occurrence of convergence according to the network topology is important; however, it is irrelevant to the convergence value. Furthermore, a theoretical framework was developed for the design and analysis of a flocking algorithm that avoids collision with obstacles [39]. For flocking,

the relation with obstacles should be considered as well as the interaction between individuals. In this study, the synchronization algorithm is used to enable an individual to avoid colliding with obstacles and other individuals based only on the information of the surroundings, while synchronizing the velocities of all individuals.

**3.6. Solution for Optimization Problems.** Various metaheuristic optimization methods based on swarm intelligence were suggested, some examples of which include particle swarm optimization, ant colony optimization, and bee colony optimization [40]. In this type of optimization technique, numerous particles repeatedly explore a multidimensional solution space by exchanging information with each other to determine the most optimal value. This approach is similar to the synchronization phenomenon in the sense that various individuals interact in a distributed manner and converge toward a single optimal point. However, it does not align with the basic synchronization algorithm, which obtains the average of the local information of interest.

For a certain optimization problem, it was proven that the optimal solution is to synchronize specific variables [41–43]. When the value sought by each individual has a “solidarity” property (i.e., the increase/decrease of the value of an individual causes the decrease/increase of the values of the others) and the objective problem is to maximize the minimum of these values (i.e., the max–min objective problem), it was proven that the optimal solution is to make these values equal, that is, to synchronize them.

Based on the above proposition, [41] suggested a distributed transmit-power control algorithm that maximizes the end-to-end throughput in wireless multihop networks and [43] proposed a distributed transmit-power control algorithm that minimizes the outage probability in wireless cellular networks. The rate of each link in the wireless network has a solidarity property on account of mutual interference. Moreover, the end-to-end throughput in multihop networks is decided by the minimum value of the rates of links consisting of a multihop path. Thus, the maximization problem of multihop end-to-end throughput has the following max–min objective problem:

$$\max_{\mathbf{P}} R_{e2e} = \max_{\mathbf{P}} \min \{R_{12}, R_{23}, \dots, R_{n(n+1)}\}, \quad (10)$$

where vector  $\mathbf{P}$  indicates the transmit power set of  $n$  transmitting nodes and  $R_{ij}$  denotes the rate of each link on the multihop path. Therefore, in this problem, the optimal solution is to make the transmission rate of each link equal. To this end, a synchronization algorithm of each link rate based on the Cucker–Smale flocking model [16] was proposed as

$$R_{ij}(t+1) - R_{ij}(t) = \frac{1}{n_{\forall k,l=k+1}} \sum \Psi(|x_k - x_l|) (R_{kl}(t) - R_{ij}(t)). \quad (11)$$

To support the link rate determined by (11), each transmitting node performs the transmit power control in each time slot.

In both the maximization problem of multihop end-to-end throughput in [41] and the minimization problem of

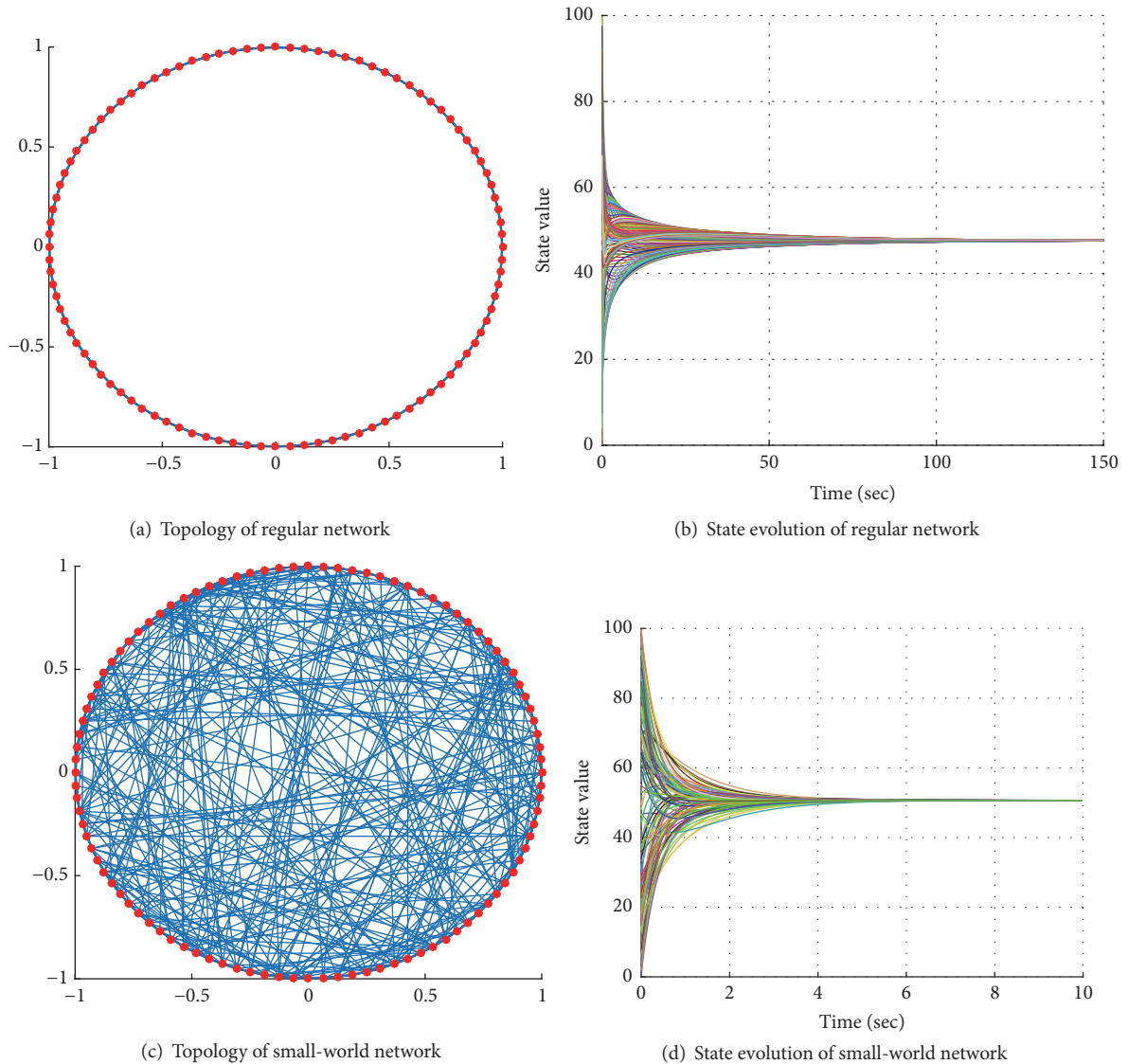


FIGURE 10: Comparison of convergence times between a regular network and a small-world network.

outage in [43], the synchronization phenomenon of each link rate and the transmission power variation in each node over time are depicted in Figure 11. The suggested algorithm basically follows the Cucker–Smale flocking model. Therefore, the convergence is guaranteed. The results also show that when the link transmission rates converge, the node with the best initial link rate uses the lowest transmit power, while the node with the worst initial link rate uses the maximum transmit power. Because the higher transmit power induces more interference, the optimal strategy is to enable the transmitting node with the best link to use the lowest transmit power, thereby causing the least interference and increasing the transmission rates of the other links.

The above reasoning is supported by the notion that the strongest bird positions itself at the head of the flock, where the air resistance is the greatest, in order to support the weaker birds that follow when migrating as a flock. The velocity vectors of birds have a solidarity property owing

to the air resistance. Additionally, they have a max–min objective because their flying speed is limited by the speed of the slowest bird. Accordingly, we can understand that the synchronization of velocity in the flights of flocking birds is the optimal solution for achieving their objective [42, 43].

*3.7. Comparisons between Nature-Inspired and Conventional Synchronizations.* In this section, the nature-inspired and conventional approaches are compared in terms of time synchronization. The characteristics, advantages, and disadvantages of nature-inspired time synchronization methods are evaluated compared with the conventional centralized and distributed time synchronization methods.

