

Algorithmic Foundations of IoT

Lead Guest Editor: Wei Cheng

Guest Editors: Tao Chen, Jun Liu, Sergio A. S. Monroy, and Zaobo He



Algorithmic Foundations of IoT

Wireless Communications and Mobile Computing

Algorithmic Foundations of IoT

Lead Guest Editor: Wei Cheng


Guest Editors: Tao Chen, Jun Liu, Sergio A. S.
Monroy, and Zaobo He




Copyright © 2019 Hindawi Limited. All rights reserved.

This is a special issue published in “Wireless Communications and Mobile Computing.” 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.

Chief Editor

Zhipeng Cai , USA

Associate Editors

Ke Guan , China
Jaime Lloret , Spain
Maode Ma , Singapore

Academic Editors

Muhammad Inam Abbasi, Malaysia
Ghufran Ahmed , Pakistan
Hamza Mohammed Ridha Al-Khafaji ,
Iraq
Abdullah Alamoodi , Malaysia
Marica Amadeo, Italy
Sandhya Aneja, USA
Mohd Dilshad Ansari, India
Eva Antonino-Daviu , Spain
Mehmet Emin Aydin, United Kingdom
Parameshchhari B. D. , India
Kalapaveen Bagadi , India
Ashish Bagwari , India
Dr. Abdul Basit , Pakistan
Alessandro Bazzi , Italy
Zdenek Becvar , Czech Republic
Nabil Benamar , Morocco
Olivier Berder, France
Petros S. Bithas, Greece
Dario Bruneo , Italy
Jun Cai, Canada
Xuesong Cai, Denmark
Gerardo Canfora , Italy
Rolando Carrasco, United Kingdom
Vicente Casares-Giner , Spain
Brijesh Chaurasia, India
Lin Chen , France
Xianfu Chen , Finland
Hui Cheng , United Kingdom
Hsin-Hung Cho, Taiwan
Ernestina Cianca , Italy
Marta Cimitile , Italy
Riccardo Colella , Italy
Mario Collotta , Italy
Massimo Condoluci , Sweden
Antonino Crivello , Italy
Antonio De Domenico , France
Floriano De Rango , Italy

Antonio De la Oliva , Spain
Margot Deruyck, Belgium
Liang Dong , USA
Praveen Kumar Donta, Austria
Zhuojun Duan, USA
Mohammed El-Hajjar , United Kingdom
Oscar Esparza , Spain
Maria Fazio , Italy
Mauro Femminella , Italy
Manuel Fernandez-Veiga , Spain
Gianluigi Ferrari , Italy
Luca Foschini , Italy
Alexandros G. Fragkiadakis , Greece
Ivan Ganchev , Bulgaria
Óscar García, Spain
Manuel García Sánchez , Spain
L. J. García Villalba , Spain
Miguel Garcia-Pineda , Spain
Piedad Garrido , Spain
Michele Girolami, Italy
Mariusz Glabowski , Poland
Carles Gomez , Spain
Antonio Guerrieri , Italy
Barbara Guidi , Italy
Rami Hamdi, Qatar
Tao Han, USA
Sherief Hashima , Egypt
Mahmoud Hassaballah , Egypt
Yejun He , China
Yixin He, China
Andrej Hrovat , Slovenia
Chunqiang Hu , China
Xuexian Hu , China
Zhenghua Huang , China
Xiaohong Jiang , Japan
Vicente Julian , Spain
Rajesh Kaluri , India
Dimitrios Katsaros, Greece
Muhammad Asghar Khan, Pakistan
Rahim Khan , Pakistan
Ahmed Khattab, Egypt
Hasan Ali Khattak, Pakistan
Mario Kolberg , United Kingdom
Meet Kumari, India
Wen-Cheng Lai , Taiwan

Jose M. Lanza-Gutierrez, Spain
Pavlos I. Lazaridis , United Kingdom
Kim-Hung Le , Vietnam
Tuan Anh Le , United Kingdom
Xianfu Lei, China
Jianfeng Li , China
Xiangxue Li , China
Yaguang Lin , China
Zhi Lin , China
Liu Liu , China
Mingqian Liu , China
Zhi Liu, Japan
Miguel López-Benítez , United Kingdom
Chuanwen Luo , China
Lu Lv, China
Basem M. ElHalawany , Egypt
Imadeldin Mahgoub , USA
Rajesh Manoharan , India
Davide Mattera , Italy
Michael McGuire , Canada
Weizhi Meng , Denmark
Klaus Moessner , United Kingdom
Simone Morosi , Italy
Amrit Mukherjee, Czech Republic
Shahid Mumtaz , Portugal
Giovanni Nardini , Italy
Tuan M. Nguyen , Vietnam
Petros Nicolaitidis , Greece
Rajendran Parthiban , Malaysia
Giovanni Pau , Italy
Matteo Petracca , Italy
Marco Picone , Italy
Daniele Pinchera , Italy
Giuseppe Piro , Italy
Javier Prieto , Spain
Umair Rafique, Finland
Maheswar Rajagopal , India
Sujan Rajbhandari , United Kingdom
Rajib Rana, Australia
Luca Reggiani , Italy
Daniel G. Reina , Spain
Bo Rong , Canada
Mangal Sain , Republic of Korea
Praneet Saurabh , India

Hans Schotten, Germany
Patrick Seeling , USA
Muhammad Shafiq , China
Zaffar Ahmed Shaikh , Pakistan
Vishal Sharma , United Kingdom
Kaize Shi , Australia
Chakchai So-In, Thailand
Enrique Stevens-Navarro , Mexico
Sangeetha Subbaraj , India
Tien-Wen Sung, Taiwan
Suhua Tang , Japan
Pan Tang , China
Pierre-Martin Tardif , Canada
Sreenath Reddy Thummaluru, India
Tran Trung Duy , Vietnam
Fan-Hsun Tseng, Taiwan
S Velliangiri , India
Quoc-Tuan Vien , United Kingdom
Enrico M. Vitucci , Italy
Shaohua Wan , China
Dawei Wang, China
Huaqun Wang , China
Pengfei Wang , China
Dapeng Wu , China
Huaming Wu , China
Ding Xu , China
YAN YAO , China
Jie Yang, USA
Long Yang , China
Qiang Ye , Canada
Changyan Yi , China
Ya-Ju Yu , Taiwan
Marat V. Yuldashev , Finland
Sherali Zeadally, USA
Hong-Hai Zhang, USA
Jiliang Zhang, China
Lei Zhang, Spain
Wence Zhang , China
Yushu Zhang, China
Kechen Zheng, China
Fuhui Zhou , USA
Meiling Zhu, United Kingdom
Zhengyu Zhu , China

Contents

Provably Secure Identity-Based Encryption and Signature over Cyclotomic Fields

Yang Wang, Mingqiang Wang , Jingdan Zou , Jin Xu, and Jing Wang


Research Article (13 pages), Article ID 1742386, Volume 2019 (2019)

Two Secure Privacy-Preserving Data Aggregation Schemes for IoT

Yuwen Pu, Jin Luo, Chunqiang Hu , Jiguo Yu , Ruifeng Zhao, Hongyu Huang, and Tao Xiang



Research Article (11 pages), Article ID 3985232, Volume 2019 (2019)

Distributed Link Scheduling Algorithm Based on Successive Interference Cancellation in MIMO Wireless Networks

Junhua Wu, Dandan Lin, Guangshun Li , Yuncui Liu, and Yanmin Yin



Research Article (12 pages), Article ID 9083282, Volume 2019 (2019)

A Lightweight Fine-Grained Searchable Encryption Scheme in Fog-Based Healthcare IoT Networks

Hui Li  and Tao Jing 



Research Article (15 pages), Article ID 1019767, Volume 2019 (2019)

Replication-Based Data Dissemination in Connected Internet of Vehicles

Xiying Fan , Chuanhe Huang , Junyu Zhu, and Bin Fu


Research Article (16 pages), Article ID 2150524, Volume 2019 (2019)

A Novel Task Allocation Algorithm in Mobile Crowdsensing with Spatial Privacy Preservation

Wenyi Tang, Qi Jin, Xu Zheng, Guangchun Luo , Guiduo Duan, and Aiguo Chen 


Research Article (13 pages), Article ID 3154917, Volume 2019 (2019)

Energy-Efficient Broadcast Scheduling Algorithm in Duty-Cycled Multihop Wireless Networks

Quan Chen, Tao Wang, Lianglun Cheng, Yongchao Tao , and Hong Gao

Research Article (14 pages), Article ID 5064109, Volume 2019 (2019)

A Center-Based Secure and Stable Clustering Algorithm for VANETs on Highways

Xiaolu Cheng and Baohua Huang 

Research Article (10 pages), Article ID 8415234, Volume 2019 (2019)

Research Article

Provably Secure Identity-Based Encryption and Signature over Cyclotomic Fields

Yang Wang,¹ Mingqiang Wang ,¹ Jingdan Zou ,¹ Jin Xu,¹ and Jing Wang²

¹School of Mathematics, Shandong University, Jinan Shandong 250100, China

²Shandong Branch of China Mobile Online Service Co. Ltd., Jinan Shandong 250100, China

Correspondence should be addressed to Mingqiang Wang; wangmingqiang@sdu.edu.cn

Received 29 March 2019; Revised 29 May 2019; Accepted 8 July 2019; Published 17 October 2019

Guest Editor: Zaobo He

Copyright © 2019 Yang Wang 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.

Identity-based cryptography is a type of public key cryptography with simple key management procedures. To our knowledge, till now, the existing identity-based cryptography based on NTRU is all over power-of-2 cyclotomic rings. Whether there is provably secure identity-based cryptography over more general fields is still open. In this paper, with the help of the results of collision resistance preimage sampleable functions (CRPSF) over cyclotomic fields, we give concrete constructions of provably secure identity-based encryption schemes (IBE) and identity-based signature schemes (IBS) based on NTRU over any cyclotomic field. Our IBE schemes are provably secure under adaptive chosen-plaintext and adaptive chosen-identity attacks, meanwhile, our IBS schemes are existentially unforgeable against adaptively chosen message and adaptively chosen identity attacks for any probabilistic polynomial time (PPT) adversary in the random oracle model. The securities of both schemes are based on the worst-case approximate shortest independent vectors problem (SIVP_γ) over corresponding ideal lattices. The secret key size of our IBE (IBS) scheme is short—only one (two) ring element(s). The ciphertext (signature) is also short—only two (three) ring elements. Meanwhile, as the case of NTRUEncrypt, our IBE scheme could encrypt n bits in each encryption process. These properties may make our schemes have more advantages for some IoT applications over postquantum world in theory.

1. Introduction

Nowadays, Internet of things (IoT) plays an extremely important role by comprising millions of smart and connected devices to offer benefits in a wide range of situations, for example, smart cities, smart grids, smart traffic, and smart buildings. The corresponding techniques have been unprecedentedly developed and adopted due to the quick evolution of smart devices and the continuous investment of leading communities. In a smart IoT system, data collected by mote devices will be transferred to gateway/cloud; the cloud will perform data analysis and send the results to the particular management system which takes suitable action. How to protect this complete network against malicious events, as well as the privacy and authenticity of data, is one of the toughest challenges for the deploying IoT technology. Several considerations and solutions are discussed in [1–4]. Due to the constrained resources (i.e., the size of memory, CPU speed, and network bandwidth), we could not directly

use the traditional public key system, since the key management is complicated and the computations and storages may consume large amount of resources.

Identity-based cryptography is a type of public key cryptography in which the public key of a user is some unique information about the identity of the user (e.g., a user's e-mail address and the MAC address of devices). This means that a sender who has access to the public parameters of the system can encrypt a message (verify a signature) by using the receiver's (signer's) identity as a public key. The receiver (signer) obtains its decryption (signing) key from a central authority, which needs to be trusted as it generates secret keys for every user. Such cryptographic primitives significantly simplify the key management procedures of certificated-based public key infrastructures.

IBE and IBS were proposed by Shamir [5]; from then on, a large number of papers have been published in this area, including IBE [6–12], IBS [13–17], and identity-based signcryption (sign-then-encrypt a message) schemes

[13, 18, 19]. Till now, the fully practical identity-based cryptographic primitives are based on bilinear pairings. With the rapid development of quantum computation, in a not-so-distant future, quantum computers are expected to break such systems, and it is urgent to design quantum-immune IBE and IBS schemes. Cryptographic primitives based on hard lattice problems are good candidates, and many such identity-based schemes were designed [6, 9, 10, 16]. However, the efficiency of these schemes is not very satisfactory, especially in the IoT applications. As we all know, cryptographic primitives based on NTRU usually have high efficiency [20] and are good candidates of lightweight cryptographic systems in the postquantum world. Therefore, IBE and IBS schemes based on NTRU may enjoy the advantages of high efficiency and quantum-immune at the same time.

To the best of our knowledge, the existing IBE [21] and IBS [17] based on NTRU are all over power-of-2 cyclotomic rings, in which NTT algorithm can be implemented and calculations can be done very fast. However, there are too many subfields in the corresponding cyclotomic fields, making these settings more sensitive to subfield attacks [22, 23, 24]. So, seeking constructions of IBE and IBS over more general fields is a meaningful work. Meanwhile, strictly speaking, both of the schemes [17, 21] lack a security proof in the following two senses: (1) The PPT key generation algorithm [21] is heuristic and the CPA security of the schemes is guaranteed by a key-encapsulation mechanism designed in the process of encryption and is measured by the Kullback–Leibler “distance”—not statistical distance. Then, security is estimated in the aspect of attacks. So, the magnitude of module q is small and the schemes are practical. (2) Parameter settings of IBS [17] were referred to [25]; while the main lemma for proving the PPT trapdoor generation algorithm of CRPSF in [25] had some deficiencies, making the parameter choices in [17] could not achieve the desired result.

1.1. Our Contributions and Technique Overview. Motivated by the above reasons, we construct provably secure IBE and IBS schemes over any cyclotomic field.

Compared with [21], our IBE scheme is strictly provably secure under adaptive chosen-plaintext and adaptive chosen-identity attacks. So, at a high level, our result implies that we can heuristically design IBE scheme by using similar parameters as [21] in any cyclotomic field. Since we use the modified algorithms of CRPSF proposed in [26], our IBS scheme is existentially unforgeable against adaptively chosen message and adaptively chosen identity attacks in theory. Though the efficiency of our IBE and IBS schemes may be not satisfactory when we set parameters to achieve the provably security, our results give a high-level implication that we can heuristically design IBE and IBS over any cyclotomic field with small parameters (for example, settings of the classical NTRU-based cryptography [20]) and construct a lightweight cryptosystem, which can be used in some IoT applications.

Next, we give a brief review of constructions.

The construction of our IBE scheme is inspired by [21] and followed the route of [10]. The setup algorithm uses the key generation algorithm of CRPSF constructed in [26] to generate some public parameters \mathbf{PP} , including a cyclotomic field K and an element $h \in R_q^\times$. Here, $R = \mathcal{O}_K$ is the ring of integers of K and R_q^\times is the set of invertible elements of $R_q = R/qR$. Meanwhile, the key generation algorithm of CRPSF also outputs a short trapdoor basis of the NTRU lattice $\Lambda_h^q = \{(x, y) \in R^2 : y = hx \bmod qR\}$. The secret key of an identity (we map an identity to R_q by using a random oracle $H : \{0, 1\}^* \mapsto R_q$) is the element in Λ_h^q outputted by the SamplePre algorithm of CRPSF by using the trapdoor basis. The encryption and decryption follow the idea of [10]. We embed the message in a Ring-LWE instance in the encryption process and the outputted ciphertext consists of two Ring-LWE instances (only the b -component) (u, v) with the “implied” relation that $v - u \cdot \mathbf{sk}$ is short. Then, the decryption process only need to remove the errors by rounding ($\lfloor \cdot \rfloor$). Security (indistinguishability) is based on the hardness of corresponding decision Ring-LWE problems, and we do not need to use the key-encapsulation mechanism in the encryption process.

The construction of IBS follows the route of [17], which is a combination of techniques shown in [10, 27]. We also use the key generation algorithm of CRPSF to generate \mathbf{Msk} . The secret key (σ_1, σ_2) of an identity \mathbf{id} is produced by the SamplePre algorithm of CRPSF, satisfying $h\sigma_1 - \sigma_2 = H(\mathbf{id})$. The signing and verification follow the idea of [27] by using a rejection sampling algorithm. The signature of a message μ contains a triple (z_1, z_2, u) with $y_i \leftarrow D_{R, s}$, $z_i = y_i + \sigma_i \cdot u$, and $u = H'(hy_1 - y_2 \bmod qR, \mu)$. The rejection sampling algorithm could make it seem that z_i is independent of y_i , in particular, $z_i \leftarrow D_{R, s}$. Then, to verify a signature, one only needs to make sure that z_i is short and $u = H'(hz_1 - z_2 - H(\mathbf{id}) \cdot u \bmod qR, \mu)$. Unforgeability of our scheme can be reduced to the corresponding Ring-SIS problems.