Centralized time synchronization methods use global time orientations, such as global positioning system (GPS) time [44] and network time protocol (NTP) [45]. All nodes listen for the reference time provided from the central clock and adjust their own time in order to synchronize. This



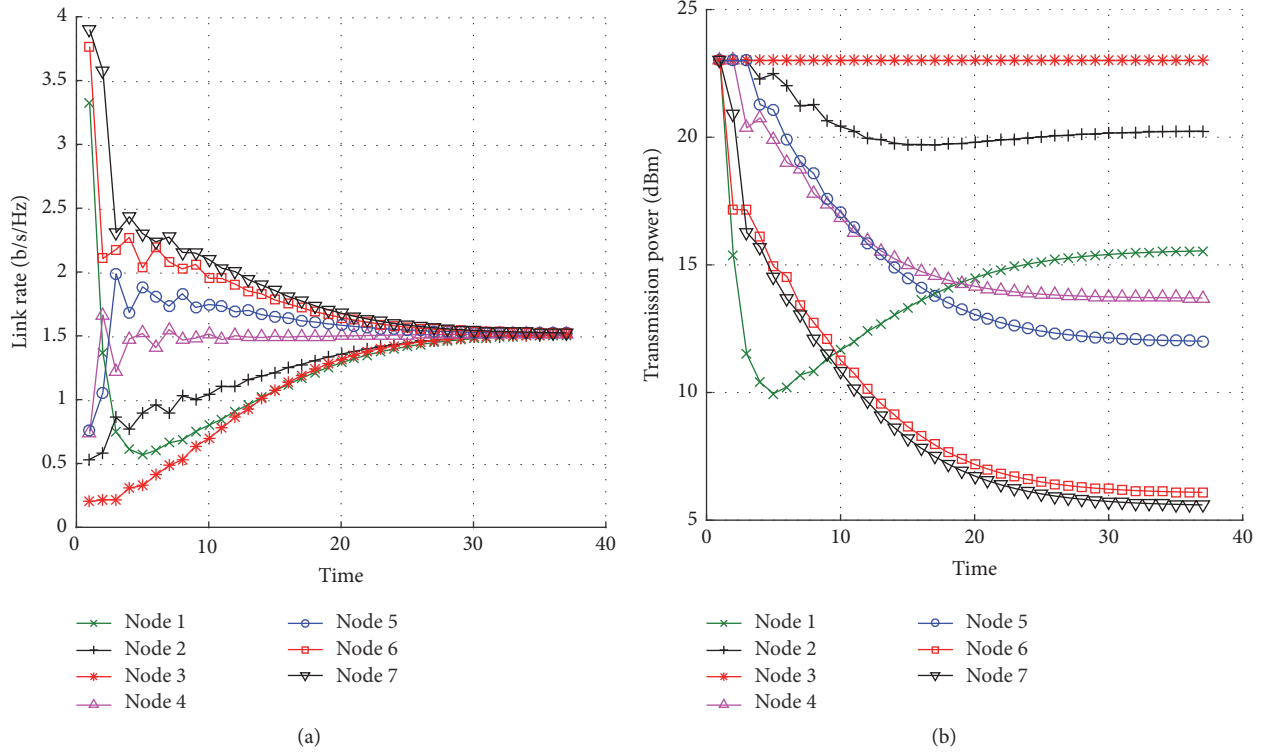


FIGURE 11: (a) Link rate and (b) transmission power over time for the maximization problem of end-to-end throughput in wireless multihop networks.

process enables accurate synchronization with high reliability; however, it involves a high cost in terms of hardware implementation and energy consumption. Moreover, the synchronization performance is related to the respective distance to the central clock, which limits the network scalability.

The conventional reference broadcast synchronization (RBS) method [46] is a typical distributed time synchronization method that is not nature-inspired. Without using a central global clock, RBS engenders mutual synchronization among the network nodes. Thus, precise synchronization is possible without a delay from the central clock. However, the time information of each node should be shared with all the other nodes so that an immense amount of data can be exchanged between the nodes. As the number of nodes increases, multihop RBS is required and the data exchange overhead likewise increases, resulting in the decrease of performance.

Unlike these methods, nature-inspired time synchronization methods [19–24] can achieve synchronization in a distributed and efficient manner regardless of the number of nodes in a large-scale network. Table 1 summarizes the characteristics, advantages, and disadvantages of the conventional synchronization methods and those of the nature-inspired synchronization methods.

#### 4. Research Challenges

In this section, we examine further research challenges that should be considered in utilizing natural synchronization phenomena for the future communication and networking

systems. The considered research challenges are classified as in Figure 12.

**4.1. Challenges for Convergence Performance.** The synchronization algorithm conducts iterations in a dynamic environment. Thus, the performance related to convergence is very important. According to the number of nodes, initial value, network topology, and coupling strength, we must theoretically analyze the existence, value, and rate of convergence, or we must empirically recognize the tendency of convergence. To increase the convergence rate, we must design a network topology that can stimulate rapid synchronization based on the results of many experiments in a variety of network environments. These include random, small-world, and large-scale networks, as well as locally connected regular networks.

According to studies conducted thus far, a network wherein all nodes are fully connected is known to guarantee the convergence regardless of the number of nodes and the initial value [9]. Therefore, in order to guarantee the convergence, it is necessary to arbitrarily make the network fully connected. Alternatively, a hierarchical method can be considered in which a network is clustered as a fully connected network at a local level. Then, synchronization is engaged at the upper level with the locally converged information.

The convergence rate has a tradeoff relationship with the accuracy of the convergence value. If the convergence value is set at a discrete level instead of a real value from a practical perspective, the convergence rate can be controlled according to the number of levels. For example, the convergence value in the synchronization of transmission rate is mapped to

TABLE 1: Comparison of conventional and nature-inspired synchronization methods.

Item	Conventional synchronization method	Nature-inspired synchronization method
Reliability	High reliability and accuracy; short convergence time	Guaranteed reliability; accurate synchronization in large-scale networks
Scalability	Limited by the numbers of nodes and hops	Scalability maintained regardless of the number of nodes and changes
Robustness	Vulnerable to network topology changes, such as cluster head failures	Robust to poor communication environments and network topology changes
Cost	Very high; message exchange of upper layers is required	Low amount of data exchange; can be simply operated through pulse exchange at the physical layer without upper layer protocols
Complexity	Very high hardware (HW)/software (SW) complexity	Low HW/SW complexity (not necessary to save time information of other nodes in the memory)
Influential factors for performance	Synchronization accuracy depends on the distance to root nodes	Affected by node density, coupling strength, delay, noise, path loss, and modulation method
Advantage	High reliability; effective with an appropriate number of nodes; not dependent on the global clock in the case of RBS	High scalability and robustness; low cost and complexity; effective with a large number of nodes
Disadvantage	Poor performance of outer nodes in the topology in cases of centralized synchronization; a large amount of exchanged data is required in distributed synchronization	Synchronization is impossible when the transmission and reception delays take a long time; synchronization speed decreases with weak coupling strength and a small number of nodes

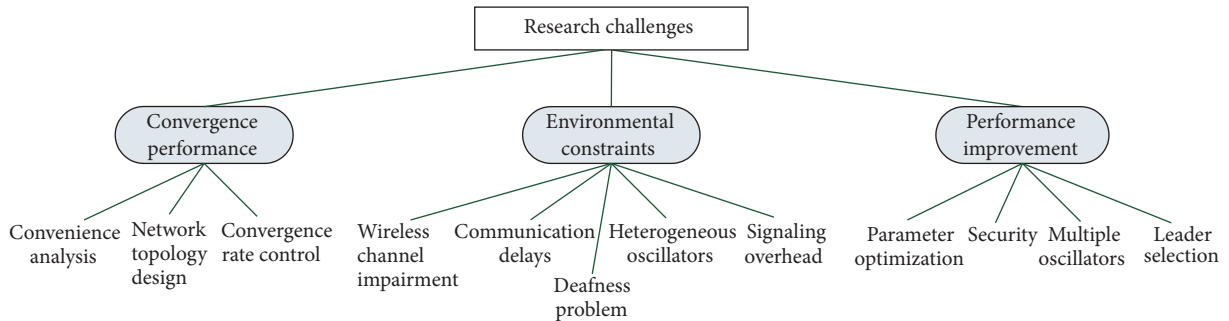


FIGURE 12: Classification of research challenges.

the modulation and coding scheme (MCS) level in reality. Thus, the convergence rate can be controlled according to the number of MCS levels. On the contrary, the number of MCS levels can be determined according to the required convergence rate.

**4.2. Challenges of Environmental Constraints.** Path loss, noise, delay, and packet loss occurring in real environments influence synchronization. The weakening of coupling strength due to path loss and packet loss reduces the convergence rate, while delay and noise are major factors that interfere with synchronization [10]. When applying a synchronization algorithm to a real environment, it is therefore necessary to design the algorithm with consideration of all these interfering factors.

A receiving node may be unable to immediately respond on account of a communication delay. Consequently, the synchronization may not be completed. Specifically, the communication delay can be divided into a transmission delay,

decoding delay, and propagation delay. It is necessary to redesign a synchronization algorithm considering the statistical ranges of these delays. Moreover, a wireless communication node cannot receive data while simultaneously transmitting. Thus, it cannot recognize the firing of other nodes while it is firing. Therefore, the firing period should be set to cross one another to avoid such “deafness” problems in wireless communication environments.

To date, research has been conducted only on homogeneous oscillators that follow the same rule. However, in reality, heterogeneous oscillators may exist that adhere to different synchronization rules in the same situation on account of the network heterogeneity. Therefore, research should be conducted on synchronization algorithms when the heterogeneous oscillators are intended to cooperatively achieve a certain collective purpose.

For the real operation of the synchronization algorithms described in Section 2, periodic data exchange among neighboring nodes is required. Consequently, signaling overhead

occurs. Considering this kind of overhead cost, the point at which it is more effective to apply nature-inspired synchronization methods over conventional synchronization methods should be determined. Depending on the situation, a hybrid synchronization algorithm can be considered that adaptively applies both types of methods.

*4.3. Challenges to Performance Improvement.* Depending on network topology changes in the wireless environment, synchronization performance can be improved by optimizing the operation parameters of the synchronization algorithm, such as the coupling strength, weight, and step size. The mapping relations between operation parameters and environment variables should be configured through various experiments.

In terms of network security, a method is required to enhance the tolerance of systems against malicious nodes that intentionally interfere with the synchronization. For example, a synchronization algorithm can be designed to ignore certain information with the maximum deviation from the average of the information received from neighboring nodes. A synchronization algorithm that considers unnecessary information should be devised and it should be verified whether it is possible to achieve a desired convergence performance.

When there are two or more oscillators in a single node, a method that can utilize multiple oscillators should be considered. In [28], proportional fair scheduling was conducted in a distributed manner using two oscillators. Therefore, complex synchronization phenomena with several oscillators need to be studied for developing a new application that utilizes them.

It is possible to converge the values of all nodes into a desired value by selecting a leader and setting only its value. For example, in aircraft engineering, by controlling the single leader in flock-like flight formation, this method can enable all the rest of the aircrafts to automatically follow that specific aircraft. In addition, the transmission rate of all nodes can be equalized by controlling only the rate of the leader. However, this kind of approach that controls the convergence value only through a leader may be unable to guarantee the convergence depending on the environment. Thus, a theoretical study to address that issue should be conducted.

## 5. Discussion and Prospect

The reason why we focus on synchronization phenomena can be explained as follows. First of all, the natural environment in which synchronization occurs is similar to the environments in which many engineering problems occur. Synchronization occurs among a number of individuals. It also occurs in a completely distributed system in which each individual decides its own action with consideration of only the surrounding environment or situation. In addition, individuals perfectly synchronize in dynamic environments where obstacles or adversaries may appear, and nodes enter or exit. These kinds of environments in which synchronization occurs inspire many engineering problems in large-scale networking, distributed, and dynamic environments.

Secondly, the purpose of synchronization in nature can be understood as the same as engineering objectives. Insects,

birds, fish, and other animals that form groups are known to synchronize for migration, mating, protection from predators, and efficient searching for food [3]. These creatures exhibit solidarity and thus engage in synchronization for collective goals. These goals are typically for survival purposes. Such animal populations use synchronization as an optimal strategy for surviving while evolving through time. Likewise, a number of nodes in systems, such as sensor networks, multivehicle systems, and multihop network, must achieve a collective objective with solidarity. Their objectives are to rapidly obtain useful information, quickly move together over long distances, and maximize the end-to-end throughput, respectively. These goals are not for maximizing the performance of individual nodes; rather, they serve to maximize the performance of the entire system. Therefore, the purposes of engineering systems correspond to those demonstrated by collectives of creatures. It is thus reasonable to employ synchronization as the optimal solution to problems that exist in engineering systems.

Lastly, synchronization in nature is more efficient than any centralized or distributed system available. For example, thousands of fireflies engender the synchronization phenomenon in the fastest and most efficient way with limited resources. They use a very simple method and achieve synchronization within a short period of time. Therefore, mimicking these natural approaches can enable engineering systems to achieve performances at a similar level of low complexity and cost in situations with limited resources.

The point at which nature-inspired technologies will become more effective than existing approaches will be when the number of nodes comprising a system increases beyond a certain level. Owing to the advancement of communication network technologies, the number of nodes connected to networks is drastically increasing each year. The increase of nodes accelerates the shortage of resources and incurs the high cost of centralized management. Conventional centralized technologies developed in consideration of a limited number of nodes can neither operate with a large number of nodes nor guarantee the required performance. Thus, naturally, a shift to distributed systems is required. However, distributed system technologies developed under the centralized paradigm still bear the limitation concerning the increase in the number of nodes. Furthermore, they are vulnerable to environmental changes due to the incompleteness of the distribution [46]. Therefore, a paradigm shift to new distributed systems is necessary and it is imperative to focus on the distributed algorithms of collective organisms in nature as an effective methodology to be applied.

## 6. Conclusion

In this article, we identified the principles of various natural synchronization phenomena and studied how they can be applied in practical communication and networking systems. In terms of time synchronization methods in networks, the strengths and weaknesses of conventional approaches and nature-inspired approaches were compared. Further research challenges that are necessary for realizing nature-inspired synchronization technologies were presented. The

rapid increase of the number of networking nodes will accelerate the occurrence of complex and distributed network environments, which are getting similar to the environment of nature. Therefore, it would be a great help for us to mimic and apply the fundamental principles of biocommunities in which each entity has shown collective behaviors to sustain their lives in the midst of complex and chaotic environments. In the future, various studies considering practical issues are required to effectively apply these principles to mobile communication and networking systems.

## Competing Interests

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

## Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2016R1C1B1016261), by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2015R1D1A1A01060207), and by the Human Resources Development of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea government Ministry of Trade, Industry and Energy (no. 20154030200860).

## References

- [1] A. Pikovsky, M. Rosenblum, and J. Kurths, *Synchronization: A Universal Concept in Nonlinear Sciences*, Cambridge University Press, 2001.
- [2] H.-H. Choi and J.-R. Lee, "Communication and networking technologies based on bio-inspired algorithms," *KICS Information & Communications Magazine*, vol. 29, no. 4, pp. 62–71, 2012 (Korean).
- [3] S. H. Strogatz, *Sync: How Order Emerges from Chaos in the Universe, Nature, and Daily Life*, Hyperion, New York, NY, USA, 2004.
- [4] F. Dressler and O. B. Akan, "A survey on bio-inspired networking," *Computer Networks*, vol. 54, no. 6, pp. 881–900, 2010.
- [5] Z. Zhang, K. Long, J. Wang, and F. Dressler, "On swarm intelligence inspired self-organized networking: its bionic mechanisms, designing principles and optimization approaches," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 513–537, 2014.
- [6] C. Zheng and D. C. Sicker, "A survey on biologically inspired algorithms for computer networking," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1160–1191, 2013.
- [7] C. W. Reynolds, "Flocks, herds, and schools: a distributed behavioral model," *ACM Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.
- [8] C. S. Peskin, "Mathematical aspects of heart physiology," Tech. Rep., Courant Institute of Mathematical Sciences, New York University, 1975.
- [9] R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM Journal on Applied Mathematics*, vol. 50, no. 6, pp. 1645–1662, 1990.
- [10] U. Ernst, K. Pawelzik, and T. Geisel, "Synchronization induced by temporal delays in pulse-coupled oscillators," *Physical Review Letters*, vol. 74, no. 9, pp. 1570–1573, 1995.
- [11] Y. Kuramoto and H. Araki, Eds., *Lecture Notes in Physics, International Symposium on Mathematical Problems in Theoretical Physics*, Springer, New York, NY, USA, 1975.
- [12] S. H. Strogatz, "From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators," *Physica D. Nonlinear Phenomena*, vol. 143, no. 1-4, pp. 1–20, 2000.
- [13] J. A. Acebrón, L. L. Bonilla, C. J. Perez-Vicente, F. Ritort, and R. Spigler, "The Kuramoto model: a simple paradigm for synchronization phenomena," *Reviews of Modern Physics*, vol. 77, no. 1, pp. 137–185, 2005.
- [14] R. Sepulchre, D. Paley, and N. Leonard, "Collective motion and oscillator synchronization," in *Proceedings of the Block Island Workshop Cooperative Control*, Block Island, RI, USA, June 2003.
- [15] A. Papachristodoulou and A. Jadbabaie, "Synchronization in oscillator networks: switching topologies and non-homogeneous delays," in *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference (CDC-ECC '05)*, pp. 5692–5697, December 2005.
- [16] F. Cucker and S. Smale, "Emergent behavior in flocks," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 852–862, 2007.
- [17] S.-Y. Ha and J.-G. Liu, "A simple proof of the Cucker-Smale flocking dynamics and mean-field limit," *Communications in Mathematical Sciences*, vol. 7, no. 2, pp. 297–325, 2009.
- [18] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [19] A. Tyrrell, G. Auer, and C. Bettstetter, "Fireflies as role models for synchronization in ad hoc networks," in *Proceedings of the 1st Bio-Inspired Models of Network, Information and Computing Systems*, pp. 1–7, Madonna di Campiglio, Italy, December 2006.
- [20] A. Tyrrell and G. Auer, "Imposing a reference timing onto firefly synchronization in wireless networks," in *Proceedings of the IEEE 65th Vehicular Technology Conference (VTC '07)-Spring*, pp. 222–226, April 2007.
- [21] Y.-W. Hong and A. Scaglione, "A scalable synchronization protocol for large scale sensor networks and its applications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 5, pp. 1085–1099, 2005.
- [22] G. A. Puerta, E. A. Aguirre, and M. A. Alzate, "Effect of topology and mobility in bio-inspired synchronization of mobile ad hoc networks," in *Proceedings of the IEEE Latin-American Conference on Communications (LATINCOM '10)*, pp. 1–6, September 2010.
- [23] H. Yoo, M. Lee, and Y. Cho, "A distributed frequency synchronization technique for OFDMA-based mesh networks using bio-inspired algorithm," *The Journal of Korea Information and Communications Society*, vol. 37B, no. 11, pp. 1022–1032, 2012.
- [24] G. Werner-Allen, G. Tewari, A. Patel, M. Welsh, and R. Nagpal, "Firefly-inspired sensor network synchronicity with realistic radio effects," in *Proceedings of the 3rd ACM International Conference on Embedded Networked Sensor Systems (SenSys '05)*, pp. 142–153, ACM, San Diego, Calif, USA, November 2005.
- [25] J. Degeysys, I. Rose, A. Patel, and R. Nagpal, "DESYNC: self-organizing desynchronization and TDMA on wireless sensor networks," in *Proceedings of the 6th International Symposium on*