Finally, we remark that techniques used in [28] are also vital to bound the decryption error of our IBE scheme. Though we design our IBE schemes in R^\vee , the dual ideal of R , we can convert it to work in an integral ideal of R or we can directly design the IBE scheme in R by using the hardness result shown in [29] (with larger parameter γ and q). Also, we can discuss the practicability under the Kullback–Leibler “distance” by using the same method as in [21]. Meanwhile, our construction provides an important support for designing IBE and IBS over general cyclotomic rings with relative small parameters (with no provably secure guarantee, but the key generation algorithm is PPT by our results) and analyzing the security from the view of attacks. How to reduce the magnitudes of parameters of provably secure identity-based cryptographic primitives and improve the efficiency of these schemes are important and meaningful open problems.

1.2. Organization. In Section 2, we will introduce some notations and basic results we need in our discussion. In Section 3, we shall discuss the IBE schemes, including the basic definitions, security models, constructions, and

security analysis. Discussions of IBS schemes are put in Section 4.

2. Preliminaries

In this section, we introduce some background results and notations.

2.1. Notations. We use $[n]$ to denote the set $\{1, 2, \dots, n\}$. $\|\cdot\|$ represents the l_2 norm corresponding to the canonical embedding. For two random variables X and Y , $\Delta(X, Y)$ stands for their statistic distance. When we write $X \leftarrow \xi$, we mean that the random variable X obeys to a distribution ξ . If S is a finite set, then $|S|$ is its cardinality and $U(S)$ is the uniform distribution over S . Symbols \mathbb{Z}^+ and \mathbb{R}^+ stand for the sets of positive integers and positive reals. Symbol $\log x$ represents $\log_2 x$ for $x \in \mathbb{R}^+$. Functions $\varphi(n)$ and $\mu(n)$ stand for the Euler function and the Möbius function.

2.2. Cyclotomic Fields, Space H , and Ideal Lattices. Throughout this paper, we only consider cyclotomic fields. For a cyclotomic field $K = \mathbb{Q}(\zeta)$ with $\zeta = \zeta_l$ the primitive l -th root of unity, its minimal polynomial is $\Phi_l(x) = \prod_{i|l} (x^i - 1)^{\mu(l/i)} \in \mathbb{Z}[x]$ with degree $n = \varphi(l)$. As usual, we set $R = \mathcal{O}_K = \mathbb{Z}[\zeta]$, which is the ring of integers of K . Then, $[K : \mathbb{Q}] = n = 2r$, $K \cong \mathbb{Q}[x]/\Phi_l(x)$ and $R \cong \mathbb{Z}[x]/\Phi_l(x)$. K is Galois over \mathbb{Q} . We set $\text{Gal}(K/\mathbb{Q}) = \{\sigma_i : i = 1, \dots, n\}$ and use the canonical embedding σ on K , which maps $x \in K$ to a space $\{\sigma_i(x)\}_i \in H := \{(x_1, \dots, x_n) \in \mathbb{C}^n : x_{n+1-i} = \overline{x_i}, \forall i \in [r]\}$ via embeddings in $\text{Gal}(K/\mathbb{Q})$. H is isomorphic to \mathbb{R}^n as an inner product space via the orthonormal basis $h_{i \in [n]}$ defined as follows: for $1 \leq j \leq r$,

$$\begin{cases} h_j = \frac{1}{\sqrt{2}}(e_j + e_{n+1-j}), \\ h_{n+1-j} = \frac{i}{\sqrt{2}}(e_j - e_{n+1-j}), \end{cases} \quad (1)$$

where $e_j \in \mathbb{C}^n$ is the vector with 1 in its j -th coordinate and 0 elsewhere and i is the imaginary number such that $i^2 = -1$. For any element $x \in K$, we can define its norm by $\|x\| := \|\sigma(x)\|$ and its infinity norm by $\|x\|_\infty := \max_{i \in [n]} |\sigma_i(x)|$.

We define a lattice as a discrete additive subgroup of H . The dual lattice of $\Lambda \subseteq H$ is defined as $\Lambda^\vee = \{y \in H : \forall x \in \Lambda, \langle x, \overline{y} \rangle = \sum_{i=1}^n x_i \cdot y_i \in \mathbb{Z}\}$. One can check that this definition is actually the complex conjugate of the dual lattice as usually defined in \mathbb{C}^n . All of the properties of the dual lattice that we use also hold for the conjugate dual. Any fractional ideal I of K is a free \mathbb{Z} module of rank n . So, $\sigma(I)$ is a lattice of H , and we call $\sigma(I)$ an ideal lattice and identify I with this lattice and associate with I all the usual lattice quantities. Meanwhile, its dual is defined as $I^\vee = \{a \in K : \text{Tr}(a \cdot I) \subseteq \mathbb{Z}\}$. Then, it is easy to verify that $(I^\vee)^\vee = I$, I^\vee is a fractional ideal, and I^\vee embeds under σ as the dual lattice of I as defined above.

2.3. Gaussian Distributions, Ring-SIS Problems, and Ring-LWE Problems. The Gaussian distribution is defined as usual. For any $s > 0$, $c \in H$, which is taken to be $s = 1$ or $c = 0$ when omitted, define the Gaussian function $\rho_{s,c} : H \rightarrow (0, 1]$ as $\rho_{s,c}(x) = e^{-\pi(\|x-c\|^2/s^2)}$. By normalizing this function, we obtain the continuous Gaussian probability distribution $D_{s,c}$ of parameter s , whose density function is given by $s^{-n} \cdot \rho_{s,c}(x)$. For a real vector $r = (r_1, \dots, r_n) \in (\mathbb{R}^+)^n$, we define the elliptical Gaussian distributions in the basis $\{h_i\}_{i \in [n]}$ as follows: a sample from D_r is given by $\sum_{i \in [n]} x_i h_i$, where x_i is chosen independently from the Gaussian distribution D_{r_i} over \mathbb{R} . Note that if we define a map $\varphi : H \rightarrow \mathbb{R}^n$ by $\varphi(\sum_{i \in [n]} x_i h_i) = (x_1, \dots, x_n)$, then D_r is also a (elliptical) Gaussian distribution over \mathbb{R}^n .

For a lattice $\Lambda \subseteq H$, $\sigma > 0$ and $c \in H$, we define the lattice Gaussian distribution of support Λ , deviation σ , and center c by $D_{\Lambda, \sigma, c}(x) = (\rho_{\sigma, c}(x) / \rho_{\sigma, c}(\Lambda))$ for any $x \in \Lambda$. For $\delta > 0$, we define the smoothing parameter $\eta_\delta(\Lambda)$ as the smallest $\sigma > 0$ such that $\rho_{1/\sigma}(\Lambda^\vee \setminus 0) \leq \delta$. The following theorem comes from [10, 30]. Here we use \tilde{B} to represent the Gram-Schmidt orthogonalization of B and regard the columns of B as a set of vectors. For $B = (b_1, \dots, b_n)$, define $\|B\| = \max_i \|b_i\|$.

Theorem 1. *There is a probabilistic polynomial time algorithm that, given a basis B of an n -dimensional lattice $\Lambda = \mathcal{L}(B)$, a standard deviation $\sigma \geq \|B\| \cdot \omega(\sqrt{\log n})$, and a $c \in H$, outputs a sample whose distribution is statistically close to $D_{\Lambda, \sigma, c}$.*

We will use following lemmata from [10, 31].

Lemma 1. *For any full-rank lattice Λ and positive real $\varepsilon > 0$, we have $\eta_\varepsilon(\Lambda) \leq \sqrt{\ln(2n(1 + (1/\varepsilon)))}/\pi \cdot \lambda_n(\Lambda)$.*

Lemma 2. *For any full-rank lattice Λ , $c \in H$, $\varepsilon \in (0, 1)$ and $\sigma \geq \eta_\varepsilon(\Lambda)$, we have $\Pr_{b \leftarrow D_{\Lambda, \sigma, c}}[\|b - c\| \geq \sigma\sqrt{n}] \leq (1 + \varepsilon/1 - \varepsilon) \cdot 2^{-n}$.*

Lemma 3. *For any full-rank lattice $\Lambda \subseteq H$, $c \in H$, $\delta \in (0, 1)$, $\sigma \geq 2\eta_\delta(\Lambda)$ and $b \in \Lambda$, we have $D_{\Lambda, \sigma, c}(b) \leq (1 + \delta/1 - \delta) \cdot 2^{-n}$.*

The following useful rejection sampling theorem comes from [27]. We state an adapted version, corresponding to the canonical embedding and space H . Its proof is essentially the same as that in [27], so we put it in Appendix with a remark that the constant M can be effectively calculated in practice.

Theorem 2. *Let $\Lambda \subseteq H$ be an arbitrary lattice, $V \subseteq H$ be a set in which all elements have norms less than T , σ be some elements in \mathbb{R} such that $\sigma = \omega(T \cdot \sqrt{\log n})$, and $h : V \rightarrow [0, 1]$ be a probability distribution. Then, there exists an absolute constant M such that the distribution of the output of the following algorithm \mathcal{A} :*

- (1) $v \leftarrow h$
- (2) $z \leftarrow D_{\Lambda, \sigma, v}$
- (3) Output (z, v) with probability $\min(D_{\Lambda, \sigma}(z)/M \cdot D_{\Lambda, \sigma, v}(z), 1)$

is within statistical distance $2^{-\omega(\log n)}/M$ of the distribution of the output of the following algorithm \mathcal{F} :

- (1) $v \leftarrow h$
- (2) $z \leftarrow D_{\Lambda, \sigma}$
- (3) Output (z, v) with probability $1/M$.

Moreover, the probability p that \mathcal{A} outputs something satisfies $(1 - 2^{-\omega(\log n)})/M \leq p \leq (1/M)$.

The hard lattice problems we use are Ring-SIS and Ring-LWE problems. For an element $z = (z_1, \dots, z_m) \in R^m$, let us define $\|z\| := (\sum_{i=1}^m \|z_i\|^2)^{1/2}$. We first introduce the Ring-SIS problem. The definition is as follows.

Definition 1. Let R be the ring of integers of K , q and m be positive integers, and β be a real number. The small integer solution problem over R (R -SIS $_{q, m, \beta}$) is given $a_1, \dots, a_m \in R_q$ chosen independently from $U(R_q)$, find $z = (z_1, \dots, z_m) \in R^m$ such that $\sum_{i=1}^m a_i z_i = 0 \pmod{qR}$ and $0 < \|z\| \leq \beta$.

For appropriate parameters, the following theorem comes from [32], which shows that the Ring-SIS problem is hard.

Theorem 3. For $\varepsilon \in (0, 1)$, there is a PPT reduction from solving Ideal-SIVP $_{\gamma, \sqrt{\ln(2n(1+1/\varepsilon))/\pi}}$ with high probability in polynomial time in the worst case to solving R -SIS $_{q, m, \beta}$ with nonnegligible probability in polynomial time, for any m, q, β, γ such that $\gamma \geq \beta \sqrt{n} \cdot \omega(\sqrt{\log n})$, $q \geq \beta \sqrt{n} \cdot \omega(\log n)$, and $m, \beta, \log q \leq \text{poly}(n)$.

The Ring-LWE problem is defined as follows. Let $\mathbb{T} = H/R^V$.

Definition 2. For $s \in R_q^V$ and an error distribution ψ over H , the Ring-LWE distribution $A_{s, \psi}^V$ over $R_q \times \mathbb{T}$ is sampled by independently choosing a uniformly random $a \leftarrow U(R_q)$ and an error term $e \leftarrow \psi$ and outputting $(a, b = (a \cdot s/q) + e \pmod{R^V})$.

Definition 3. Let Ψ be a family of distributions over H . The average-case Ring-LWE decision problem, denoted R -DLWE $_{q, \Psi}^V$, is to distinguish (with nonnegligible advantage) between independent samples from $A_{s, \psi}^V$ for a random choice of $(s, \psi) \leftarrow U(R_q^V) \times \Psi$ and the same number of uniformly random and independent samples from $R_q \times \mathbb{T}$.

In [33], a reduction from Ideal-SIVP $_{\gamma}$ to decision Ring-LWE problem over any algebraic number field is given.

Theorem 4. Let K be an algebraic number field and $R = \mathcal{O}_K$, $[K : \mathbb{Q}] = n$. Assume $\alpha \in (0, 1)$ such that $\alpha \leq \sqrt{\log n/n}$, and let $q \geq 2$ be an integer such that $\alpha q \geq \omega(1)$. Then there is a polynomial time quantum reduction from Ideal-SIVP $_{\gamma}$ (in the worst case) to R -DLWE $_{q, D_{\xi}}^V$, where $\xi = \alpha(nk/\log(nk))^{1/4}$ with k the number of samples to be used and $\gamma = \omega(\sqrt{n} \cdot \log n/\alpha)$.

We can modify the sample (a, b) of Ring-LWE distribution to $R_q \times R_q^V$ as in [28]. We scale the b component by a factor of q , so that it is an element in $H/(qR^V)$. The

corresponding error distribution is $D_{q\xi}$ with $\xi = \alpha \cdot (nk/\log(nk))^{1/4}$ and k the number of samples. Then, we discretize the error, by taking $e \leftarrow \lfloor D_{q\xi} \rfloor_{R^V}$. The decision version of Ring-LWE becomes to distinguish between the modified distribution of $A_{s, \lfloor D_{q\xi} \rfloor_{R^V}}^V$ and the uniform samples from $R_q \times R_q^V$. Notice that by using the same method proposed in [34], we can change the secret s to obey the error distributions, i.e., $s \leftarrow \lfloor D_{q\xi} \rfloor_{R^V}$. We will use the symbol R -DLWE $_{q, \lfloor D_{q\xi} \rfloor_{R^V}}$ to denote this problem. Meanwhile, note that, if we constrain $a \leftarrow U(T)$ for some $T \subseteq R_q$, where $|T| = c \cdot |R_q|$ and $c \neq \text{negl}(n)$, the hardness of the corresponding problem does not decrease. We will use the symbol R -DLWE $_{q, \lfloor D_{q\xi} \rfloor_{R^V}}^{\times}$ to denote this problem. For more details, one can refer to [28, 34].

2.4. Key Generation Algorithm and Regularity Result. In this subsection, we shall introduce some useful algorithms and results we need. The following algorithm plays a key role in our constructions of IBE and IBS. It is a modified version of key generation algorithm of traditional NTRU signatures. For simplicity, we denote it by N -KeyGen.

The following theorem comes from [26] (Algorithm 1). Note that in the case of cyclotomic fields, it was shown in [26] that the value of Dedekind zeta function at 2 (i.e. $\zeta_K(2)$) has a relatively small absolute upper bound.

Theorem 5. Let $K = \mathbb{Q}(\zeta_l)$ be a cyclotomic field, $R = \mathcal{O}_K$, $n = \varphi(l)$, $q \geq 64n\zeta_K(2)$ be a prime such that $q \nmid \Delta_K$ and the prime ideal decomposition of qR in R is $qR = \mathcal{B}_1 \cdot \dots \cdot \mathcal{B}_g$ such that $\ell \cdot g = n$, $\varepsilon > 0$ be an arbitrary positive number. Assume that $\sigma \geq \max\{8n^{3.6} \ln n, \omega(n \ln^{0.5} n) \cdot q^{1/g}, \omega(n^{0.25} q^{0.5} l^{-0.25})\}$. Then, the key generation algorithm proposed in this section terminates in polynomial time, and the output

matrix $\begin{bmatrix} f & g \\ F & G \end{bmatrix}$ is an R basis of Λ_h^q for $h = gf^{-1} \pmod{qR}$.

Meanwhile, if $\sigma \geq n^{3/2} \sqrt{\ln(8nq)} \cdot q^{(1/2)+(1+(\ell/2))\varepsilon}$, the distribution of h is rejected with probability $c < 1$ for some absolute constant c from a distribution whose statistical distance from $U(R_q^{\times})$ is $\leq (2^{8n}/q^{\lfloor \sigma n \rfloor})$.