- Information Processing in Sensor Networks (IPSN '07)*, pp. 11–20, April 2007.
- [26] A. Patel, J. Degeysys, and R. Nagpal, “Desynchronization: the theory of self-organizing algorithms for round-robin scheduling,” in *Proceedings of the 1st International Conference on Self-Adaptive and Self-Organizing Systems (SASO '07)*, pp. 87–96, Boston, Mass, USA, July 2007.
- [27] J. Degeysys and R. Nagpal, “Towards desynchronization of multi-hop topologies,” in *Proceedings of the 2nd IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO '08)*, pp. 129–138, Venice, Italy, October 2008.
- [28] R. Pagliari, Y.-W. P. Hong, and A. Scaglione, “Bio-inspired algorithms for decentralized round-robin and proportional fair scheduling,” *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 4, pp. 564–575, 2010.
- [29] R. Olfati-Saber and J. S. Shamma, “Consensus filters for sensor networks and distributed sensor fusion,” in *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference (CDC-ECC '05)*, pp. 6698–6703, December 2005.
- [30] D. P. Spanos, R. Olfati-Saber, and R. M. Murray, “Approximate distributed kalman filtering in sensor networks with quantifiable performance,” in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 133–139, Los Angeles, Calif, USA, April 2005.
- [31] L. Xiao, S. Boyd, and S. Lall, “A Scheme for robust distributed sensor fusion based on average consensus,” in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 63–70, April 2005.
- [32] D. Spanos, R. Olfati-Saber, and R. M. Murray, “Dynamic consensus on mobile networks,” in *Proceedings of the 16th International Federation of Automatic Control World Congress (IFAC '05)*, Prague, Czech Republic, 2005.
- [33] L. Xiao and S. Boyd, “Fast linear iterations for distributed averaging,” *Systems and Control Letters*, vol. 53, no. 1, pp. 65–78, 2004.
- [34] D. J. Watts and S. H. Strogatz, “Collective dynamics of ‘small-world’ networks,” *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
- [35] R. Olfati-Saber, “Ultrafast consensus in small-world networks,” in *Proceedings of the American Control Conference (ACC '05)*, pp. 2371–2378, June 2005.
- [36] J. A. Fax and R. M. Murray, “Information flow and cooperative control of vehicle formations,” *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.
- [37] J. Lin, A. S. Morse, and B. D. O. Anderson, “The multi-agent rendezvous problem,” in *Proceedings of the 42nd IEEE Conference on Decision and Control*, pp. 1508–1513, Maui, Hawaii, USA, December 2003.
- [38] J. Cortes, S. Martinez, and F. Bullo, “Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions,” *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 51, no. 8, pp. 1289–1298, 2006.
- [39] R. Olfati-Saber, “Flocking for multi-agent dynamic systems: algorithms and theory,” *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.
- [40] X.-S. Yang, Z. Cui, R. Xiao, A. H. Gandomi, and M. Karamanoglu, *Swarm Intelligence and Bio-Inspired Computation, Theory and Applications*, Elsevier, 1st edition, 2013.
- [41] H.-H. Choi and J.-R. Lee, “Distributed transmit power control for maximizing end-to-end throughput in wireless multi-hop networks,” *Wireless Personal Communications*, vol. 74, no. 3, pp. 1033–1044, 2014.
- [42] H.-H. Choi and J.-R. Lee, “A bio-inspired transmit power control algorithm for linear multi-hop wireless networks,” in *Proceedings of the IARIA International Conference on Networks (ICN '14)*, February 2014.
- [43] H. Choi and J. Lee, “A biologically-inspired power control algorithm for energy-efficient cellular networks,” *Energies*, vol. 9, no. 3, article no. 161, 2016.
- [44] J. Mannermaa, K. Kalliomaki, T. Mansten, and S. Turunen, “Timing performance of various GPS receivers,” in *Proceedings of the IEEE International Frequency Control Symposium, 1999. Proceedings of the 1999 Joint Meeting of the European*, pp. 287–290, April 1999.
- [45] D. L. Mills, “Internet time synchronization: the network time protocol,” *IEEE Transactions on Communications*, vol. 39, no. 10, pp. 1482–1493, 1991.
- [46] J. Elson, L. Girod, and D. Estrin, “Fine-grained network time synchronization using reference broadcasts,” *SIGOPS Operating Systems Review*, vol. 36, pp. 147–163, 2002.

## Review Article

# Delay Tolerance in Underwater Wireless Communications: A Routing Perspective

**Safdar Hussain Bouk, Syed Hassan Ahmed, and Dongkyun Kim**

*School of Computer Science and Engineering, Kyungpook National University, Daegu, Republic of Korea*

Correspondence should be addressed to Dongkyun Kim; [dongkyun@knu.ac.kr](mailto:dongkyun@knu.ac.kr)

Received 24 September 2016; Accepted 21 November 2016

Academic Editor: Yujin Lim

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

Similar to terrestrial networks, underwater wireless networks (UWNs) also aid several critical tasks including coastal surveillance, underwater pollution detection, and other maritime applications. Currently, once underwater sensor nodes are deployed at different levels of the sea, it is nearly impossible or very expensive to reconfigure the hardware, for example, battery. Taking this issue into account, considerable amount of research has been carried out to ensure minimum energy costs and reliable communication between underwater nodes and base stations. As a result, several different network protocols were proposed for UWN, including MAC, PHY, transport, and routing. Recently, a new paradigm was introduced claiming that the intermittent nature of acoustic channel and signal resulted in designing delay tolerant routing schemes for the UWN, known as an underwater delay tolerant network. In this paper, we provide a comprehensive survey of underwater routing protocols with emphasis on the limitations, challenges, and future open issues in the context of delay tolerant network routing.

## 1. Introduction

Communication and monitoring of underwater environment have been long-standing challenges in a vast range of applications that are enabled by underwater wireless communications. The scope of these applications ranges from monitoring ocean pollution to environment, climate, natural disturbances, and marine ecosystem for search and survey operations, surveillance, offshore exploration, navigation, and disaster prevention. Applications such as the above mentioned have led to a flurry of research investigating how better communication and monitoring could be achieved in underwater wireless communications [1].

Networks that enable underwater communication are called underwater wireless networks, underwater acoustic networks, or underwater acoustic sensor networks, as shown in Figure 1. However, in this context, we refer to them collectively as underwater wireless networks (UWNs). A UWN is composed of offshore and onshore static and mobile nodes. The offshore static nodes are battery operated, with sensing capability, anchored to or resting on the seabed, and floating at different heights. The other set of nodes floating

over the sea surface or under the sea, either autonomously or manually controlled, are called offshore mobile nodes [2].

The main objective of UWNs is to sense and monitor underwater environment as well as control autonomous underwater vehicles (AUVs) and provide this information to an offshore center through intermediate data collection points or sinks floating over the sea surface [3]. Each UWN node below the sea surface is equipped with sensing functionality to collect features and parameters of the underwater environment. This sensed data is forwarded to the onshore stations through stations floating over the sea surface (e.g., boats or buoys) in a multihop fashion [4].