Based on the N -KeyGen algorithm, Wang and Wang [26] gave a detailed construction of CRPSF, which was first proposed in [10], over any cyclotomic field. The preimage sampling algorithm of CRPSF is useful for us to design our IBE and IBS. We also use NTRUCRPSF (n, q, σ, s) to represent the CRPSF and only describe the results we need. For more details, one can refer to [26]. The construction of CRPSF is as follows:

- (1) TrapGen $(1^n, q, \sigma)$: by running the N -KeyGen algorithm, we get a public key $h = g \cdot f^{-1} \in R_q^{\times}$ and a private key $\mathbf{sk} = \begin{bmatrix} f & g \\ F & G \end{bmatrix}$. The key h defines function $f_h(z) = f_h((z_1, z_2)) = hz_1 - z_2 \in R_q$ with domain $\mathcal{D}_n = \{z \in R^2 : \|z\| < s\sqrt{2n}\}$ and range $\mathfrak{R}_n = R_q$. The trapdoor of f_h is \mathbf{sk} .
- (2) SampleDom $(1^n, q, s)$: sample $z \leftarrow D_{R^2, s}$, if $\|z\| \geq s \cdot \sqrt{2n}$, resample.

- (i) **Input:** $n, q \in \mathbb{Z}^+, \sigma > 0$.
- (ii) **Output:** A key pair $(\mathbf{sk}, \mathbf{pk}) \in R^{2 \times 2} \times R_q^\times$.
- (1) Sample f from $D_{R, \sigma}$, if $(f \bmod q) \notin R_q^\times$, resample.
- (2) Sample g from $D_{R, \sigma}$, if $(g \bmod q) \notin R_q^\times$, resample.
- (3) If $\|f\| \geq \sqrt{n}\sigma$ or $\|g\| \geq \sqrt{n}\sigma$, restart.
- (4) If $(f, g) \neq R$, restart.
- (5) Compute $F_q, G_q \in R$ such that $f \cdot G_q - g \cdot F_q = q$, e.g., using a Hermite normal form algorithm.
- (6) Use Babai rounding nearest plane algorithm to approximate (F_q, G_q) in the lattice spanned by (f, g) , let $r(f, g)$ be the output, set $(F, G) = (F_q, G_q) - r(f, g)$ for some $r \in R$.
- (7) If $\|(F, G)\| > n\sigma\sqrt{l}$, restart.
- (8) Return secret key $\mathbf{sk} = \begin{bmatrix} f & g \\ F & G \end{bmatrix}$ and public key $\mathbf{pk} = h = g \cdot f^{-1} \in R_q^\times$.

ALGORITHM 1

- (3) **SamplePre** (\mathbf{sk}, t): to find a preimage in \mathcal{D}_n for a target $t \in \mathfrak{R}_n = R_q$ under f_h by using the trapdoor \mathbf{sk} , sample $z \leftarrow D_{\Lambda_h^q + c, s}$ with $\Lambda_h^q = \{(z_1, z_2) \in R^2 : z_2 = hz_1 \bmod qR\}$ and $c = (1, h - t)$. Return z .

Theorem 6. Assume $\sigma \geq \max\{8n^{3.6} \ln n, \omega(n \ln^{0.5} n) \cdot q^{(1/\varrho)}, \omega(n^{0.25} q^{0.5} l^{-0.25}), n^{3/2} \sqrt{\ln(8nq)} \cdot q^{((1/2)+\varepsilon)}\}$ for some $\varepsilon \in (0, (1/2))$ and $s \geq n^{3/2} \cdot \sigma \cdot \omega(\log n)$. Then, the constructed NTRUCRPSF (n, q, σ, s) is a CRPSF against $\text{ploy}(n)$ time adversaries, assuming the hardness of the worst-case Ideal-SIVP $_\gamma$ over K against $\text{ploy}(n)$ time adversaries, with $\gamma = \tilde{O}(n \cdot s)$.

We also need the following regularity theorem. For more details, one can refer to [26, 28, 29].

Theorem 7. Let K be a cyclotomic field with $[K : \mathbb{Q}] = n$, $R = \mathcal{O}_K$, $m \geq 2$, q is a positive prime such that $q \nmid \Delta_K$ and the prime ideal decomposition of qR in R is $qR = \mathcal{B}_1 \cdot \dots \cdot \mathcal{B}_\varrho$, $\delta \in (0, (1/2))$, $\varepsilon > 0$, and $a_i \leftarrow U(R_q^\times)$ for all $i \in [m]$. Assume $t \leftarrow D_{R^m, \sigma}$ with $\sigma \geq n \cdot \sqrt{(\ln(2mn(1 + (1/\delta))))/\pi} \cdot q^{(1/m)+\varepsilon}$. Then, we have

$$\Delta\left(\left(a_1, \dots, a_m, \sum_{i=1}^m t_i a_i\right); U\left(\left(R_q^\times\right)^m \times R_q\right)\right) \leq 2\delta + 2^{2m(n+\varrho)} q^{-\varepsilon mn}. \quad (2)$$

As in [28], we only use the powerful basis $\{\vec{p}_i\}_{i=1}^n$ of R and the decoding basis $\{\vec{d}_i\}_{i=1}^n$ of R^\vee . We mainly use the following definition and arrangements. More details can be found in [28].

Definition 4. Given a basis $B = \{b_1, \dots, b_n\}$ of a fractional ideal J , for any $x \in J$ with $x = x_1 b_1 + \dots + x_n b_n$, the B -coefficient embedding of x is defined as the vector (x_1, \dots, x_n) and the B -coefficient embedding norm of x is defined as $\|x\|_B^c = (\sum_{i=1}^n x_i^2)^{1/2}$.

Set $\hat{l} = l$ when l is odd and $\hat{l} = (l/2)$ when l is even. If $l = \prod_{i=1}^m p_i^{\alpha_i}$ for primes p_i , then we define $\text{rad}(l) = \prod_{i=1}^m p_i$. If

we represent $x \in R$ (or R^\vee) with respect to the powerful basis (or decoding basis), we have

$$\sqrt{\frac{l}{\text{rad}(l)}} \|x\|_{\sigma(\vec{p})}^c \leq \|\sigma(x)\| \leq \sqrt{\hat{l}} \|x\|_{\sigma(\vec{p})}^c, \quad \text{for } x \in R, \quad (3)$$

$$\frac{1}{\sqrt{\hat{l}}} \|x\|_{\sigma(\vec{d})}^c \leq \|\sigma(x)\| \leq \sqrt{\frac{\text{rad}(l)}{l}} \|x\|_{\sigma(\vec{d})}^c, \quad \text{for } x \in R^\vee. \quad (4)$$

We will omit the subscripts $\sigma(\vec{d})$ and $\sigma(\vec{p})$ in the following applications when it does not cause ambiguities.

When we write $x \bmod qR^\vee$, we use the representative element of the coset $x + qR^\vee$ as $\sum_{i=1}^N x_i \vec{d}_i$ with $x_i \in [-(q/2), (q/2))$. It is similar for element $x \in R$. Notice that $R \subseteq R^\vee$, and any element of R can also be represented as a \mathbb{Z} -linear combination of the decoding basis.

3. Identity-Based Encryption Schemes

In this section, we shall give the definition of IBE schemes and then construct a provably secure IBE scheme based on NTRU over any cyclotomic field.

3.1. Basic Definition and Security Model. We give the definition of IBE system first.

Definition 5. An identity-based encryption system consists of four PPT algorithms: **Setup**, **KeyGen**, **Encrypt**, and **Decrypt**.

- (i) **Setup** (λ): this algorithm takes as input a security parameter λ and generates public parameters \mathbf{PP} and a master secret key \mathbf{Msk} .
- (ii) **KeyGen** ($\mathbf{id}, \mathbf{Msk}, \mathbf{PP}$): this algorithm uses the master secret key \mathbf{Msk} to generate an identity private key $\mathbf{sk}_{\mathbf{id}}$ corresponding to an identity \mathbf{id} .
- (iii) **Encrypt** ($\mathbf{PP}, \mathbf{id}, m$): this algorithm takes the public parameters \mathbf{PP} to encrypt a message m for any given identity \mathbf{id} .
- (iv) **Decrypt** ($c, \mathbf{sk}_{\mathbf{id}}$): this algorithm decrypts ciphertext c by using the identity private key $\mathbf{sk}_{\mathbf{id}}$ if the identity of the ciphertext matches the identity of the private key.

The security model of IBE is defined through the following game between an adversary \mathcal{A} and a challenger \mathcal{B} . For a security parameter λ , let \mathcal{M}_λ be the plaintext space and

\mathcal{C}_λ be the ciphertext space. The game, which appraises the indistinguishability of plaintext under adaptive chosen-plaintext and adaptive chosen-identity attack (IND-ID-CPA), is defined as follows:

- (i) **Setup**: \mathcal{B} runs the algorithm **Setup** (λ) to get the public parameters \mathbf{PP} and the master secret key \mathbf{Msk} ; then, it sends \mathbf{PP} to \mathcal{A} and keeps the master secret key \mathbf{Msk} .
- (ii) **Phase 1**: \mathcal{A} adaptively issues private key queries q_1, \dots, q_k for identity $\mathbf{id}_1, \dots, \mathbf{id}_k$. In each query q_i for $i = 1, \dots, k$, \mathcal{B} runs **KeyGen** to generate $\mathbf{sk}_{\mathbf{id}_i}$ and sends it to \mathcal{A} .
- (iii) **Challenge**: once \mathcal{A} decides the **Phase 1** is over, it outputs a challenge identity \mathbf{id}^* , which has not been queried during **Phase 1**, and two plaintext message $m_0, m_1 \in \mathcal{M}_\lambda$. \mathcal{B} chooses a random element $b \in \{0, 1\}$ uniformly and sends $c_b = \mathbf{Encrypt}(\mathbf{PP}, \mathbf{id}^*, m_b)$ to \mathcal{A} .
- (iv) **Phase 2**: \mathcal{A} adaptively issues more private key queries q_{k+1}, \dots, q_Q for identity $\mathbf{id}_{k+1}, \dots, \mathbf{id}_Q$. The only requirement is that $\mathbf{id}^* \neq \mathbf{id}_i$ for any $i = k+1, \dots, Q$.
- (v) **Guess**: \mathcal{A} outputs an element $b' \in \{0, 1\}$ and wins if and only if $b' = b$.

We refer to such an adversary \mathcal{A} as an IND-ID-CPA adversary and define the advantage (in the security parameter λ) of \mathcal{A} in attacking an IBE scheme \mathcal{E} as $\text{Adv}_{\mathcal{E}, \mathcal{A}}(\lambda) = |\Pr(b' = b) - (1/2)|$.

Definition 6. For a security parameter λ , we say that an IBE scheme \mathcal{E} is adaptively IND-ID-CPA secure if for any PPT adversary \mathcal{A} that takes at most $Q = \text{poly}(\lambda)$ private key queries, $\text{Adv}_{\mathcal{E}, \mathcal{A}}(\lambda) \leq \text{negl}(\lambda)$.

3.2. Constructions of IBE Based on NTRU. Now, we can give the construction of IBE system over any cyclotomic field. The construction is inspired by [21], which follows the route of [10] and could be regarded as a generalization from power of 2 cyclotomic field to arbitrary cyclotomic field. The detailed construction is as follows, where Δ_K denotes the discriminant of K and $qR = \mathcal{B}_1, \dots, \mathcal{B}_g$.

- (i) **Setup** (λ): given a security parameter λ , first construct a set of parameters (K, R, q, σ, s) such that $K = \mathbb{Q}(\zeta_l)$ with $n = \varphi(l) \geq \lambda$, $R = \mathcal{O}_K$, and $q \geq 64n\zeta_K(2)$ such that $q \nmid \Delta_K$. Meanwhile, $\sigma \geq \max\{8n^{3.6} \ln n, \omega(n \ln^{0.5} n) \cdot q^{(1/g)}, \omega(n^{0.25} q^{0.5l-0.25}), n^{(3/2)} \sqrt{\ln(8nq)} \cdot q^{(1/2)+\epsilon}\}$ for some $\epsilon \in (0, (1/2))$, $s \geq n^{(3/2)} \cdot \sigma \cdot \omega(\log n)$. Then, call the N -KeyGen algorithm to generate a public key h and a secret key $\mathbf{sk} = \begin{bmatrix} f & g \\ F & G \end{bmatrix} \in R^{2 \times 2}$. Set the public parameters $\mathbf{PP} = (K, R, q, \sigma, R_q, R_q^\vee, h, H)$, where

$H : \{0, 1\}^* \mapsto R_q$ is a random oracle, and the master

$$\text{secret key } \mathbf{Msk} = \mathbf{sk} = \begin{bmatrix} f & g \\ F & G \end{bmatrix}.$$

- (ii) **KeyGen** ($\mathbf{id}, \mathbf{Msk}, \mathbf{PP}$): if the pair $(\mathbf{id}, \mathbf{sk}_{\mathbf{id}})$ is in the local storage, output $\mathbf{sk}_{\mathbf{id}}$ to the user \mathbf{id} . Otherwise,
 - (1) Set $t = H(\mathbf{id}) \in R_q$.
 - (2) Take $(\sigma_1, \sigma_2) = \text{SamplePre}(\mathbf{Msk}, t)$, where (σ_1, σ_2) satisfies $h\sigma_1 - \sigma_2 = t \bmod qR$.
 - (3) Output $\mathbf{sk}_{\mathbf{id}} = \sigma_1$ and keep the pair $(\mathbf{id}, \mathbf{sk}_{\mathbf{id}})$ in the local storage.
- (iii) **Encrypt** ($\mathbf{PP}, \mathbf{id}, m$): given a plaintext $m = \sum_{i=1}^n m_i \cdot \overrightarrow{d}_i \in R_q^\vee$ with coefficients $m_i \in \{0, 1\}$, the encryption process is as follows:
 - (1) Sample $r, e_1, e_2 \leftarrow \chi := [D_{\xi, q}]_{R^\vee}$ with $\xi = \alpha \cdot (nk/\log(nk))^{(1/4)}$, where $k = O(1)$ is a positive integer.
 - (2) Compute $t = H(\mathbf{id}) \in R_q$, $u = r \cdot h + e_1 \bmod qR^\vee$ and $v = t \cdot r + e_2 + (\lfloor q/4 \rfloor) \cdot m \bmod qR^\vee$.
 - (3) Output the ciphertext $c = (u, v)$.
- (iv) **Decrypt** ($c = (u, v), \mathbf{sk}_{\mathbf{id}}$): this algorithm first computes $w = v - u \cdot \mathbf{sk}_{\mathbf{id}} = \sum_{i=1}^n w_i \cdot \overrightarrow{d}_i \bmod qR^\vee$ and returns $m = \sum_{i=1}^n \lfloor (4/q) \cdot w_i \rfloor \cdot \overrightarrow{d}_i \bmod qR^\vee$.

Note that we have $w = v - u\sigma_1 = rt + e_2 + (\lfloor q/4 \rfloor) \cdot m - r h \sigma_1 - e_1 \sigma_1 = (\lfloor q/4 \rfloor) \cdot m + e \bmod qR^\vee$ where $e = e_2 - r\sigma_2 - e_1 \sigma_1 \in R^\vee$ for some (σ_1, σ_2) satisfying $h\sigma_1 - \sigma_2 = t \bmod qR$. If $\|e\|_\infty^c < q/10$, then we get that w has the representation of the form $(\lfloor q/4 \rfloor) \cdot m + e$ in R_q^\vee . Setting $w = \sum_{i=1}^n w_i \cdot \overrightarrow{d}_i$ and $e = \sum_{i=1}^n e'_i \cdot \overrightarrow{d}_i$, we can conclude that for any $q > 40$,

$$\frac{4}{q} w_i = \frac{4}{q} \lfloor \frac{q}{4} \rfloor \cdot m_i + \frac{4}{q} e'_i = \begin{cases} \frac{4}{q} e'_i \in \left(-\frac{2}{5}, \frac{2}{5}\right), & \text{if } m_i = 0, \\ \frac{4}{q} \lfloor \frac{q}{4} \rfloor + \frac{4}{q} e'_i \in \left(\frac{1}{2}, \frac{3}{2}\right), & \text{if } m_i = 1. \end{cases} \quad (5)$$

Therefore, the decryption process succeeds in recovering the encrypted message m whenever $\|e_2 - r\sigma_2 - e_1 \sigma_1\|_\infty^c < (q/10)$. Now, we bound the probability that $\|e_2 - r\sigma_2 - e_1 \sigma_1\|_\infty^c \geq (q/10)$. Here, $\|\cdot\|_\infty^c$ represents the basis-coefficient norm under the decoding basis with respect to the l_∞ norm.