Diverse communication modes, for instance, radio frequency (RF) electromagnetic waves [5] and acoustic and/or optical signals [6], are used between offshore surface stations and onshore data centers as well as among offshore nodes. RF electromagnetic waves are used to enable communication between nodes that are floating over the sea surface and onshore data collection and control center. However, the same cannot be applied to propagate information between the wireless nodes below the sea surface because sea water is not a good conductor of electromagnetic waves due to

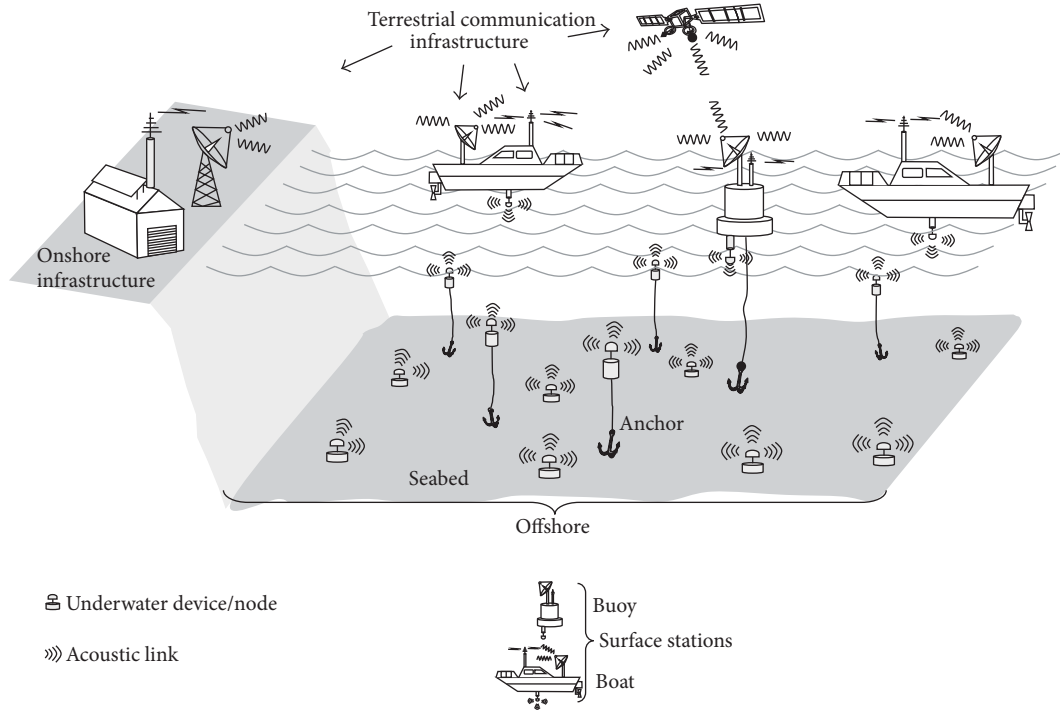


FIGURE 1: Underwater wireless communications network.

salinity. The alternative communication mode in underwater environment is optical communication. However, in a turbid underwater environment, optical communication severely suffers from high scattering, which makes it impracticable for underwater communication.

Other than the RF electromagnetic and optical signals, acoustic signals are employed as an enabling communication mode in the UWNs. Acoustic signals can propagate efficiently in the aquatic environment for very long distances (measured in kilometers). However, acoustic signals and aquatic channels collectively put forward many constraints and challenges that need to be overcome. These constraints pose an uncertain end-to-end path between source and destination nodes as well as frequent network partitions and intermittent links between nodes. This makes real-time communication a challenging task in UWNs [7]. Hence, it is necessary to get aware of the current state-of-the-art routing solutions and understand the limitations of an acoustic channels that hinder the data forwarding in such a constrained environment. In this article, we therefore provide a comprehensive overview of routing in underwater communications. The key contributions of this article are as follows:

- (i) An overview of the limitations of acoustic signals and channels (Section 2)
- (ii) A brief discussion on the traditional routing classes for UWN (Section 3)
- (iii) An introduction to the underwater DTN and state-of-the-art routing solutions (Section 4)

- (iv) Detailed discussion and unique classification and comparison of underwater DTN routing protocols (Section 5)
- (v) Enlisted shortcomings of existing solutions along with the future research directions (Section 6)

## 2. UWN Limitations

The comprehensive list of acoustic signals and aquatic channel limitations of the UWN [8] is as follows.

- (1) *Propagation Delay*. The main constraint of acoustic signals in underwater is its high propagation delay that is approximately  $2 \times 10^5$  times slower than the electromagnetic propagation in terrestrial environments. The acoustic delay is highly dynamic and contingent on different factors that include water temperature, depth, and salinity.
- (2) *Multivariate Attenuation*. The acoustic signal faces severe path loss that is inherent in the acoustic channel and mainly depends on signal frequency, absorption loss, sender-receiver distance, and path-loss exponent.
- (3) *Effects of Noise*. The frequency band of the acoustic channel is constrained by the effect of ambient noise in the aquatic region. There are three main categories of ambient noise sources: (i) marine animals, including Dolphins, whales, and snapping shrimp., (ii) human activity that consists of on- and offshore exploration, industrial activity, ships, fishing, and

communication, and (iii) natural events, ranging from underwater earthquakes to ice cracking, volcano eruption, bubbles, rain, lightning, and wind to the surface waves.

- (4) *Limited Bandwidth*. The acoustic bandwidth is reduced to a small range by varying factors that include high impact of sound at lower frequencies and large medium absorption at higher acoustic frequencies.
- (5) *Time-Varying Multipath Propagation*. Each acoustic ray generated by an acoustic source follows a different path with varying delays. Multiple paths are followed by each ray due to boundary reflections (e.g., sea surface, seabed) or other objects suspended in the sea water, and refraction is caused by varying sound profiles at different depths. In response, the distant receiver will receive multiple copies with varying strengths and delays that result in unstable phase response.
- (6) *High Transmission Power*. High transmission power is required to propagate the acoustic signal, which constrains the network lifetime.
- (7) *Bit Error Rate*. High bit error rate occurs as a result of time-varying characteristics and multipath interference characteristics of the acoustic channel.
- (8) *Intermittent Connectivity*. Temporary loss of connectivity is experienced due to shadow zones (the regions where acoustic signal does not reach or intensity of the signal is very low).
- (9) *Limited Energy Harvesting Options*. Energy harvesting is difficult in the underwater environment, which restricts the UWN nodes to limit their wireless communication duration to prolong the network lifetime.
- (10) *No Position Information*. Unpredictable ocean currents and intrinsic characteristics of the aquatic channel make it difficult to provide position information in UWNs.

The aforementioned constraints fail to provide deterministic end-to-end connectivity between source and destination and successive network partitions. With all these limitations, it is hard to achieve real-time communication. Therefore, the existing routing solutions for terrestrial networks cannot be directly applied to the UWNs. Therefore, a new paradigm of routing protocols that tolerates the delays and disruptions, called the delay tolerant network (DTN) routing in the UWN, has been explored by researchers.

Fall [9] introduced the concept of a Delay Tolerant Networks by proposing the architecture. In the DTN architecture, the authors characterized communication delay and frequent network partitioning due to the intermittent connectivity. Designing routing protocols for such networks is always a challenge as there is no continuous path from source to destination and hence conventional routing protocols usually fail. To the best of our knowledge, every DTN routing protocol follows a store-carry and forward mechanism, where sensor nodes need to keep messages in buffer until they

meet an intermediate node or destination directly (most of schemes do not consider direct delivery enabling relay based routing in DTN). A representative scheme for DTN store-carry and forward (SCAF) approach is Epidemic routing [10], which forwards each copy of message to each node. Epidemic routing scheme is flooding based in nature, thus exhausting network resources. In order to deal with a resource utilization issue, Wang et al. proposed Spray and Wait [11], in which only a limited number of message copies are replicated among nodes. One improvement in DTN for vehicles is MaxProp [12], where priority is given to nodes on some criterion and also messages are also characterized while forwarding.

Data forwarding in underwater DTN is being conducted with the SCAF mechanism. In case of unavailability of next-hop connectivity, an intermediate node stores the received packet(s). The packets are forwarded to the next-hop node when the opportunistic connectivity to the next-hop node is available. This type of approach provides eventual data delivery to the destination with a certain probability. The authors in [13] provide a survey of the generalized routing protocols for the underwater DTNs and classify them based on the nature of the link contact between the UWN nodes.

### 3. Routing Features in UWNs

Reliable and energy efficient data delivery with minimal delay from source to destination nodes in battery-operated multihop networks is prime design goals of the routing layer. An efficient routing protocol first discovers suitable path(s) and then forwards data towards the destination node by following the selective path(s) [14]. There is a plethora of routing solutions proposed for terrestrial multihop networks. Similar routing solutions have also been tested in the UWNs and have been shown to cause poor performance due to intrinsic characteristics of the UWNs.

In terrestrial multihop networks, the routing protocols are classified on the basis of network characteristics, route discovery, routing information management, and/or data communication mechanism. The classification parameters include *path discovery* mechanism (on-demand and periodic or reactive and proactive), *network topology* (flat and hierarchical), *topology management* (centralized and distributed), *number of paths* (single and multiple paths), *data communication mechanism* (unicast, multicast and broadcast), and *location information* (geocasting with or without flooding). Similarly, the classification of UWN routing protocols revolves around identical features, as shown in Figure 2, some of which are discussed below.

*Node Position/Localization*. Location information such as the global positioning system (GPS) is not directly available to the UWN nodes under the sea surface. However, the nodes floating at the sea surface receive location information and share it with the underwater nodes through acoustic beams. Similarly, the position information of the nodes below the sea surface is acquired through depth and pressure sensors. Routing protocols that perform data forwarding based on the above information are classified as location-based, depth-based, and pressure-based routing algorithms.



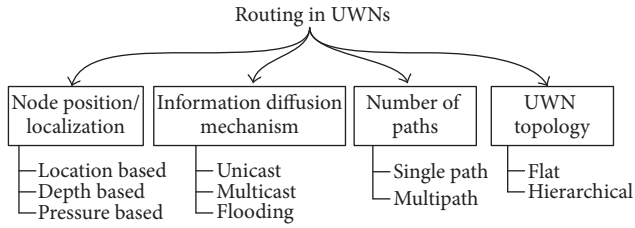


FIGURE 2: General classification of UWN routing protocols.

*Information Diffusion Mechanism.* Efficient information diffusion is the prime objective of any routing protocol. The UWN routing protocols use various information propagation methods and are classified accordingly. For example, information may be disseminated to single, multiple, or all the UWN nodes in unicast, multicast, or broadcast manner, respectively. However, each method has its own drawbacks. For example, high control overhead is required to maintain the UWN topology information in unicast, while broadcast or flooding consumes large amount of energy because of message duplication in the network.

*Number of Paths.* Data from the UWN source node(s) towards the destination or sink node are forwarded through single or multiple paths. The multiple path formation algorithms enhance reliability and robustness in data delivery in an intermittent communication environment such as the UWN.

*UWN Topology.* The UWN routing protocols are further classified in terms of network topology as flat or hierarchical. In a flat topology UWN, all nodes equally participate in the routing task, and the routing protocol has high control overhead to maintain topology information. In contrast, in hierarchical topology UWN, the routing protocols divide the network into small groups, called clusters, based on localization or other parameters. Each cluster is assigned a group head or cluster head that efficiently manages the network resources among the group members. As a result, spatial reuse of network resources can be achieved within the network, which alleviates network capacity.

*Mobility Influence.* To sense a vast underwater area, the underwater networks are sparsely deployed due to the high cost of underwater sensor nodes, which results in a delay or disruption tolerant underwater network. The sparsity may also have resulted due to oceanic and natural forces, for example, water currents, the wind, and water depth [15]. Therefore, the AUVs are used to collect data from the sparse network. Mostly, the AUVs have a planned network traversal path that is suitable for the static or quasi-static networks, to collect data from the network. This path may be computed based on the sufficient contact duration between the nodes to collect the data. However, the underwater sensor node mobility may disrupt this contact duration that will change the network performance. In addition to that, various underwater communication technologies have diverse characteristics of transmission range, bandwidth,

communication delay, and power consumption. For example, an acoustic channel has long range, small bandwidth, and high delay and radio frequency plus Magnetic induction has short range, high bandwidth, and moderate delay. Therefore, it is difficult to choose how fast or how far the communication is required. Thus, the mobility pattern of AUVs and the deployed underwater nodes have a significant impact on the performance of the routing schemes in the underwater DTN.

## 4. Routing in Underwater DTN

The concept of DTN refers to the network architecture that exhibits communication delay and frequent network partitioning due to intermittent connectivity. Designing routing protocols for such networks is always a challenge as there is no continuous path from source to destination and, hence, conventional routing protocols usually fail. Therefore, the recent paradigm shift drives us towards such routing solutions that ensure network performance at lower network resource costs.

Every underwater DTN routing protocol follows a SCAF mechanism, where underwater nodes need to store messages in their buffer until they encounter an intermediate node or destination directly, as shown in Figure 3. Therefore, each underwater DTN routing protocol must address the above constraints of DTN and provide energy efficiency, minimum end-to-end delay, and maximize overall network performance.

The authors in [13] survey the existing underwater DTN routing protocols and classify them based on the nature of the communication link between nodes. There are two main sets of routing protocols that assume that scheduled communication links or contact between nodes can be scheduled or unscheduled. Furthermore, the unscheduled contact schemes are subdivided into opportunistic and predicted contact. Please refer to [13] for more details.

In this survey, we provide an overview of recent advancements in underwater DTN routing schemes and classify them based on the forwarding and routing decisions, copies of messages, and performance metrics. Following is the detailed description of the classification parameters.

*Communication Initiation Mechanism.* Generally, data communication achieved by underwater network protocols is either Sender-initiated (push-based) or Receiver-initiated (pull-based). In the former, the data sender node selects its next-hop sensor node that relays the data further to the indicated destination. However, in the latter, the receiver node initiates the request to start the data communication.

*Infrastructure Assistance.* The underwater DTN routing schemes that perform routing decisions based on the information provided by the infrastructure nodes, for example, gateway(s), node(s) mounted on buoy(s), boats, and ships, are called infrastructure-assisted schemes. These infrastructure nodes periodically or on-demand maintain the network topology information and this information is made available to each node in the underwater DTN.

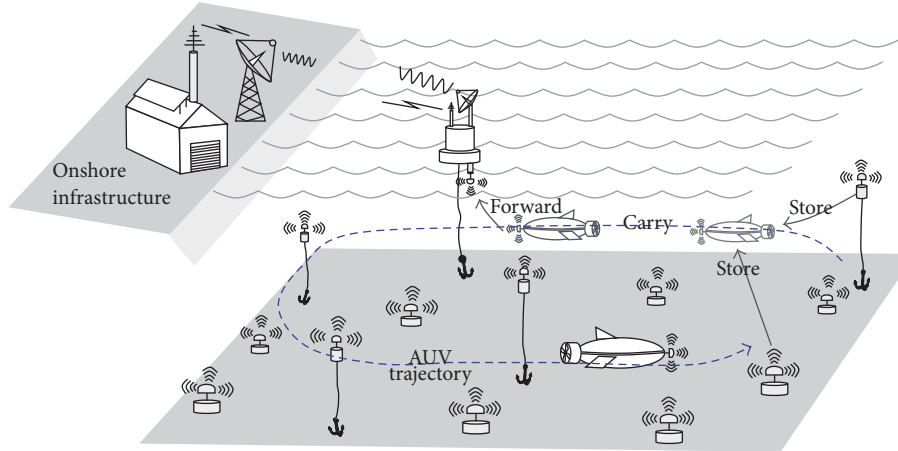


FIGURE 3: Store-carry-and-forward mechanism in UWNs.

*Routing Decision.* Routing decision is defined as the method by which an underwater DTN node selects next-hop based on local or global network topology information. We classify routing decisions into two main categories, namely, centralized or distributed. The routing protocols that perform routing decisions based on the global topology information are called centralized schemes. On the other hand, the underwater DTN nodes perform forwarding decisions subject to the locally discovered topology information. These schemes are termed as distributed routing schemes.

*Location Information Dependence.* The location information is one of the main routing metrics to carry out efficient routing. We consider the location information as one of the metrics to categorize the underwater DTN routing schemes.

*Number of Message Copies.* The source nodes in underwater DTN routing protocols generate and forward single or multiple copies of the same data messages in the network to ensure data delivery to the destination node.

*Performance Indicator.* Each routing protocol is proposed with a motive to improve network performance indicator metric(s). That performance metric can be throughput, end-to-end delay, control overhead, energy consumption, network lifetime, and so on.

Next, we briefly survey the state-of-the-art underwater DTN routing schemes by maintaining the above classification parameters, as shown in Figure 4, as a baseline for comparison.

### 5. Routing Protocols for Underwater Delay Tolerant Networks

Recently, many routing protocols for underwater DTN have been proposed to improve network performance under the adverse underwater environment. Following is a brief survey of recent progress in terms of research on routing protocols, especially for underwater DTNs.