Lemma 4. Assume that $\alpha \in (0, 1)$ such that $\alpha \leq \sqrt{(\log n/n)}$ and let $q \geq 2$ be an integer such that $\alpha q \geq \omega(1)$; meanwhile, $\omega(n^{(3/2)} \sqrt{\log n \log \log n} \cdot \alpha^2 \cdot q^2 \cdot s) < (q/30\sqrt{2})$; then, we have $\|e_2 - r\sigma_2 - e_1 \sigma_1\|_\infty^c < (q/10)$ with probability at least $1 - n^{-\omega(\sqrt{n \log n})}$.

Proof. Lemma 5.1 of [28] implies that $\Pr_{x \leftarrow \chi}[\|x\|_\infty > \omega(\sqrt{n \log n} \cdot \alpha^2 \cdot q^2)] \leq n^{-\omega(\sqrt{n \log n})}$. Note that $\|(\sigma_1, \sigma_2)\| \leq \sqrt{2n} \cdot s$; we have

$$\begin{aligned}
\|e_2 - r\sigma_2 - e_1\sigma_1\|_\infty^c &\leq \sqrt{\hat{l}} \cdot (\|e_2\| + \|r \cdot \sigma_2\| + \|e_1 \cdot \sigma_1\|) \\
&\leq \sqrt{\hat{l}} \cdot \left(\sqrt{n} \cdot \|e_2\|_\infty + \|r\|_\infty \cdot \|\sigma_2\| \right. \\
&\quad \left. + \|e_1\|_\infty \cdot \|\sigma_1\| \right). \quad (6)
\end{aligned}$$

Therefore, we get

$$\|e_2 - r\sigma_2 - e_1\sigma_1\|_\infty^c \leq 3\sqrt{2} \cdot \omega' \left(n^{(3/2)} \sqrt{\log n \log \log n} \cdot \alpha^2 \cdot q^2 \cdot s \right), \quad (7)$$

with probability at least $1 - n^{-\omega(\sqrt{n \log n})}$, where we have used that $\sqrt{\hat{l}} = O(\sqrt{n \log \log n})$.

Overall, we get the following lemma. \square

Lemma 5. *Assume that $\alpha \in (0, 1)$ such that $\alpha \leq \sqrt{(\log n/n)}$ and let $q \geq 2$ be an integer such that $\alpha q \geq \omega(1)$; meanwhile, $\omega(n^{(3/2)} \sqrt{\log n \log \log n} \cdot \alpha^2 \cdot q^2 \cdot s) < (q/30\sqrt{2})$; then, the decryption algorithm of the IBE scheme succeeds in recovering the encrypted message with probability at least $1 - n^{-\omega(\sqrt{n \log n})}$.*

We can prove that our IBE scheme is secure, assuming that $R - DLWE_{q, LD_{q\zeta}^{\times} \Gamma_{R^V}}$ problem and $R - DLWE_{q, LD_{q\zeta}^{\times} \Gamma_{R^V}}$ problem are hard. We first give a IND-CPA secure public key encryption scheme (denoted by BasicPub). Note that Lemma 5 is suitable for BasicPub as well.

- (i) **Setup** (λ): given a security parameter λ , do as the **Setup** algorithm of IBE scheme. Set the public parameters $\mathbf{PP} = (K, R, q, \sigma, R_q, R_q^V, h)$.
- (ii) **KeyGen** (\mathbf{PP}): sample $(\sigma_1, \sigma_2) = \text{SampleDom}(\mathbf{PP})$; set the secret key $sk = \sigma_1$ and the public key $pk = h\sigma_1 - \sigma_2 \bmod qR$.
- (iii) **Encrypt** (\mathbf{PP}, pk, m): do as the **Encrypt** algorithm of IBE scheme with $t = pk$.
- (iv) **Decrypt** ($c = (u, v), sk$): the same as the **Decrypt** algorithm of IBE scheme.

Lemma 6. *Let $K = \mathbb{Q}(\zeta_l)$ be a cyclotomic field, $n = \varphi(l)$, $R = \mathcal{O}_K$, and $q \geq 64n\zeta_K(2)$ be a prime such that $q \nmid \Delta_K$. Set $\sigma \geq \max\{8n^{3.6} \ln n, \omega(n \ln^{0.5} n) \cdot q^{(1/\varphi)}, \omega(n^{0.25} q^{0.5} l^{-0.25}), n^{3/2} \sqrt{\ln(8nq)} \cdot q^{(1/2)+\varepsilon}\}$ for some $\varepsilon \in (0, (1/2))$ and $s \geq n^{3/2} \cdot \sigma \cdot \omega(\log n)$; meanwhile, assume that $\alpha \in (0, 1)$ such that $\alpha \leq \sqrt{(\log n/n)}$, $\alpha q \geq \omega(1)$, and $\omega(n^{3/2} \sqrt{\log n \log \log n} \cdot \alpha^2 \cdot q^2 \cdot s) < (q/30\sqrt{2})$. Then, the BasicPub is IND-CPA secure assuming that $R - DLWE_{q, LD_{q\zeta}^{\times} \Gamma_{R^V}}$ problem and $R - DLWE_{q, LD_{q\zeta}^{\times} \Gamma_{R^V}}$ problem are hard.*

Proof. Note that, by the property of SampleDom algorithm, the distribution of pk is statistically close to $U(R_q)$. Then, for a ciphertext (u, v) of either m_0 or m_1 , by our choices of

parameters, the entire view $(h, pk, u, v) \in R_q^{\times} \times R_q \times R_q \times R_q^V$ of the adversary is indistinguishable from the uniform distribution, assuming the hardness of $R - DLWE_{q, LD_{q\zeta}^{\times} \Gamma_{R^V}}$ problem and $R - DLWE_{q, LD_{q\zeta}^{\times} \Gamma_{R^V}}$ problem. Hence, the adversary could not distinguish the ciphertexts of 0 and 1. We get the results, as desired. \square

Theorem 8. *Suppose that Lemma 6 holds, i.e., the BasicPub is correct and IND-CPA secure in the standard model; then, the IBE scheme is adaptively IND-ID-CPA secure in the random oracle model.*

Proof. Let \mathcal{A} be a PPT adversary that attacks the IBE scheme with advantage δ by using $Q = \text{poly}(n)$ distinct H queries. We shall construct an algorithm \mathcal{B} to attack the BasicPub scheme with advantage (δ/Q) . The algorithm \mathcal{B} works as follows:

- (1) \mathcal{B} calls an oracle (or the challenger) to get the public parameters $\mathbf{PP}' = (K, R, q, \sigma, R_q, R_q^V, h)$ and a public key pk . Then, it sends the public parameters $\mathbf{PP} = (K, R, q, \sigma, R_q, R_q^V, h, H)$ to \mathcal{A} . Here, \mathcal{B} simulates the random oracle H ; meanwhile, \mathcal{B} chooses an $i \in [Q]$ uniformly at random.
- (2) \mathcal{B} simulates the view of \mathcal{A} as follows:
 - (i) **Hash queries:** on \mathcal{A} 's j th distinct query \mathbf{id}_j to H , if $j = i$, then store the tuple $(\mathbf{id}_i, pk, \perp)$ and return pk to \mathcal{A} . Otherwise, $j \neq i$, \mathcal{A} runs the BasicPub.KeyGen (\mathbf{PP}') to generate a public/secret key pair (\mathbf{sk}_j, pk_j) , locally store the tuple $(\mathbf{id}_j, pk_j, \mathbf{sk}_j)$, and return \mathbf{sk}_j to \mathcal{A} .
 - (ii) **KeyGen queries:** when \mathcal{A} asks for a secret key for an identity \mathbf{id} , assume without loss of generality that \mathcal{A} has already queried H on \mathbf{id} . Retrieve the unique tuple $(\mathbf{id}, pk, \mathbf{sk})$ from local storage. If $\mathbf{sk} = \perp$, then output a random bit and abort. Otherwise, return \mathbf{sk} to \mathcal{A} .
- (3) When \mathcal{A} produces a challenge identity \mathbf{id}^* which is distinct from all its secret key queries and two messages m_0, m_1 , assume without loss of generality that \mathcal{A} has already queried H on \mathbf{id}^* . If $\mathbf{id}^* \neq \mathbf{id}_i$, output a random bit and abort. Otherwise, return $c_b = \text{BasicPub.Encrypt}(\mathbf{PP}', pk, m_b)$ for $b \leftarrow U(\{0, 1\})$ to \mathcal{A} .

When \mathcal{A} terminates with some output, \mathcal{B} terminates with the same output.

Assume \mathcal{A} makes N distinct **KeyGen queries** for some $N \leq Q$. Notice that the probability that \mathcal{B} does not abort is

$$\begin{aligned}
\Pr &= \left(1 - \frac{1}{Q}\right) \cdot \left(1 - \frac{1}{Q-1}\right) \cdots \left(1 - \frac{1}{Q-(N-1)}\right) \\
&\quad \cdot \frac{1}{Q-N} = \frac{1}{Q}. \quad (8)
\end{aligned}$$

Meanwhile, conditioned on \mathcal{B} not aborting, the view it provides to \mathcal{A} is statistically close to the view of the real IBE scheme. Hence, the advantage that \mathcal{B} attacks the IND-CPA secure of BasicPub is (δ/Q) , as desired.

Overall, we conclude the following theorem. \square

Theorem 9. *Let $K = \mathbb{Q}(\zeta_l)$ be a cyclotomic field, $n = \varphi(l)$, $R = \mathcal{O}_K$, and $q \geq 64n\zeta_K(2)$ be a prime such that $q \nmid \Delta_K$. Set $\sigma \geq \max\{8n^{3.6} \ln n, \omega(n \ln^{0.5} n) \cdot q^{1/g}, \omega(n^{0.25} q^{0.5} l^{-0.25}), n^{3/2} \sqrt{\ln(8nq)} \cdot q^{(1/2)+\varepsilon}\}$ for some $\varepsilon \in (0, (1/2))$ and $s \geq n^{3/2} \cdot \sigma \cdot \omega(\log n)$; meanwhile, assume that $\alpha \in (0, 1)$ such that $\alpha \leq \sqrt{(\log n/n)}$, $\alpha q \geq \omega(1)$, and $\omega(n^{3/2} \sqrt{\log n \log \log n} \cdot \alpha^2 \cdot q^2 \cdot s) < (q/30\sqrt{2})$. Then, the IBE scheme is adaptively IND-ID-CPA secure against any PPT adversary in the random oracle model, assuming the hardness of worst-case Ideal-SIVP $_\gamma$ over K against PPT adversaries, with $\gamma = \tilde{O}(n^2 \cdot s)$.*

Remark 1. If we choose $\alpha q = \omega(1)$, then $s = \tilde{O}(n^{7.5})$, $q = \tilde{O}(n^9)$ and $\gamma = \tilde{O}(n^{9.5})$. As remarked in [28], we can also convert our constructions to work in an ideal of R , or we can directly design our schemes in R (with larger γ and q). Moreover, when we require that $q = 1 \pmod l$ with l having some special cases (for example, $l = p^\alpha, 2^\alpha p$ or $2^\alpha pq$ for some prime p, q), we can use the hardness results shown in [35] and techniques shown in [36] to reduce the magnitude of the parameters q and γ . Usually, the module q is far away from practicality. A heuristic practical choice of parameters (with respect to coefficient embedding) is shown in [21]. How to reduce the size of q and γ is a hard problem which is worth studying.

4. Identity-Based Signature Schemes

In this section, we shall give the definition of IBS schemes and then construct a provably secure IBS scheme based on NTRU over any cyclotomic field.

4.1. Basic Definition and Security Model. We give the definition of IBS system first.

Definition 7. An identity-based signature system consists of four PPT algorithms: **Setup**, **KeyGen**, **Sign**, and **Verification**.

- (i) **Setup** (λ): this algorithm takes as input a security parameter λ and generates public parameters **PP** and a master secret key **Msk**.
- (ii) **KeyGen** (**id**, **Msk**, **PP**): this algorithm uses the master secret key **Msk** to generate an identity private key $\mathbf{sk}_{\mathbf{id}}$ corresponding to an identity **id**.
- (iii) **Sign** (**PP**, **id**, $\mathbf{sk}_{\mathbf{id}}$, μ): this algorithm takes the public parameters **PP**, a message μ , an identity **id**,

and the secret key $\mathbf{sk}_{\mathbf{id}}$ to generate a signature **Sig** of μ .

- (iv) **Verification** (**PP**, μ , **Sig**, **id**): on input of the identity **id**, the message μ , the parameters **PP**, and a signature **Sig**, this algorithm outputs 1 when the verification is correct (i.e., the signature is valid) and outputs 0 otherwise.

The security model of IBS is defined through the following game between an adversary \mathcal{A} and a challenger \mathcal{B} . For a security parameter λ , let \mathcal{M}_λ be the message space and \mathcal{S}_λ be the signature space. The game, which appraises the property of existentially unforgeable against adaptively chosen message and adaptively chosen identity attacks, is defined as follows:

- (i) **Setup**: \mathcal{B} runs the algorithm **Setup** (λ) to get the public parameters **PP** and the master secret key **Msk**; then, it sends **PP** to \mathcal{A} and keeps the master secret key **Msk**.
- (ii) **Phase 1**: \mathcal{A} adaptively issues private key queries q_1, \dots, q_k for identity $\mathbf{id}_1, \dots, \mathbf{id}_k$. In each query q_i for $i = 1, \dots, k$, \mathcal{B} runs **KeyGen** to generate $\mathbf{sk}_{\mathbf{id}_i}$ and sends it to \mathcal{A} .
- (iii) **Challenge**: once \mathcal{A} decides the **Phase 1** is over, it outputs an identity \mathbf{id}^* , which has not been queried during **Phase 1**.
- (iv) **Phase 2**: \mathcal{A} adaptively issues more queries q_{k+1}, \dots, q_Q where each query q_i is one of the following:
 - (1) Private key query for $\mathbf{id}_i \neq \mathbf{id}^*$: \mathcal{B} responds as in **Phase 1**.
 - (2) Signature query for a message μ under identity \mathbf{id}^* : this query can be regarded as an oracle, and \mathcal{B} runs the oracle to get a signature **Sig** = **Sign**(**PP**, \mathbf{id}^* , $\mathbf{sk}_{\mathbf{id}^*}$, μ) and sends **Sig** to \mathcal{A} .
- (v) **Forge**: \mathcal{A} outputs a forge \mathbf{Sig}^* for a message μ under identity \mathbf{id}^* . It wins if and only if one of the following two cases happens:
 - (1) If μ is queried in **Phase 2**, then we require that $\mathbf{Sig}^* \neq \mathbf{Sig}$, where **Sig** is the signature of μ that \mathcal{A} got in **Phase 2**. Meanwhile, **Verification** (**PP**, μ , \mathbf{Sig}^* , \mathbf{id}^*) = 1.
 - (2) Otherwise, we simply require that **Verification** (**PP**, μ , \mathbf{Sig}^* , \mathbf{id}^*) = 1.

We define the advantage (in the security parameter λ) of \mathcal{A} in attacking an IBS scheme \mathcal{E} as $\text{Adv}_{\mathcal{E}, \mathcal{A}}(\lambda) = |\text{Pr}(\mathcal{A} \text{ wins}) - (1/2)|$.

Definition 8. For a security parameter λ , we say that an IBS scheme \mathcal{E} is existentially unforgeable against adaptively chosen message and adaptively chosen identity attacks if for any PPT adversary \mathcal{A} that takes at most $Q = \text{poly}(\lambda)$ queries, $\text{Adv}_{\mathcal{E}, \mathcal{A}}(\lambda) \leq \text{negl}(\lambda)$.

4.2. *Constructions of IBS Based on NTRU.* Now, we can give the construction of IBS system over any cyclotomic field. The detailed construction is as follows:

- (i) **Setup** (λ): given a security parameter λ , first construct a set of parameters (K, R, q, σ, s) such that $K = \mathbb{Q}(\zeta_l)$ with $n = \varphi(l) \geq \lambda$, $R = \mathcal{O}_K$, and $q \geq 64n\zeta_K(2)$ such that $q \nmid \Delta_K$. Meanwhile, $\sigma \geq \max\{8n^{3.6} \ln n, \omega(n \ln^{0.5} n) \cdot q^{1/q}, \omega(n^{0.25} q^{0.5} l^{-0.25}), n^{3/2} \sqrt{\ln(8nq)} \cdot q^{(1/2)+\varepsilon}\}$ for some $\varepsilon \in (0, (1/2))$, $s \geq n^{3/2} \cdot \sigma \cdot \omega(\log n)$. Then, call the N -KeyGen algorithm to generate a public key h and a secret key

$$\mathbf{sk} = \begin{bmatrix} f & g \\ F & G \end{bmatrix} \in R^{2 \times 2}. \text{ Set the public parameters}$$

$$\mathbf{PP} = (K, R, q, \sigma, R_q, R_q^\vee, h, H, H'), \text{ where}$$

$H: \{0, 1\}^* \mapsto R_q$ and $H': R_q \times \{0, 1\}^* \mapsto R_q$ are two random oracles, and the master secret key $\mathbf{Msk} =$

$$\mathbf{sk} = \begin{bmatrix} f & g \\ F & G \end{bmatrix}.$$

- (ii) **KeyGen** ($\mathbf{id}, \mathbf{Msk}, \mathbf{PP}$): if the pair $(\mathbf{id}, \mathbf{sk}_{\mathbf{id}})$ is in the local storage, output $\mathbf{sk}_{\mathbf{id}}$ to the user \mathbf{id} . Otherwise,

- (1) Set $t = H(\mathbf{id}) \in R_q$.
- (2) Take $(\sigma_1, \sigma_2) = \text{SamplePre}(\mathbf{Msk}, t)$, where (σ_1, σ_2) satisfies $h\sigma_1 - \sigma_2 = t \pmod{qR}$.
- (3) Output $\mathbf{sk}_{\mathbf{id}} = (\sigma_1, \sigma_2)$ and keep the pair $(\mathbf{id}, \mathbf{sk}_{\mathbf{id}})$ in the local storage.

- (iii) **Sign** ($\mathbf{PP}, \mathbf{id}, \mathbf{sk}_{\mathbf{id}}, \mu$): given a message μ , the signature process is as follows:

- (1) Sample $y_1, y_2 \leftarrow D_{R, s}$.
- (2) Compute $u = H'(hy_1 - y_2 \pmod{qR}, \mu) \in R_q$ and $z_i = y_i + \sigma_i \cdot u$ for $i = 1, 2$.
- (3) Output the signature $\text{Sig} = (z_1, z_2, u)$ of message μ with probability $\min((D_{R^2, s}(z)/M \cdot D_{R^2, s, v}(z)), 1)$ with $v = (\sigma_1 u, \sigma_2 u)$ and $M = O(1)$ (in practice, M can be computed efficiently).

- (iv) **Verification** ($\mathbf{PP}, \mu, \text{Sig}, \mathbf{id}$): for $\text{Sig} = (z_1, z_2, u)$, if $\|(z_1, z_2)\| \leq \sqrt{2n} \cdot s$ and $H'(hz_1 - z_2 - H(\mathbf{id}) \cdot u \pmod{qR}, \mu) = u \in R_q$, output 1. Otherwise, output 0.

The signing algorithm outputs something with probability $\min((D_{R^2, s}(z)/M \cdot D_{R^2, s, (\sigma_1 u, \sigma_2 u)}(z)), 1)$, if nothing was output, the signer runs the signing algorithm again until some signature is outputted. Note that $hz_1 - z_2 - H(\mathbf{id}) \cdot u = hy_1 - y_2 + (h\sigma_1 - \sigma_2) \cdot u - H(\mathbf{id}) \cdot u = hy_1 - y_2 \pmod{qR}$. Meanwhile, Lemma 2 and Theorem 2.2 imply that $\|(z_1, z_2)\| \leq \sqrt{2n} \cdot s$ with overwhelming probability. We conclude the following lemma.

Lemma 7. *The IBS scheme proposed above satisfies correctness.*

The security of the IBS scheme can be reduced to the worst-case SIVP $_\gamma$ problem over K .

Theorem 10. *Let $K = \mathbb{Q}(\zeta_l)$ be a cyclotomic field, $n = \varphi(l)$, $R = \mathcal{O}_K$, and $q \geq 64n\zeta_K(2)$ be a prime such that $q \nmid \Delta_K$. Assume that $\sigma \geq \max\{8n^{3.6} \ln n, \omega(n \ln^{0.5} n) \cdot q^{1/q}, \omega(n^{0.25} q^{0.5} l^{-0.25}), n^{3/2} \sqrt{\ln(8nq)} \cdot q^{(1/2)+\varepsilon}\}$ for some $\varepsilon \in (0, (1/2))$, $s \geq n^{3/2} \cdot \sigma \cdot \omega(\log n)$. The IBS scheme is existentially unforgeable against adaptively chosen message and adaptively chosen identity attacks for any PPT adversary in the random oracle model, assuming the hardness of worst-case Ideal-SIVP $_\gamma$ over K against PPT adversaries, with $\gamma = \tilde{O}(n \cdot s)$.*

Proof. Suppose that there is an adversary \mathcal{A} which can break the existentially unforgeable IBS scheme with advantage δ ; we can construct an algorithm \mathcal{B} to solve the R -SIS $_{q, 2, \beta}$ problem over K for $\beta = 2\sqrt{2n} \cdot s$. The interactions between \mathcal{B} and \mathcal{A} are described as follows:

- (1) For an R -SIS $_{q, 2, \beta}$ instance (a_1, a_2) , if $(a_1, a_2) \notin (R_q^\times)^2$, abort. Otherwise, \mathcal{B} sends $h = a_2^{-1} \cdot a_1 \pmod{qR} \leftarrow U(R_q^\times)$ to \mathcal{A} .
- (2) \mathcal{A} can adaptively query in the following ways. In general, we can assume that \mathcal{A} has to query the random oracle H for \mathbf{id} before it makes other kinds of queries.

- (i) H query: at the beginning, \mathcal{B} keeps an ID-list which consists of elements of the form $(\mathbf{id}, t_{\mathbf{id}}, \mathbf{sk}_{\mathbf{id}})$. The list is empty initially. For a query of identity \mathbf{id}^* , if it is contained in the ID-list, \mathcal{B} simply sends $t_{\mathbf{id}^*}$ to \mathcal{A} . Otherwise, \mathbf{id}^* is fresh. \mathcal{B} samples $z = (\sigma_1, \sigma_2) \leftarrow D_{R^2, s}$ and computes $t_{\mathbf{id}^*} = h\sigma_1 - \sigma_2 \pmod{qR}$. Then, \mathcal{B} sends $t_{\mathbf{id}^*}$ to \mathcal{A} and stores $(\mathbf{id}^*, t_{\mathbf{id}^*}, \mathbf{sk}_{\mathbf{id}^*} = (\sigma_1, \sigma_2))$ in the ID-list.

- (ii) **KeyGen** query: given \mathbf{id}^* , \mathcal{B} looks up the ID-list to find $\mathbf{sk}_{\mathbf{id}^*}$ corresponding to \mathbf{id}^* and sends $\mathbf{sk}_{\mathbf{id}^*}$ to \mathcal{A} .

- (iii) **Sign** query: \mathcal{B} also keeps a SIGN-list which is empty initially and consists of elements of the form $(\mu, \mathbf{id}, (y_1, y_2), \mathbf{sk}_{\mathbf{id}}, u, (z_1, z_2))$. To obtain the signature of message $\mu^* \in (0, 1)^*$ under the identity \mathbf{id}^* , if (μ^*, \mathbf{id}^*) is in the SIGN-list, \mathcal{B} simply sends (z_1^*, z_2^*, u^*) to \mathcal{A} . Otherwise, μ^* is fresh and \mathcal{B} looks up the ID-list for $\mathbf{sk}_{\mathbf{id}^*}$ and runs **Sign** ($\mathbf{PP}, \mathbf{id}^*, \mathbf{sk}_{\mathbf{id}^*}, \mu^*$) to get a signature (z_1^*, z_2^*, u^*) . \mathcal{B} sends (z_1^*, z_2^*, u^*) to \mathcal{A} and stores $(\mu^*, \mathbf{id}^*, (y_1^*, y_2^*), \mathbf{sk}_{\mathbf{id}^*}, u^*, (z_1^*, z_2^*))$ in the SIGN-list. Here, (y_1^*, y_2^*) is obtained through the algorithm **Sign** ($\mathbf{PP}, \mathbf{id}^*, \mathbf{sk}_{\mathbf{id}^*}, \mu^*$).

- (iv) H' query: when \mathcal{A} sends a message μ^* under identity \mathbf{id}^* to \mathcal{B} for the H' query, \mathcal{B} finds the corresponding u^* in the SIGN-list and sends it to \mathcal{A} (if μ^* is not in the SIGN-list, \mathcal{B} implements **Sign** query for (μ^*, \mathbf{id}^*) and sends corresponding u^* obtained by **Sign** query to \mathcal{A}).

- (3) Forge: after finishing the queries listed above, \mathcal{A} outputs a forgery $(z_1^{*'}, z_2^{*'}, u^{*'})$ for (\mathbf{id}^*, μ^*) with a nonnegligible probability δ .

Note that, without loss of generality, we can assume that before outputting the attempted forgery $(z_1^{*'}, z_2^{*'}, u^{*'})$, \mathcal{A} has made a query for **Sign** (or strictly speaking, \mathcal{A} has made a query for H' , but a H' query is equivalent to a **Sign** query, by our constructions), i.e. $u^{*'}$ is for a u^* in the SIGN-list. \mathcal{B} can get (z_1^*, z_2^*) from the SIGN-list, which satisfies $H'(hz_1^* - z_2^* - H(\mathbf{id}^*) \cdot u^*, \mu^*) = H'(hz_1^{*'}$ $- z_2^{*'}$ $- H(\mathbf{id}^*) \cdot u^{*'}, \mu^*) = u^* = u^{*'}$. Hence, we have $hz_1^* - z_2^* - H(\mathbf{id}^*) \cdot u^* = hz_1^{*'}$ $- z_2^{*'}$ $- H(\mathbf{id}^*) \cdot u^{*'}$ $\text{mod } qR$ (up to a negligible probability). Therefore, $a_1(z_1^* - z_1^{*'}) + a_2(z_2^* - z_2^{*'}) = 0 \text{ mod } qR$. Let $z = (z_1^* - z_1^{*'}, z_2^* - z_2^{*'})$; we have $\|z\|^2 = \|z_1^* - z_1^{*'}\|^2 + \|z_2^* - z_2^{*'}\|^2 \leq 8ns^2$. Hence, if $z \neq 0$, it is a valid solution of R-SIS $_{q,2,2\sqrt{2n}\cdot s}$.

Also, note that in order to give a valid forge, \mathcal{A} needs to find $(z_1^{*'}, z_2^{*'})$ to fulfil that $\|(z_1^{*'}, z_2^{*'})\| \leq \sqrt{2n} \cdot s$ and $hz_1^{*'}$ $- z_2^{*'}$ $= w \text{ mod } qR$ for $w = hz_1^* - z_2^* \text{ mod } qR$. Theorem 2.2 implies that we can regard $z_i^* \leftarrow D_{R,s}$. Theorem 2.7 implies that $w \leftarrow U(R_q)$. For any $w \in R_q$, the solutions of the equation $hx_1 - x_2 = w \text{ mod } qR$ form a lattice $\Lambda^i = (z_1^*, z_2^*) + \Lambda_h^q$. Hence, for the parameter choices of s and σ , Lemma 3 indicates that the probability that $z = 0$ is negligible. Therefore, except with some negligible probability $\varepsilon(n)$, we can solve R-SIS $_{q,2,2\sqrt{2n}\cdot s}$ with advantage $\delta^i = (1 - \varepsilon(n))\delta$. \square

Remark 2. By the conditions in Theorem 4.1, we can take $s = \tilde{O}(n^7)$, $q = \tilde{O}(n^8)$ and $\gamma = \tilde{O}(n^8)$. Also, the module q is far away from practicality. How to reduce the size of q and γ is a hard problem which is worth studying.

One may note that the trapdoor generation algorithms used in IBE and IBS schemes are the same, so as the case of IBE in power-of-2 cyclotomic rings; we can also use the parameter choices (with respect to coefficient embedding) as in [21], together with the parameter choices of rejection sampling as in [27] to give a practical implementation of our schemes. A more heuristic implementation with respect to coefficient embedding in power-of-2 cyclotomic rings is also shown in [17].

Appendix

We first introduce a useful ‘‘rejection sampling’’ lemma which is a modified version of Lemma 4.7 in [27]. Their proof is essentially the same.

Lemma 8. *Let $V \subseteq H$ be an arbitrary set and $\Lambda \subseteq H$ be an arbitrary lattice. Assume $h : V \mapsto [0, 1]$ and $f : \Lambda \mapsto [0, 1]$*

be probability distributions. If $g_v : \Lambda \mapsto [0, 1]$ is a family of probability distributions indexed by all $v \in V$ with the property that

$$\exists M \in \mathbb{R} \text{ such that } \forall v, \Pr_{z \leftarrow f} [M \cdot g_v(z) \geq f(z)] \geq 1 - \varepsilon, \quad (\text{A.1})$$

then the distribution of the output of the following algorithm \mathcal{A} :

- (1) $v \leftarrow h$
- (2) $z \leftarrow g_v$
- (3) output (z, v) with probability $\min(f(z)/M \cdot g_v(z), 1)$

is within statistical distance (ε/M) of the distribution of the output of the following algorithm \mathcal{F} :

- (1) $v \leftarrow h$
- (2) $z \leftarrow f$
- (3) output (z, v) with probability $1/M$.

Moreover, the probability p that \mathcal{A} outputs something satisfies $p \in [(1 - \varepsilon)/M, (1/M)]$.

Proof. For each $v \in V$, define S_v to be the set that consists of all $z \in \Lambda$ such that $M \cdot g_v(z) \geq f(z)$. Notice that by definition, for all $z \in S_v$, the probability that \mathcal{A} outputs z is $g_v(z) \min(f(z)/M \cdot g_v(z), 1) = (f(z)/M)$ and for all $z \notin S_v$, the probability that z is output is $g_v(z)$. Let p denote the probability that \mathcal{A} outputs something. Then, we have

$$p = \sum_{v \in V} h(v) \left(\sum_{z \in S_v} \frac{f(z)}{M} + \sum_{z \notin S_v} g_v(z) \right) \quad (\text{A.2})$$

$$\geq \sum_{v \in V} h(v) \sum_{z \in S_v} \frac{f(z)}{M} \geq \frac{1 - \varepsilon}{M},$$

$$p = \sum_{v \in V} h(v) \left(\sum_{z \in S_v} \frac{f(z)}{M} + \sum_{z \notin S_v} g_v(z) \right) \quad (\text{A.3})$$

$$\leq \sum_{v \in V} h(v) \left(\sum_{z \in S_v} \frac{f(z)}{M} + \sum_{z \notin S_v} \frac{f(z)}{M} \right) = \frac{1}{M}.$$

For the estimation of the statistical distance of the distribution of the output of \mathcal{A} and \mathcal{F} , let $N_{\mathcal{A}}$ and $N_{\mathcal{F}}$ be the probabilities that \mathcal{A} and \mathcal{F} do not output anything, respectively. It is obvious that $N_{\mathcal{F}} = 1 - (1/M)$ and $1 - (1/M) \leq N_{\mathcal{A}} \leq 1 - (1 - \varepsilon)/M$. Then, we have

$$\begin{aligned}
\Delta(\mathcal{A}, \mathcal{F}) &= \frac{1}{2} \left(\sum_{z \in \Lambda, v \in V} |\mathcal{A}(z, v) - \mathcal{F}(z, v)| + |N_{\mathcal{A}} - N_{\mathcal{F}}| \right) \\
&= \frac{1}{2} \left(\sum_{z \in \Lambda} \sum_{v \in V} \left| h(v) g_v(z) \min\left(\frac{f(z)}{M g_v(z)}, 1\right) - h(v) \frac{f(z)}{M} \right| + |N_{\mathcal{A}} - N_{\mathcal{F}}| \right) \\
&= \frac{1}{2} \left(\sum_{z \in \Lambda} \sum_{v \in V} h(v) \left| g_v(z) \min\left(\frac{f(z)}{M g_v(z)}, 1\right) - \frac{f(z)}{M} \right| + |N_{\mathcal{A}} - N_{\mathcal{F}}| \right) \\
&= \frac{1}{2} \sum_{v \in V} h(v) \left(\sum_{z \in \Lambda} \left| g_v(z) \min\left(\frac{f(z)}{M g_v(z)}, 1\right) - \frac{f(z)}{M} \right| + |N_{\mathcal{A}} - N_{\mathcal{F}}| \right) \\
&= \frac{1}{2} \sum_{v \in V} h(v) \left(\sum_{z \in \mathcal{S}_v} \left| \frac{f(z)}{M} - \frac{f(z)}{M} \right| + \sum_{z \notin \mathcal{S}_v} \left| g_v(z) - \frac{f(z)}{M} \right| + |N_{\mathcal{A}} - N_{\mathcal{F}}| \right) \\
&\leq \frac{1}{2} \sum_{v \in V} h(v) \left(\sum_{z \notin \mathcal{S}_v} \frac{f(z)}{M} + |N_{\mathcal{A}} - N_{\mathcal{F}}| \right) \leq \frac{1}{2} \sum_{v \in V} h(v) \left(\frac{\varepsilon}{M} + \left(\left(1 - \frac{1-\varepsilon}{M}\right) - \left(1 - \frac{1}{M}\right) \right) \right), \\
&= \frac{\varepsilon}{M}.
\end{aligned} \tag{A.4}$$

The proof is finished.