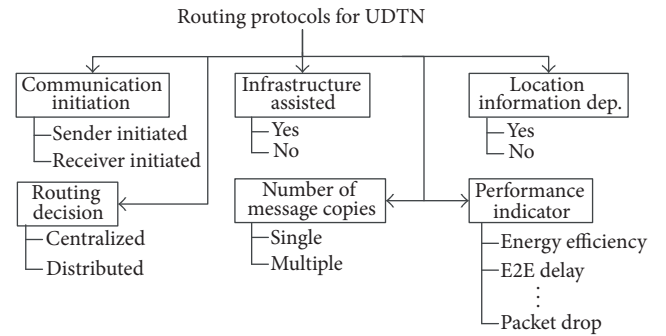


FIGURE 4: Classification parameters of underwater DTN routing protocols.

*5.1. PASR.* Guo et al. [16] proposed a generic scheme called prediction-assisted single-copy routing (PASR) for underwater DTNs to minimize delivery delay. PASR is a single-copy routing protocol where the authors divided routing operations into two phases, namely, learning and routing phases. In the learning phase, short duration traces are collected while connecting the network and characterization is performed by using an offline greedy algorithm. An aggressive chronological projected graph (ACPG) algorithm is used for the learning phase. As we know, if a mobility pattern of nodes is stable for a maximum time, its history can be predicted. In the routing phase, historical information such as a recent trajectory, duration of average contact and intercontact, and last contact and frequency of contacts is collected and used to predict mobility routes. On the basis of those patterns, a sender adaptively selects the most optimal routes while forwarding the data to its next hop. The authors in [17] proposed an adaptive routing scheme for underwater delay/disruption tolerant networks. The scheme assumes that each node knows its 3D position through localization schemes. The priority of any data packet is calculated and assigned on the basis of varying properties of a node and data packet including node density, residual energy, age of packet,

and urgency of packet. The protocol achieves a good tradeoff between energy consumption, delay, and delivery ration.

**5.2. DDD.** E. Magistretti et al. [18] proposed a delay tolerant data dolphin (DDD) approach to apply in a resource-constrained underwater environment. For ensuring energy efficiency, mobile collectors, called Dolphins, are introduced. In order to limit the utilization of multihop channels, the underwater sensors are obligated to communicate within a single hop when a Dolphin is within transmission range. The rationale is to increase the overall network lifetime. Hence, the sensors can save power by limiting their transmission to only mobile Dolphins and avoiding multihop communication. Once data are delivered, the sensors' memory is flushed out and data are secured in the Dolphins' memory. Dolphins are prepared to transmit periodic beacons to inform the sensors about their arrival in the region. As soon as nodes determine the presence of a Dolphin, they turn on acoustic modems and upload stored events to the Dolphin. Later, when the Dolphin moves within the communication range of a base station, it uploads all gathered events. However, adjusting the optimal speed of the Dolphin and its trajectory is still an important issue.

**5.3. Azad et al. UDTN.** Autonomous coastal surveillance is one of several applications for underwater networks. Mostly, AUVs are deployed under the shores to monitor a given area. However, AUVs depend on a fixed trajectory. Even if the trajectory is adaptive, in case of critical data, they require time to reach a base station for data delivery. For coping with the issue, Azad et al. [19] proposed a potential communication between two AUVs that can minimize the delay in data delivery during coastal surveillance. Generally, the given area is quite large, so multiple AUVs are deployed and each AUV aims to reach the base station after completing its full-length trajectory, which causes a very long delay. Hence, the core idea is to forward the data between two or more AUVs after exchanging beacon, information, and response messages. The information message is sent as a response to the beacon and it includes the identities, routes, and movement speed of AUV(s). This information ensures the decision-making, that is, either transmitting data to the AUV(s) passing by or not.

**5.4. Sweeper.** One of the important factors affecting wireless communication is antenna capabilities. In [12], the authors proposed a scheme called Sweeper that considers a central node with an adaptive directional antenna array. This antenna array forms a beam in a specific direction and width, called downlink, which implies that the central node can communicate with the normal underwater sensor nodes within the specific region. The central node sends a wakeup or query message, which consists of a specific task for the normal nodes in the specific region through downlink. The receiving node(s) first calculate(s) depth level using a received signal strength indicator to estimate the region. After successful reception of the wakeup message, the normal nodes in the region send their sensed data towards the central node through the same direction of arrival, that is, uplink direction. In this manner, the central node sweeps the area to collect

data and keeps track of location information of the normal nodes in the region.

**5.5. Tier-Based Cross-Layer.** A tier-based cross-layer routing protocol for the underwater DTN has been proposed in [20]. The network is divided into tiers with a reference from the sink node. In one tier, both transmitter and receiver should be in each other's transmission range and the maximum range is defined as  $d_{\max}$ . The value of  $d_{\max}$  is determined to achieve a certain level of channel capacity, because the channel capacity is highly affected by the distance between transmitter and receiver. The cross-layer optimization is achieved by synchronizing the MAC and routing functionalities together to minimize collision that causes the delay. If a node in tier  $j$  wants to communicate with the sink, it sends its tier information in a request-to-send message and the node(s) in the upper tier,  $j1$ , reply with clear-to-send (CTS) message. After receiving multiple CTSs from the upper tier nodes, the source node sends intent-to-send with the node number that acts as a next-hop routing node. The selection of the next-hop node is done based on the minimum distance that requires small transmit power. As a result, the proposed protocol has minimum delay and high accuracy rate and consumes less energy.

**5.6. QDTR.** In [10], authors proposed Q-Learning-based routing protocol for underwater DTNs. Q-Learning is one of the reinforcement learning algorithms that is used to take optimal data-forwarding decisions. In the given protocol, underwater nodes forward single or multicopy packets at the beginning and spray packets among the neighbors of the destination at the end, which achieves better network performance and energy.

However, the protocol does not assume any node mobility pattern, as normally used in terrestrial DTNs, where future meeting events can be predicted with the aid of the mobility model. The proposed protocol is able to adapt to packet priority. When the deadline is approaching, nodes forward more copies of packets than usual, which is a tradeoff between higher energy and shorter delay.

**5.7. RBAR.** Redundancy Based Adaptive Routing (RBAR) [21] for underwater DTNs has been proposed allowing a node to hold packet as long as possible until it has to make another copy to satisfy its delay requirement. In order to achieve this goal, RBAR takes advantage from a binary tree based forwarding procedure, which helps packet replication process to be explicitly determined. The main contribution of RBAR is packet-forwarding process.

**5.8. PBDTP.** Zhang et al. intended to address channel throughput issue for underwater DTNs [22]. Due to inconsistent delay in acoustic channel, many traditional communication protocols are insufficient since they depend on accurate estimations of RTT, that is, Round Trip Time between two nodes. In order to deal with long delays, a Prediction-Based Delay Tolerant Protocol has been proposed, where prediction for lost data by any sensor node with in cluster is performed by cluster head. The core idea behind this protocol is based

TABLE 1: Summary of underwater DTN routing protocols.

Routing protocol	Communication initiation mechanism	Infrastructure assistance	Routing decision	Location information dependence	Number of copies	Performance indicator
PASR	Sender	No	Distributed	No	Single	Replication control
DDD	Receiver	No	Centralized	No	Single	Energy efficiency
UDTN	Receiver	No	Distributed	No	Single	End-to-end delay minimization
Sweeper	Receiver	Yes	Centralized	Yes	Single	UW nodes localization
Tier-based UDTN	Sender	Yes	Centralized	Yes	Single	Energy, delay, and packet drop constraints
QDTR	Sender	No	Distributed	No	Multiple	Energy efficiency and end-to-end delay
RBAR	Sender	Yes	Centralized	No	Multiple	Energy efficiency
PBDTP	Sender	Yes	Not Available	No	Multiple	Long delay tolerance

on two assumptions: (a) a sensor node most likely will send the same data with small time interval and (b) adjacent sensor nodes will also report almost identical sensed data. In case of lost data packet, the proposed algorithm predicts the missing value based on either previous values of specific sensor node or its neighbor's values. On the other hand, prediction is also being done by combination of both. In meanwhile, the lost data packets are compensated at cluster head nodes avoiding extra delays caused by sending retransmission requests and waiting for sensor to resend data.

Table 1 shows the summary of underwater DTN protocols in terms of the previously discussed classification parameters. It shows that PASR utilizes the offline greedy algorithm for mobility prediction. Based on mobility patterns archived in the learning phase, two different schemes have been tested for random mobility traces [16]. While selecting a next-hop, a sender has information about the most optimal route for a destination. This leads to low congestion and an overhead transmitting single copy of the data. Therefore, it belongs to the sender-initialized communication.