The following lemma is helpful for us to estimate the upper bound of $|\langle z, v \rangle|$ for any $v \in \Lambda \subseteq H$ and $z \leftarrow D_{\Lambda, \sigma}$. \square

Lemma 9. For any lattice $\Lambda \subseteq H$, $v \in \Lambda$ and $t > 0$, we have

$$\Pr_{z \leftarrow D_{\Lambda, \sigma}} [|\langle z, v \rangle| > t] \leq 2 \cdot e^{-(\pi t^2 / \|v\|^2 \cdot \sigma^2)}. \tag{A.5}$$

Proof. For any $r > 0$, we have

$$\begin{aligned}
E \left[e^{(2\pi r / \sigma^2) \langle z, v \rangle} \right] &= \sum_{z \in \Lambda} \Pr(z) e^{(2\pi r / \sigma^2) \langle z, v \rangle}, \\
&= \left(\sum_{y \in \Lambda} e^{(-\pi \|y\|^2 / \sigma^2)} \right)^{-1} \\
&\quad \cdot \sum_{z \in \Lambda} e^{(-\pi \|z\|^2 / \sigma^2)} \cdot e^{(2\pi / \sigma^2) \langle z, r \cdot v \rangle} \\
&= \left(\sum_{y \in \Lambda} e^{(-\pi \|y\|^2 / \sigma^2)} \right)^{-1} \\
&\quad \cdot \sum_{z \in \Lambda} e^{-\pi (\|z - r \cdot v\|^2 / \sigma^2)} \cdot e^{(\pi r^2 \|v\|^2 / \sigma^2)} \\
&= \frac{\rho_{\sigma, r \cdot v}(\Lambda)}{\rho_{\sigma}(\Lambda)} \cdot e^{(\pi r^2 \|v\|^2 / \sigma^2)} \\
&\leq e^{(\pi r^2 \|v\|^2 / \sigma^2)},
\end{aligned} \tag{A.6}$$

where the last inequality has used the fact that $r \cdot v \in H$ and Lemma 2.9 of [31]. Therefore, by applying Markov's inequality, we get

$$\begin{aligned}
\Pr[\langle z, v \rangle > t] &= \Pr \left[e^{2\pi r / \sigma^2 \langle z, v \rangle} > e^{(2\pi r t / \sigma^2)} \right] \\
&\leq \frac{E \left[e^{(2\pi r / \sigma^2) \langle z, v \rangle} \right]}{e^{(2\pi r t / \sigma^2)}} \leq e^{-(2\pi r t / \sigma^2) + (\pi r^2 \|v\|^2 / \sigma^2)}.
\end{aligned} \tag{A.7}$$

Taking $r = t / \|v\|^2$, we get $\Pr[\langle z, v \rangle > t] \leq e^{-(\pi t^2 / \sigma^2 \|v\|^2)}$. Then, applying the union bound gives us the required result.

The last lemma will be instrumental in bounding the success probability of our rejection sampling algorithm. \square

Lemma 10. For any lattice $\Lambda \subseteq H$ and $v \in \Lambda$, if $\sigma = \omega(\|v\| \cdot \sqrt{\log n})$, then there exists an absolute constant M such that

$$\Pr_{z \leftarrow D_{\Lambda, \sigma}} \left[\frac{D_{\Lambda, \sigma}(z)}{D_{\Lambda, \sigma, v}(z)} < M \right] \geq 1 - 2^{\omega'(\log n)}. \tag{A.8}$$

Proof. By definition, for any $z \in \Lambda$, we have $(D_{\Lambda, \sigma}(z) / D_{\Lambda, \sigma, v}(z)) = (\rho_{\sigma}(z) / \rho_{\sigma, v}(z))$, where we have used that $\rho_{\sigma}(\Lambda) = \rho_{\sigma, v}(\Lambda)$ for any $v \in \Lambda$. Therefore, we can deduce that

$$\frac{D_{\Lambda, \sigma}(z)}{D_{\Lambda, \sigma, v}(z)} = \frac{e^{-\pi (\|z\|^2 / \sigma^2)}}{e^{-\pi (\|z - v\|^2 / \sigma^2)}} = e^{(\pi / \sigma^2) (\|v\|^2 - 2 \langle z, v \rangle)}. \tag{A.9}$$

By using Lemma 9 with $t = \omega(\sqrt{\log n} / 2\pi) \cdot \|v\| \cdot \sigma$, we get

$$\begin{aligned}
e^{\pi/\sigma^2 (\|v\|^2 - 2\langle z, v \rangle)} &< e^{\pi/\sigma^2 (\|v\|^2 + 2\omega((\sqrt{\log n}/2\pi) \cdot \|v\| \cdot \sigma))} \\
&= e^{1+(\pi/\omega(\log n))} = O(1),
\end{aligned} \tag{A.10}$$

with probability at least $1 - 2e^{-(1/4\pi)\omega(\log n)} = 1 - 2^{-\omega'(\log n)}$. We conclude the desired result. \square

Proof of Theorem 2. We can let the set V in Lemma 8 be all vectors $v \in \Lambda$ of length at most T , the function f be $D_{\Lambda, \sigma}$, and the functions g_v be $D_{\Lambda, \sigma, v}$. Lemma 10 implies that there is an absolute constant M , which satisfies the requirements of Lemma 8. We get the result we need. \square

Data Availability

No data were used to support this study. Any lemma or theorem cited in this paper can be obtained openly according to the reference.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was funded by the National Cryptography Development Fund (grant no. MMJJ20180210) and National Natural Science Foundation of China (grant nos. 61832012 and 61672019).

References

- [1] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.
- [2] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, p. 1.
- [3] Z. Cai, X. Zheng, and J. Yu, "A differential-private framework for urban traffic flows estimation via taxi companies," *IEEE Transactions on Industrial Informatics*, p. 1, 2019.
- [4] X. Zheng, Z. Cai, and Y. Li, "Data linkage in smart internet of things systems: a consideration from a privacy perspective," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 55–61, 2018.
- [5] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology*, G. R. Blakley and D. Chaum, Eds., Springer, Berlin, Germany, pp. 47–53, 1985.
- [6] S. Agrawal, D. Boneh, and X. Boyen, "Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE," in *Proceedings of the 30th Annual Conference on Advances in Cryptology. CRYPTO'10*, pp. 98–115, Springer-Verlag, Santa Barbara, CA, USA, August 2010, <http://dl.acm.org/citation.cfm?id=1881412.1881420>.
- [7] D. Boneh, X. Boyen, and E. J. Goh, "Hierarchical identity based encryption with constant size ciphertext," in *Advances in Cryptology—EUROCRYPT 2005*, R. Cramer, Ed., Springer, Berlin, Germany, pp. 440–456, 2005.
- [8] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Advances in Cryptology—CRYPTO 2001*, J. Kilian, Ed., pp. 213–229, Springer, Berlin, Germany, 2001.
- [9] D. Cash, D. Hofheinz, E. Kiltz, and C. Peikert, "Bonsai trees, or how to delegate a lattice basis," *Journal of Cryptology*, vol. 25, no. 4, pp. 601–639, 2012.
- [10] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proceedings of the fortieth annual ACM symposium on Theory of computing—STOC 08*, pp. 197–206, ACM, Victoria, Canada, May 2008.
- [11] B. Waters, "Efficient identity-based encryption without random oracles," in *Advances in Cryptology—EUROCRYPT 2005*, R. Cramer, Ed., pp. 114–127, Springer, Berlin, Germany, 2005.
- [12] B. Waters, "Dual system encryption: realizing fully secure ibe and hibe under simple assumptions," in *Advances in Cryptology—CRYPTO 2009*, S. Halevi, Ed., pp. 619–636, Springer, Berlin, Germany, 2009.
- [13] P. S. L. M. Barreto, B. Libert, N. McCullagh, and J. J. Quisquater, "Efficient and provably-secure identity-based signatures and signcryption from bilinear maps," in *Advances in Cryptology—ASIACRYPT 2005*, B. Roy, Ed., Springer, Berlin, Germany, pp. 515–532, 2005.
- [14] F. Hess, "Efficient identity based signature schemes based on pairings," in *Selected Areas in Cryptography*, K. Nyberg and H. Heys, Eds., Springer, Berlin, Germany, pp. 310–324, 2003.
- [15] K. G. Paterson and J. C. N. Schuldt, "Efficient identity-based signatures secure in the standard model," in *Information Security and Privacy*, L. M. Batten and R. Safavi-Naini, Eds., Springer, Berlin, Germany, pp. 207–222, 2006.
- [16] M. Rückert, "Strongly unforgeable signatures and hierarchical identity-based signatures from lattices without random oracles," in *Post-quantum Cryptography*, N. Sendrier, Ed., Springer, Berlin, Germany, pp. 182–200, 2010.
- [17] J. Xie, Y.-P. Hu, J.-T. Gao, and W. Gao, "Efficient identity-based signature over ntru lattice," *Frontiers of Information Technology & Electronic Engineering*, vol. 17, no. 2, pp. 135–142, 2016.
- [18] X. Boyen, "Multipurpose identity-based signcryption," in *Advances in Cryptology—CRYPTO 2003*, D. Boneh, Ed., Springer, Berlin, Germany, pp. 383–399, 2003.
- [19] L. Chen and J. Malone-Lee, "Improved identity-based signcryption," in *Public Key Cryptography—PKC 2005*, S. Vaudenay, Ed., pp. 362–379, Springer, Berlin, Germany, 2005.
- [20] J. Hoffstein, J. Pipher, J. M. Schanck, J. H. Silverman, W. Whyte, and Z. Zhang, "Choosing parameters for NTRUEncrypt," *Cryptology ePrint Archive*, Report 2015/708, 2015, <https://eprint.iacr.org/2015/708>.
- [21] L. Ducas, V. Lyubashevsky, and T. Prest, "Efficient identity-based encryption over NTRU lattices," in *Advances in Cryptology—ASIACRYPT 2014*, P. Sarkar and T. Iwata, Eds., pp. 22–41, Springer, Berlin, Germany, 2014.
- [22] M. Albrecht, S. Bai, and L. Ducas, "A subfield lattice attack on overstretched NTRU assumptions," in *Advances in Cryptology—CRYPTO 2016*, M. Robshaw and J. Katz, Eds., pp. 153–178, Springer, Berlin, Germany, 2016.
- [23] J. H. Cheon, J. Jeong, and C. Lee, "An algorithm for NTRU problems and cryptanalysis of the GGH multilinear map without a low-level encoding of zero," *LMS Journal of Computation and Mathematics*, vol. 19, no. A, pp. 255–266, 2016.
- [24] P. Kirchner and P.-A. Fouque, "Revisiting lattice attacks on overstretched NTRU parameters," in *Advances in*

- Cryptology—EUROCRYPT 2017*, J.S. Coron and J.B. Nielsen, Eds., pp. 3–26, Springer International Publishing, Cham, Switzerland, 2017.
- [25] D. Stehlé and R. Steinfeld, “Making ntruencrypt and ntrusign as secure as standard worst-case problems over ideal lattices,” *Cryptology ePrint Archive*, Report 2013/004, 2013, <https://eprint.iacr.org/2013/004>.
- [26] Y. Wang and M. Wang, “Crpsf and NTRU signatures over cyclotomic fields,” *Cryptology ePrint Archive*, Report 2018/445, 2018, <https://eprint.iacr.org/2018/445>.
- [27] V. Lyubashevsky, “Lattice signatures without trapdoors,” in *Advances in Cryptology—EUROCRYPT 2012*, D. Pointcheval and T. Johansson, Eds., pp. 738–755, Springer, Berlin, Germany, 2012.
- [28] Y. Wang and M. Wang, “Provably secure NTRUEncrypt over any cyclotomic field,” in *Selected Areas in Cryptography—SAC 2018*, C. Cid and M. J. Jacobson Jr., Eds., pp. 391–417, Springer International Publishing, Cham, Switzerland, 2019.
- [29] M. Rosca, D. Stehlé, and A. Wallet, “On the ring-LWE and polynomial-LWE problems,” in *Advances in Cryptology—EUROCRYPT 2018*, J. B. Nielsen and V. Rijmen, Eds., pp. 146–173, Springer International Publishing, Cham, Switzerland, 2018.
- [30] C. Peikert, “An efficient and parallel Gaussian sampler for lattices,” in *Advances in Cryptology—CRYPTO 2010*, T. Rabin, Ed., pp. 80–97, Springer, Berlin, Germany, 2010.
- [31] D. Micciancio and O. Regev, “Worst-case to average-case reductions based on gaussian measures,” *SIAM Journal on Computing*, vol. 37, no. 1, pp. 267–302, 2007.
- [32] A. Langlois and D. Stehlé, “Worst-case to average-case reductions for module lattices,” *Designs, Codes and Cryptography*, vol. 75, no. 3, pp. 565–599, 2015.
- [33] C. Peikert, O. Regev, and N. Stephens-Davidowitz, “Pseudorandomness of ring-lwe for any ring and modulus,” in *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing—STOC 2017*, pp. 461–473, ACM, Montreal, Canada, June 2017.
- [34] V. Lyubashevsky, C. Peikert, and O. Regev, “A toolkit for ring-LWE cryptography,” in *Advances in Cryptology—EUROCRYPT 2013*, T. Johansson and P. Q. Nguyen, Eds., pp. 35–54, Springer, Berlin, Germany, 2013.
- [35] L. Ducas and A. Durmus, “Ring-LWE in polynomial rings,” in *Public Key Cryptography—PKC 2012*, M. Fischlin, J. Buchmann, and M. Manulis, Eds., pp. 34–51, Springer, Berlin, Germany, 2012.
- [36] Y. Yu, G. Xu, and X. Wang, “Provably secure ntruencrypt over more general cyclotomic rings,” *Cryptology ePrint Archive*, Report 2017/304, 2017, <https://eprint.iacr.org/2017/304>.

Research Article

Two Secure Privacy-Preserving Data Aggregation Schemes for IoT

Yuwen Pu,^{1,2} Jin Luo,^{1,2} Chunqiang Hu ,^{1,2} Jiguo Yu ,³ Ruifeng Zhao,⁴ Hongyu Huang,⁵ and Tao Xiang⁵

¹School of Big Data & Software Engineering, Chongqing University, Chongqing 400044, China

²The Key Laboratory of Dependable Service Computing in Cyber Physical Society, Ministry of Education (Chongqing University), Chongqing, China

³School of Computer Science and Technology, Qilu University of Technology, Jinan, Shandong, China

⁴Electric Power Dispatching and Control Center of Guangdong Power Grid Co., Ltd., Guangzhou, China

⁵College of Computer Science, Chongqing University, Chongqing, China

Correspondence should be addressed to Chunqiang Hu; chu@cqu.edu.cn

Received 29 March 2019; Accepted 29 August 2019; Published 17 September 2019

Guest Editor: Tao Chen

Copyright © 2019 Yuwen Pu 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 next generation of information and communication infrastructure, Internet of Things (IoT) enables many advanced applications such as smart healthcare, smart grid, smart home, and so on, which provide the most flexibility and convenience in our daily life. However, pervasive security and privacy issues are also increasing in IoT. For instance, an attacker can get health condition of a patient via analyzing real-time records in a smart healthcare application. Therefore, it is very important for users to protect their private data. In this paper, we present two efficient data aggregation schemes to preserve private data of customers. In the first scheme, each IoT device slices its actual data randomly, keeps one piece to itself, and sends the remaining pieces to other devices which are in the same group via symmetric encryption. Then, each IoT device adds the received pieces and the held piece together to get an immediate result, which is sent to the aggregator after the computation. Moreover, homomorphic encryption and AES encryption are employed to guarantee secure communication. In the second scheme, the slicing strategy is also employed. Noise data are introduced to prevent the exchanged actual data of devices from disclosure when the devices blend data each other. AES encryption is also employed to guarantee secure communication between devices and aggregator, compared to homomorphic encryption, which has significantly less computational cost. Analysis shows that integrity and confidentiality of IoT devices' data can be guaranteed in our schemes. Both schemes can resist external attack, internal attack, colluding attack, and so on.