Controversially, DDD [18] uses Dolphins, mobile AUVs, as data collectors. Dolphins initiate communication and collect data from the one-hop UWN nodes while on the move. On successful reception of data by the Dolphin, each UWN node refreshes its buffer. The tier-based underwater DTN routing scheme [20] is completely based on the infrastructure nodes. We argue that the infrastructure-dependent schemes have to rely on relevant ideal assumptions for showing better performance as compared to those with no dependency on infrastructure.

As mentioned before, the anatomy of overall network lifetime in UWN is very important. For achieving this, Sweeper [12] provides a centralized localization scheme where a central node sweeps the entire network to estimate the position of the sensor nodes. Further contrast between the selected protocols is elaborated in Table 1. Next, we present

the current challenges and open issues in underwater DTN routing.

## 6. Challenges and Open Issues

Here, we further identify the remaining challenges and issues, providing a concise guide for researchers. As we discussed earlier, routing is a key challenge in underwater DTNs where random mobility and intermittent environment complicate the acoustic channels. It is important to select the next hop using time-varying information while considering the usage of link quality and buffer space as well as energy. However, there still exist the following challenges and open issues that need to be investigated.

- (i) The protocols using a multiple paths transmission strategy provide reliability and robustness. These protocols also have low end-to-end delays. However, robustness is further improved with the cost of retransmission. As repetitions limit the overall performance of a protocol and the complexity of maintaining more paths enlarges the processing overhead, efficient efforts are required to avoid retransmissions. More specifically, sending the same packet through multiple routes can be addressed in an intelligent manner.
- (ii) There always exists a tradeoff between delivery ratio, energy constraints, and end-to-end delay while applying different approaches in the underwater DTNs. Artificial intelligence-aware routing can be expected to minimize the tradeoff.
- (iii) Numerous routing protocols are based on historical information being used to select next-hop for communications. The problem is to determine the mechanism of information collection. The accuracy

in case of random movements or future mobility of nodes can make it more challenging.

- (iv) In most cases, researchers are emphasizing energy constraints while investigating routing issues. However, reallocating buffer space and maximum use of other resources can also be addressed in order to test algorithms in real-time experiments.
- (v) In addition, a smart router should be configured to support many routing algorithms at once using a fuzzy logic technique. It should be a great effort to enhance and provide multiple routing decisions in a specific network.

Although many routing algorithms exist in the literature for DTNs and underwater acoustic networks, the number providing real-time analysis is still very limited. Therefore, there is a need for developing algorithms that facilitate real-time analysis.

## 7. Conclusion

In this article, we first provided an overview of routing in underwater communication and then presented delay tolerance features being adapted in the UWN research paradigm. The main goal of this survey is to make researchers and members of the industry familiar with the recent technology convergence in the field of underwater sensor networks. For that purpose, we summarized state-of-the-art routing protocols considering no end-to-end connectivity. In addition, we also provided a list of open challenges and future directions followed by a comparative analysis of the selected underwater DTN protocols. We hope this article will further motivate research interest in underwater DTNs.

## Competing Interests

There are no competing interests regarding the publication of this paper.

## Acknowledgments

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2016R1D1A3B01015510).

## References

- [1] A. Darehshoorzadeh and A. Boukerche, "Underwater sensor networks: a new challenge for opportunistic routing protocols," *IEEE Communications Magazine*, vol. 53, no. 11, pp. 98–107, 2015.
- [2] D. Pompili and I. F. Akyildiz, "Overview of networking protocols for underwater wireless communications," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 97–102, 2009.
- [3] B. Li, Y. Xu, and W. Xu, "Multi-AUV localization for an underwater acoustic sensor network," in *Underwater Acoustics and Ocean Dynamics*, pp. 117–127, Springer, Singapore, 2016.
- [4] J. J. Kartha, A. Jabbar, A. Baburaj, and L. Jacob, "Maximum lifetime routing in underwater sensor networks using mobile sink for delaytolerant applications," in *Proceedings of the IEEE Region 10 Conference (TENCON '15)*, pp. 1–6, Macao, China, November 2015.
- [5] D. Park, K. Kwak, W. K. Chung, and J. Kim, "Development of underwater short-range sensor using electromagnetic wave attenuation," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 2, pp. 318–325, 2016.
- [6] A. Caiti, E. Ciaramella, G. Conte et al., "OptoCOMM: introducing a new optical underwater wireless communication modem," in *Proceedings of the IEEE 3rd Underwater Communications and Networking Conference (UComms '16)*, pp. 1–5, Lerici, Italy, August 2016.
- [7] R. Grasso, P. Braca, S. Fortunati, F. Gini, and M. S. Greco, "Dynamic underwater sensor network for sparse field estimation," in *Proceedings of the 3rd IEEE International Workshop on Compressed Sensing Theory and Its Applications to Radar, Sonar and Remote Sensing (CoSeRa '15)*, pp. 169–173, Genova, Italy, May 2015.
- [8] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: propagation models and statistical characterization," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 84–89, 2009.
- [9] K. Fall, "A delay-tolerant network architecture for challenged internet," in *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '03)*, pp. 27–34, ACM, Karlsruhe, Germany, 2003.
- [10] T. Hu and Y. Fei, "An adaptive and energy-efficient routing protocol based on machine learning for underwater delay tolerant networks," in *Proceedings of the 18th Annual IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS '10)*, pp. 381–384, August 2010.
- [11] G. Wang, B. Wang, and Y. Gao, "Dynamic spray and wait routing algorithm with quality of node in delay tolerant network" in *Proceedings of the IEEE International Conference on Communications and Mobile Computing (CMC '10)*, vol. 3, pp. 452–456, April 2010.
- [12] K. T. Dharan, C. Srimathi, and S.-H. Park, "A sweeper scheme for localization and mobility prediction in underwater acoustic sensor networks," in *Proceedings of the OCEANS'10 IEEE Sydney (OCEANSSYD '10)*, Sydney, Australia, May 2010.
- [13] H.-H. Cho, C.-Y. Chen, T. K. Shih, and H.-C. Chao, "Survey on underwater delay/disruption tolerant wireless sensor network routing," *IET Wireless Sensor Systems*, vol. 4, no. 3, pp. 112–121, 2014.
- [14] S. H. Ahmed, A. Wahid, and D. Kim, "EENC - Energy efficient nested clustering in UASN," in *Proceedings of the 29th Annual ACM Symposium on Applied Computing (SAC '14)*, pp. 706–710, March 2014.
- [15] A. Caruso, F. Paparella, L. F. M. Vieira, M. Erol, and M. Gerla, "The meandering current mobility model and its impact on underwater mobile sensor networks," in *Proceedings of the 27th IEEE Communications Society Conference on Computer Communications (INFOCOM '08)*, pp. 771–779, Phoenix, Ariz, USA, April 2008.
- [16] Z. Guo, B. Wang, and J.-H. Cui, "Generic prediction assisted single-copy routing in underwater delay tolerant sensor networks," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1136–1149, 2013.
- [17] Z. Guo, G. Colombi, B. Wang, J.-H. Cui, D. Maggiorini, and G. P. Rossi, "Adaptive routing in underwater delay/disruption

- tolerant sensor networks,” in *Proceedings of the 5th Annual Conference on Wireless on Demand Network Systems and Services (WONS '08)*, pp. 31–39, Garmisch-Partenkirchen, Germany, 2008.
- [18] E. Magistretti, J. Kong, U. Lee, M. Gerla, P. Bellavista, and A. Corradi, “A mobile delay-tolerant approach to longterm energy-efficient underwater sensor networking,” in *Proceedings of the Wireless Communications and Networking Conference (WCNC '07)*, pp. 2866–2871, March 2007.
- [19] S. Azad, P. Casari, and M. Zorzi, “Coastal patrol and surveillance networks using AUVs and delay-tolerant networking,” in *Proceedings of the IEEE OCEANS*, pp. 1–8, Yeosu, South Korea, May 2012.
- [20] L. Kuo and T. Melodia, “Tier-based underwater acoustic routing for applications with reliability and delay constraints,” in *Proceedings of the IEEE International Workshop on Wireless Mesh and Ad Hoc Networks (WiMAN '11)*, pp. 1–6, Maui, Hawaii, USA, 2011.
- [21] Z. Guo, Z. Peng, B. Wang, J.-H. Cui, and J. Wu, “Adaptive routing in underwater delay tolerant sensor networks,” in *Proceedings of the 6th International ICST Conference on Communications and Networking in China (CHINACOM '11)*, pp. 1044–1051, IEEE, Harbin, China, August 2011.
- [22] Z. Zhang, S.-L. Lin, and K.-T. Sung, “A prediction-based delay-tolerant protocol for underwater wireless sensor networks,” in *Proceedings of the International Conference on Wireless Communications and Signal Processing (WCSP '10)*, pp. 1–6, IEEE, Suzhou, China, October 2010.