1. Introduction

As the important component of the new generation of information technology, Internet of Things (IoT) connects the physical world and information society. It is usually composed of a large number of various sensors and servers. The former is responsible for collecting data, and the latter is responsible for processing data, storing data, and maintaining situational knowledge of the whole system, thus making better decisions [1–3]. In recent decades, with the development of hardware and network, more and more IoT applications are emerging continuously, which bring us unprecedented accuracy, efficiency, and economic benefit. The various IoT applications,

including smart healthcare [4, 5], smart city [6, 7], smart grid [8–10], smart home [11], social network [12–15], smart phone [16], and so on, have different functions and have changed our lifestyle much. For example, in smart healthcare application, many medical sensors which are embedded or attached to the skin of patients collect the real-time health data. Doctors can analyze patients' health condition via monitoring the collected data [17]. In smart phone, a tourist can search for places like restaurants, hotels, scenic spots, and so on by location-based service [18]. In smart home, sensors collect the data of household appliances and report them to management center, which can assist users to know the running state of home appliances [19, 20].

Obviously, IoT applications bring us much convenience and efficiency. However, many security and privacy issues are also brought [21–27]. An adversary can compromise user’s privacy information by eavesdropping the data which are collected by sensors. For example, in smart healthcare application, an adversary is able to monitor a patient’s health condition by accessing to its real-time healthcare data [4, 28]. In smart grid, an adversary can infer user’s behavior and living habits by monitoring the electricity usage data of the user without any other tools [29–31]. In smart phone, an adversary can infer its identity-related information like health status [32, 33] or residence address by eavesdropping user’s location data, which may also reveal the user’s habits. Therefore, the data collected by IoT devices are attractive for adversary. Moreover, as we know, IoT devices are usually computation capability, memory, and power limited, which indicates that encryption algorithms with high computation are not suitable. Thus, how to protect user’s privacy information effectively in IoT network by a lightweight way has attracted much attention of many researchers, and thus many related schemes have been proposed. Among them, there are a number of schemes utilizing data aggregation to achieve privacy preservation [34–37]. Unfortunately, most of them either can only protect privacy of a single side or cause disclosure of intermediate results or are vulnerable to collusion attacks. Hence, it is a challenge to design a novel data aggregation protocol which has low computational cost and can overcome the aforementioned weakness.

In this paper, we mainly propose two secure and privacy-preserving data aggregation schemes for IoT devices. Both of them can prevent user’s privacy data disclosure, thus protect users’ private information from revealing. However, *Scheme-I* achieves private-preserving goal by employing homomorphic encryption and AES encryption, and *Scheme-II* achieves it by employing noise technology to reduce the computation of IoT devices and improve efficiency.

The remainder of this paper is organized as follows: In Section 2, we introduce the related works. In Section 3, we present our system model, security requirements, and our design goals. In Section 4, we recall homomorphic encryption. Then, we present our two schemes in Section 5, which is followed by security analysis, performance evaluation and comparison between two schemes in Sections 6 and 7, respectively. Finally, we draw our conclusions in Section 8.

2. Related Works

Privacy issues of IoT have attracted attention of researchers and many schemes have been proposed. In this section, some state-of-the-art privacy-preserving data aggregation schemes are listed.

Lu et al. presented a lightweight privacy-preserving data aggregation scheme which can not only aggregate hybrid IoT devices’ data into one but also filter injected false data at the network edge by employing homomorphic Paillier encryption, Chinese remainder theorem, and one-way hash chain technique [38]. Alghamdi et al. proposed a novel method which encrypts the devices’ data by employing

elliptic-curve-based seed exchange algorithm and Hilbert-curve-based data transformation. Even if an attacker eavesdrops the transmitted message, he cannot infer the real data [39]. He et al. proposed two data additive aggregation schemes. One scheme achieves private data aggregation by leveraging clustering protocol, and another scheme achieves private data aggregation based on slicing technique and the associative property of addition [40]. Gosman et al. proposed a privacy-preserving aggregation based on symmetric cryptography for smart transportation system [41]. Li et al. proposed an efficient privacy-preserving demand aggregation (EPPDA) scheme by using homomorphic encryption to preserve users’ privacy data for smart grids [42]. Karamitsios and Orphanoudakis proposed an efficient data aggregation for the medical data which are collected real-time by medical sensors for smart healthcare application [43].

Data aggregation has been used in many fields of IoT to achieve privacy preservation. However, most of the existing data aggregation schemes are not truly reliable. In this paper, we propose two efficient and practical data aggregation schemes in which the collected data of devices are blended before reported. Therefore, neither aggregator nor server can infer the actual data of devices.

3. System Model, Security Requirements, and Design Goals

In this section, we formalize the system model, security requirements, and identify our design goals.

3.1. System Model. We consider the architecture in Figure 1 as the basis of our following discussion. Figure 1 reproduced from Hu et al. [44]. There are three entities, including server, aggregator, and devices in our system model of the proposed schemes. We mainly focus on how to report the collected data of IoT devices to the server in an efficient and privacy-preserving way. A two-level gateway topology in IoT is presented as shown in Figure 1. We assume that the server covers m aggregators and that each aggregator covers n IoT devices.

3.1.1. Server. Server is a trustable and powerful entity which provides space for IoT devices to store the collected data that can be retrieved by the users. Furthermore, it will also process and analyze the data to manage IoT applications and keep them operating smoothly.

3.1.2. Aggregator. The aggregator is an honest but curious entity, whose duty is aggregation and relaying. The responsibility of aggregator is to aggregate the received data from IoT devices into an integrated one, whereas the responsibility of relaying is to transmit the aggregation result to the server.

3.1.3. Device. Every IoT device, namely, a sensor, a smart meter, or an RFID reader, collects data, and preprocesses them. For the sake of simplicity, the IoT devices will be abbreviated as devices. We assume that the devices are honest but curious with some computational and storage

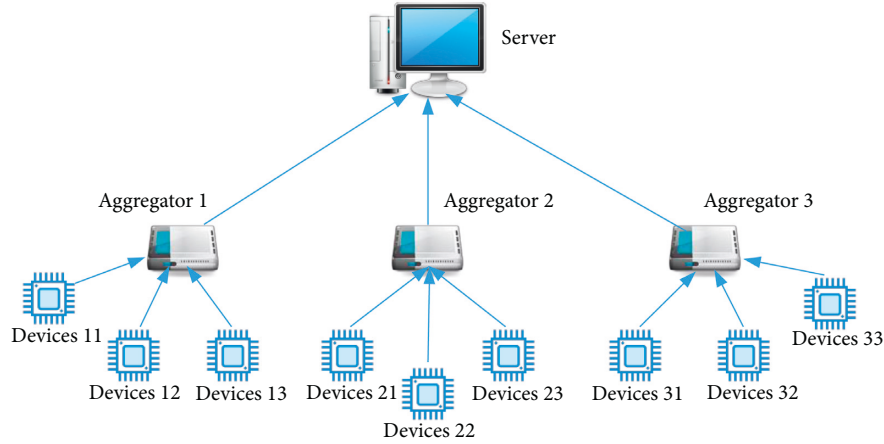


FIGURE 1: Data aggregation model in IoT (Hu et al. [44]).

capability. They keep the system running smoothly but try to infer other devices' collected data.

3.2. Attacker Model and Security Requirements. In our attack model, we consider the following three attacks in IoT system.

3.2.1. External Attack. An adversary may eavesdrop or modify the message which is transmitted between the device and aggregator. Moreover, it may also compromise the aggregator to obtain the privacy information of all devices.

3.2.2. Internal Attack. Aggregator may be curious about the privacy information of all devices and try to infer the actual data of each device, which may compromise the devices' privacy.

3.2.3. Collusion Attack. Some devices may be curious about others' data and try to infer others' privacy information via collusion activity.

In our system, the following security requirements should be achieved.

3.2.4. Privacy Preservation. An adversary cannot obtain devices' data during system communications and operations. Even if several devices collude with each other, they cannot infer other devices' privacy data.

3.2.5. Authentication. Aggregator should guarantee that the received data are valid and derived from legal entities.

3.2.6. Data Integrity. When an adversary forges or modifies a report, the malicious operations should be detected by aggregator.

3.3. Design Goals. According to the system model and security requirements, our design goal concentrates on

proposing two secure, efficient, flexible, and privacy-preserving data aggregation schemes. Specifically, the following design goals are to be achieved.

- (i) **Security:** the proposed schemes should meet all the security requirements as mentioned above.
- (ii) **Efficiency:** the proposed schemes should consider computation efficiency. In other words, the system should support real-time disposal and transmission of data from hundreds and thousands of devices.
- (iii) **Flexibility:** the proposed schemes support "plug and play." Besides, it should be convenient for system to add a new device in the IoT applications.

4. Preliminaries

Homomorphic encryption [45] allows certain computation over encrypted data. Paillier cryptosystem [46] is a popular homomorphic encryption scheme that provides fast encryption and decryption, which is a probabilistic asymmetric algorithm based on the decisional composite residuosity problem. It is adopted by the secure scalar product, which has been widely used in privacy-preserving data mining. The Paillier cryptosystem is briefly introduced as follows:

4.1. Key Generation

- (i) Choose two large prime numbers p and q randomly and independently of each other such that $\gcd(pq, (p-1)(q-1)) = 1$. This property is assured if both primes are of equal length.
- (ii) Compute $n = pq$ and $\lambda = \text{lcm}(p-1, q-1)$.
- (iii) Choose random integer g where $g \in \mathbb{Z}_n^*$.
- (iv) Ensure n divides the order of g by checking the existence of the following modular multiplicative inverse: $\mu = (L(g^\lambda \bmod n^2))^{-1} \bmod n$, where function L is defined as $L(x) = (x-1)/n$.
- (v) The public key is (n, g) .
- (vi) The private key is (λ, μ) .

4.2. *Encryption.* Given a plaintext m where $0 \leq m < n$, select random r where $0 \leq r < n$, and calculate the ciphertext as $c = g^m \cdot r^n \text{ mod } n^2$.

4.3. *Decryption.* Given a ciphertext c , calculate the plaintext as $m = L(c^\lambda \text{ mod } n^2) \cdot \mu \text{ mod } n$.

4.4. Homomorphic Addition of Ciphertexts

- (i) We assume that there are two messages m_1 and m_2 . We can encrypt them with the public key independently and obtain ciphertexts c_1 and c_2 , which are denoted as following: $c_1 = g_1^{m_1} \cdot r_1^n \text{ mod } n^2$ and $c_2 = g_2^{m_2} \cdot r_2^n \text{ mod } n^2$.
- (ii) We can calculate the product of c_1 and c_2 and obtain the result $E(m_1) \cdot E(m_2) = c_1 \cdot c_2 = (g_1^{m_1} \cdot r_1^n \text{ mod } n^2) \cdot (g_2^{m_2} \cdot r_2^n \text{ mod } n^2) = g^{m_1+m_2} \cdot (r_1 r_2)^n \text{ mod } n^2 = E(m_1 + m_2)$. Hence, the sum of plaintext can be calculated from multiplication of the ciphertext.

5. Our Schemes

In this section, two novel data aggregation schemes are introduced. In the proposed schemes, aggregator and server can obtain all of the collected data without knowing the actual data of each device. Besides, the curious and collusive IoT devices cannot infer other devices' private data either. We assume that the IoT devices have some computing power and storage. All IoT devices in the same residential area can be treated as one group. Each aggregator manages a group of IoT devices. Each device has a unique identification ID which is only known by itself and aggregator.

5.1. *Scheme-I.* In *Scheme-I*, Advanced Encryption Standard (AES) symmetrical encryption and homomorphic encryption are employed to protect the transmitted data from leaking during communication. Moreover, hash chain technique is also proposed to achieve one-time pad. The scheme consists of the following three stages: (i) Key generation, (ii) data division and confusion, and (iii) reporting and aggregation.

5.1.1. *Key Generation.* Before the IoT system starts to work, a series of keys and parameters ought to be distributed. The server will generate a private key sk and a public key pk , with the latter to be published. For each group, a group key K_i is generated and broadcasted to all group members, along with a parameter τ for the K_i updating.

5.1.2. *Data Division and Confusion.* In this step, devices segment their data and swap the data pieces pairwise. We assume a topical residential area which comprises an aggregator connected with a large number of devices $Dv = \{Dv_1, Dv_2, \dots, Dv_n\}$. The devices collect data $M = \{M_1, M_2, \dots, M_n\}$, respectively, in a certain period.

In the first place, each device Dv_i slices the data $M_i (i \in (1, 2, \dots, n))$ into n pieces $S_{ij} (i \in (1, 2, \dots, n), j \in (1, 2, \dots, n))$ randomly, where n is the amount of devices in the group. Namely,

$$\begin{cases} M_1 = \sum_{j=1}^n S_{1j}, \\ M_2 = \sum_{j=1}^n S_{2j}, \\ \dots, \\ M_n = \sum_{j=1}^n S_{nj}. \end{cases} \quad (1)$$

Secondly, they exchange the pieces with each other and finally obtain obfuscated data. The piece S_{ii} is preserved by Dv_i while the others are dispatched. Assuming that a device Dv_i wants to transmit $n - 1$ pieces of data $S_{ij} (j \neq i)$ to others, it will conduct a hash operation on them with real time T , denoted as $ch = H(S_{ij} \| T)$. Then, it encrypts $S_{ij} (j \neq i)$, T , and ch via AES, denoted as $c = E_{K_i}(S_{ij} \| T \| ch)$. The ciphertext c can be sent out.

Finally, when receiving the ciphertext c , the device decrypts it and obtains the data slice S_{ij} , the real time T , and hash value ch . T and ch will be utilized to verify whether the message has been manipulated or replayed. If the verification is passed, S_{ij} will be accepted or otherwise be discarded. All received slices and the preserved piece are added up, which is the obfuscated data.

$$\begin{cases} M'_1 = S_{11} + \sum_{i=1}^n S_{i1}, & (i \neq 1), \\ M'_2 = S_{22} + \sum_{i=1}^n S_{i2}, & (i \neq 2), \\ \dots, \\ M'_n = S_{nn} + \sum_{i=1}^n S_{in}, & (i \neq n). \end{cases} \quad (2)$$

It is worth mentioning that the keys used for device-to-device communication are updated continuously. Hash chain technique is employed for this one-time pad. Assuming that an initial key is K_1 , the subsequent secret keys K_n are shown as follows:

$$\begin{cases} K_2 = H(\tau \| K_1), \\ K_3 = H(\tau \| K_2), \\ \dots, \\ K_n = H(\tau \| K_{n-1}). \end{cases} \quad (3)$$

Table 1 shows the result data of each device after this step. $M_i (i \in (1, 2, \dots, n))$ is the actual data of devices $Dv_i (i \in (1, 2, \dots, n))$, and $M'_i (i \in (1, 2, \dots, n))$ is the blended data of $Dv_i (i \in (1, 2, \dots, n))$ after these operations as above. In this way, the actual data of all devices have been covered. Meantime, the sum of devices' data does not change. Namely, $\sum_{i=1}^n M_i = \sum_{i=1}^n M'_i$. Therefore, aggregator can obtain the correct data and not disclose the actual data of each device.

TABLE 1: The result after data division and blending in *Scheme-I*.

	Dv_1	Dv_2	...	Dv_i	...	Dv_n	Actual data
Dv_1	S_{11}	S_{12}	...	S_{1i}	...	S_{1n}	M_1
Dv_2	S_{21}	S_{22}	...	S_{2i}	...	S_{2n}	M_2
...
Dv_i	S_{i1}	S_{i2}	...	S_{ii}	...	S_{in}	M_i
...
Dv_n	S_{n1}	S_{n2}	...	S_{ni}	...	S_{nn}	M_n
Blended data	M'_1	M'_2	...	M'_i	...	M'_n	

5.1.3. Reporting and Aggregation. After devices' exchanging partial collected data with each other as mentioned above, the actual data have been blended. All the blended data $M'_i (i \in (1, 2, \dots, n))$ will be encrypted with pk which is the public key of server before transmitted, which can be denoted as $C_i = E_{pk}(M'_i) (i \in (1, 2, \dots, n))$. Moreover, devices will also compute the hash value of the identification ID, real time T , and preceding ciphertext C_i denoted as $h = H(\text{ID} \| T \| C_i)$ to assist the aggregator to check whether this message has been manipulated or replayed or not. Finally, devices will report the ciphertext C_i , real time T , and hash value h to the aggregator.

When the aggregator obtains the ciphertext C_i , real time T , and the hash value h from devices, it verifies the message based on T and h . If the verification succeeds, the aggregator will aggregate ciphertext C_i together to get immediate result denoted as $C = \prod_{i=1}^n C_i$ and transmit C to the server. The server will decrypt it with the private key sk and obtain the total data Tol of a residential area, which can be denoted as $\text{Tol} = D_{sk}(C)$. During these procedures, both the server and aggregator do not know the actual data of each device.

5.2. Scheme-II. Considering the calculative capability of IoT devices, we also propose another more efficient data aggregation scheme. In this scheme, not only slicing technology but also noise data are introduced to assist devices to blend the actual data. In communication between devices and aggregator, Advanced Encryption Standard (AES) symmetrical encryption rather than homomorphic encryption is employed in order to reduce computational cost. Similarly, the scheme also consists of the following three stages: (i) Key generation, (ii) data division and confusion, and (iii) reporting and aggregation.

5.2.1. Key Generation. A pair of asymmetric key (k_{pu}, k_{pr}) will be generated by the aggregator and k_{pu} will be published. When a device Dv_i is deployed, it generates a symmetric key k_i and a parameter μ_i which are used to update k_i based on hash chain technology. That is, $k_{i_n} = H(\mu_i \| k_{i_{n-1}})$. Then, Dv_i will send them to the corresponding aggregator via the aggregator's public key. Now, the device Dv_i can communicate with the aggregator securely via symmetric key.

5.2.2. Data Division and Confusion. We assume that there are a large number of IoT devices $Dv = \{Dv_1, Dv_2, Dv_3, \dots, Dv_n\}$ which are connected with the same aggregator. These devices collect data $D = \{D_1, D_2, \dots, D_n\}$,

respectively, in a certain period. Each device will slice its collected data into n pieces, which is the same as that in *Scheme-I*.

$$\begin{cases} D_1 = \sum_{j=1}^n S_{1j}, \\ D_2 = \sum_{j=1}^n S_{2j}, \\ \dots, \\ D_n = \sum_{j=1}^n S_{nj}. \end{cases} \quad (4)$$

Afterwards, the devices swap their data. In order to protect the actual data from disclosing, each device $Dv_i (i \in (1, 2, \dots, n))$ will generate n pieces of noise data $N_{ij} (i \in (1, 2, \dots, n), j \in (1, 2, \dots, n))$. These noise data are added to the data pieces and ought to meet the condition below:

$$\begin{cases} \sum_{j=1}^n N_{1j} = 0, \\ \sum_{j=1}^n N_{2j} = 0, \\ \dots, \\ \sum_{j=1}^n N_{nj} = 0. \end{cases} \quad (5)$$

Specifically, we have

$$\begin{cases} R_1 = \sum_{j=1}^n (S_{1j} + N_{1j}), \\ R_2 = \sum_{j=1}^n (S_{2j} + N_{2j}), \\ \dots, \\ R_n = \sum_{j=1}^n (S_{nj} + N_{nj}). \end{cases} \quad (6)$$

Each device Dv_i only preserves the piece $S_{ii} + N_{ii}$ and sends $S_{ij} + N_{ij} (j \neq i, j \in (1, 2, \dots, n))$ to others. After this process, all data are covered. The obfuscated data are shown below:

$$\begin{cases} R'_1 = (S_{11} + N_{11}) + \sum_{i=1}^n (S_{i1} + N_{i1}), & (i \neq 1), \\ R'_2 = (S_{22} + N_{22}) + \sum_{i=1}^n (S_{i2} + N_{i2}), & (i \neq 2), \\ \dots, \\ R'_n = (S_{nn} + N_{nn}) + \sum_{i=1}^n (S_{in} + N_{in}), & (i \neq n). \end{cases} \quad (7)$$

Table 2 shows the immediate result of each device after the stage of data division and confusion. It shows that the actual data of device Dv_i is R_i (because the sum of noise data which is generated by each device equals zero). However,

after blending, the immediate result of the device Dv_i will be R'_i which is different from R_i , so it is successful to conceal the actual data of the device. Moreover, the sum of R_i ($i \in (1, 2, \dots, n)$) equals that of R'_i ($i \in (1, 2, \dots, n)$), which can be denoted as $\sum_{i=1}^n R_i = \sum_{i=1}^n R'_i$.

5.2.3. Reporting and Aggregation. After finishing these operations as above, each device Dv_i will obtain immediate result R'_i . A hash operation on identification ID, real time T , and R'_i will be done, which can be denoted as $RH = H(ID||T||R'_i)$. Then, the hash value RH, the immediate result R'_i , ID, and T will be encrypted with the symmetric key k_i , which can be denoted as $C_i = E_{k_i}(R'_i||ID||T||RH)$. When obtaining the ciphertext C_i , device will report it to the corresponding aggregator.

After receiving C_i , aggregator will decrypt it with k_i and verify whether this message has been manipulated or replayed by checking hash value RH and identification ID. If that passes, aggregator will aggregate the received data R'_i ($i \in (1, 2, \dots, n)$) to an intermediate result and report it to the server. In this way, both of aggregator and server cannot know the actual data of each device.

6. Security Analysis

In this section, we will analyze the security properties of the two proposed schemes. In particular, our analysis focuses on how the schemes can resist various attacks and achieve privacy preservation.

6.1. Analysis on Scheme-I

6.1.1. Resistance to Eavesdropping Attack

Theorem 1. *An adversary cannot obtain devices' private data by eavesdropping the encrypted data during transmitting.*

Proof. All device's data are encrypted with symmetrical encryption or asymmetric encryption before transmitting. Therefore, adversary without the private key cannot decrypt the ciphertext by brute-force with a non-negligible probability. \square

6.1.2. Resistance to Replay Attack

Theorem 2. *If an adversary reports the same message to aggregator or IoT devices, it can be detected.*

Proof. If an adversary A transmits the replayed message M to aggregator or devices, when receiving M , the aggregator or devices will check the hash value of the real time T to verify whether this message is replayed or not. \square

6.1.3. Resistance to Manipulation Attack

Theorem 3. *If an adversary manipulates the message between two IoT devices during communication, it can be detected.*

Proof. We assume that an IoT device DvA transmits message M to another IoT device DvB . M is the ciphertext $E(H(S_i||T)||S_i||T)$ (S_i is the transmitted plaintext data). When DvB receiving M , it will decrypt M to obtain hash value $H(S_i||T)$, T , and S_i . Then, DvB will also do a hash operation on S_i and T and verify whether this message has been manipulated by matching the result with preceding received hash value $H(S_i||T)$. \square

Theorem 4. *If an adversary manipulates the message between the IoT device and aggregator, it can be detected.*

Proof. We assume that an IoT device DvA reports M to the aggregator. M contains the hash value $H(ID||T||C)$ (H is a hash function, ID is A 's unique identity, T is the real time, and C is the ciphertext of blended data), ciphertext C , and T . When receiving M , the aggregator will do a hash operation on DvA 's ID, T , C , and verify whether this message has been manipulated or not. \square

6.1.4. Resistance to Impersonation Attack

Theorem 5. *If an adversary masquerades as another valid device reporting collected data to aggregator, it can be detected.*

Proof. If an adversary A wants to masquerade as another valid device DvB and report message M to aggregator. When receiving M , aggregator will check the identity ID of A . Therefore, if A wants to masquerade as another valid device DvB to report message, it must have the ID of DvB . However, the ID of DvB is only known by DvB and aggregator. Adversary A cannot obtain it with a non-negligible probability. \square

6.1.5. Resistance to Internal Attack

Theorem 6. *We assume that aggregator is an internal attacker which is curious about all devices' privacy data. It still cannot obtain the actual data of all devices.*

Proof. We assume that IoT devices are $Dv = \{Dv_1, Dv_2, \dots, Dv_n\}$ and their reporting message is $E = \{E_1, E_2, \dots, E_n\}$, respectively. E is the ciphertext of blended data which are encrypted with the public key of server, whereas the aggregator does not have the private key to decrypt the ciphertext. \square

6.1.6. Resistance to Colluding Attack

Theorem 7. *Considering the curiosity of devices, some devices may conspire to reveal privacy data of others.*

Proof. We assume that there are n devices, whose collected data are $M = \{M_1, M_2, \dots, M_n\}$, respectively. After slicing

TABLE 2: The result after data division and blending in *Scheme-II*.

	Dv_1	Dv_2	...	Dv_i	...	Dv_n	Actual data
Dv_1	$S_{11} + N_{11}$	$S_{12} + N_{12}$...	$S_{1i} + N_{1i}$...	$S_{1n} + N_{1n}$	R_1
Dv_2	$S_{21} + N_{21}$	$S_{22} + N_{22}$...	$S_{2i} + N_{2i}$...	$S_{2n} + N_{2n}$	R_2
...
Dv_i	$S_{i1} + N_{i1}$	$S_{i2} + N_{i2}$...	$S_{ii} + N_{ii}$...	$S_{in} + N_{in}$	R_i
...
Dv_n	$S_{n1} + N_{n1}$	$S_{n2} + N_{n2}$...	$S_{ni} + N_{ni}$...	$S_{nn} + N_{nn}$	R_n
Blended data	R'_1	R'_2	...	R'_i	...	R'_n	

M ($i \in (1, 2, \dots, n)$) into n pieces, preserving one piece privately, and exchanging the remaining $n - 1$ pieces of data with each other as mentioned above, all devices' actual data have been blended. We also assume that $n - 1$ colluding devices want to infer another device's (Dv_A) information. The randomly divisional data of Dv_A is $M_i = \sum_{j=1}^n S_{ij}$, the preserved private data of Dv_A is S_{ii} , and the blended data of Dv_A is $M'_i = S_{ii} + \sum_{j=1}^n S_{ij}$ ($j \neq i$). The $n - 1$ colluding devices only know the data S_{ij} ($j \neq i$) which Dv_A has exchanged with them, but they do not know the private data S_{ii} which is preserved privately and only known by Dv_A itself. Moreover, they also cannot obtain the value of M'_i because M'_i has been encrypted before transmitting to the aggregator. Therefore, the colluding devices cannot reveal other devices' privacy data. \square

6.2. Analysis on Scheme-II

6.2.1. Resistance to Eavesdropping Attack

Theorem 8. *An adversary cannot obtain devices' private data by eavesdropping the transmitted data.*

Proof. For communication among devices, noise data have been added to all the transmitted data, so the adversary cannot reveal the actual data of devices. For the communication between aggregator and device, the transmitted data have been encrypted with symmetric key. However, the adversary cannot decrypt the ciphertext by brute-force with a non-negligible probability. \square

6.2.2. Resistance to Replay Attack

Theorem 9. *If an adversary reports the same message to aggregator, it can be detected.*

The proof of resistance to replay attack is the same as that in analysis on Scheme-I.

6.2.3. Resistance to Manipulation Attack

Theorem 10. *If an adversary manipulates the message between an IoT device and aggregator, it can be detected.*

Proof. We assume that an IoT device Dv_i reports the ciphertext C_i to the aggregator. The aggregator can decrypt it

with the corresponding key and verify whether the message has been manipulated by checking hash value. \square

6.2.4. Resistance to Impersonation Attack. The proof of resistance to replay attack is the same as that in analysis on *Scheme-I*.

6.2.5. Resistance to Internal Attack

Theorem 11. *We assume that aggregator is an internal attacker which is curious about all devices' privacy data. It still cannot obtain the actual data of each device.*

Proof. The data which is transmitted from device to aggregator is not the actual data of the device. It is the sum of its preserved one private piece data and the remaining $n - 1$ pieces of data from other devices. Hence, aggregator cannot reveal the actual data of each device. \square

6.2.6. Resistance to Colluding Attack

Theorem 12. *Considering the curiosity of devices, some devices may conspire to reveal privacy data of others.*

Proof. We assume that there are n devices Dv_i ($i \in (1, 2, \dots, n)$) which are connected with the same aggregator. The collected data of devices Dv_i are R_i ($i \in (1, 2, \dots, n)$), respectively. We also assume that $n - 1$ colluding devices want to infer the collected data of another device Dv_1 . The colluding devices can only obtain the data pieces which contain the actual data of Dv_1 and noise data, but they cannot infer the actual data of Dv_1 . Therefore, the colluding devices cannot reveal other devices' privacy data. \square

7. Performance Evaluation

In this section, we will evaluate the computational cost of the proposed schemes. Besides, we will also compare the two proposed schemes and analyze their advantages and disadvantages.

7.1. Computation Overhead. It is well known for us that the computational cost of modular exponentiation and multiplication operations is much higher than that of hash functions and addition operations, so we will ignore the cost

TABLE 3: The operations of RDA and a single device.

	Scheme-I		Scheme-II	
	Single IoT device	Aggregator	Single IoT device	Aggregator
Key generation	—	—	—	—
Data division and confusion	$(n-1)(A_e + A_d)$	—	—	—
Reporting and aggregation	$2C_e + C_m + C_o$	$(n-1)C_m$	A_e	nA_d
Total cost	$(n-1)(A_e + A_d) + 2C_e + C_m + C_o$	$(n-1)C_m$	A_e	nA_d

of hash operations and addition operations and only focus on the computational cost incurred by encryption and decryption operations in this study.

We assume a topical residential area which comprises an aggregator connected with a large number of IoT devices $Dv_i = \{Dv_1, Dv_2, \dots, Dv_n\}$. Note that C_e is the computational cost of an exponentiation operation; C_m is the computational cost of a multiplication operation; C_o is the computational cost of a modulo operation, and A_e and A_d are the computational costs of an AES encryption and an AES decryption, respectively.

For one device, in *Scheme-I*, in data division and confusion phase, $n-1$ AES encryption operations, $n-1$ AES decryption operations, and a series of negligible addition operations are required, thus the cost is $(n-1)(A_e + A_d)$. Moreover, two exponentiation operations, one multiplication operation and one modulo operation are required in the reporting and aggregation phase, where the computational cost is $2C_e + C_m + C_o$. Hence, the total computational cost of one device is $(n-1)(A_e + A_d) + 2C_e + C_m + C_o$. In *Scheme-II*, only the negligible addition operations are required for devices in Data division and confusion phase. In the reporting and aggregation phase, only one AES encryption operation is required for one device. Therefore, the computational cost of one device is A_e .

Moving next to the aggregator, in *Scheme-I*, only $n-1$ multiplication operations are executed, thus the total computational cost is $(n-1)C_m$. In *Scheme-II*, there are n AES decryption operations which will be executed, so the total computational cost is nA_d .

Table 3 summarizes the computational complexities of IoT device and aggregator in each phase of *Scheme-I* and that of *Scheme-II*. We conduct the experiments running in Python on a 3.7 GHz-processor 16 GB-memory computing machine on 8 byte data to study the operation costs. The experimental results indicate that an AES encryption operation with 256 bit key almost costs 0.0034 ms, an AES decryption operation with 256 bit key almost costs 0.0038 ms. When the key of Paillier cryptosystem is 256 bit, an encryption operation almost costs 110 ms and an decryption operation almost costs 0.9 ms.

7.2. Comparisons Analysis. Both of the two proposed schemes are efficient and effective to prevent devices' privacy data from revealing. However, they are also different in some respects, which makes them meet some different scenario requirements better. Firstly, *Scheme-I* employs AES to protect the exchanged pieces from leaking, but *Scheme-II* guarantees the privacy data by adding noise data. Moreover,

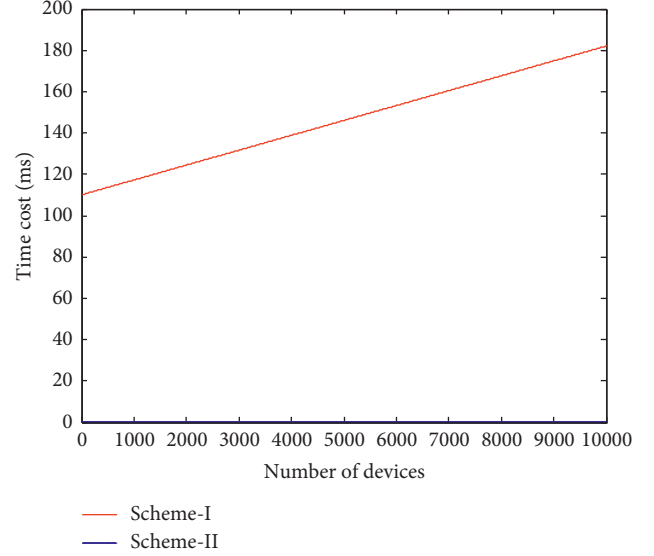


FIGURE 2: Comparison of the computational cost of one IoT device between *Scheme-I* and *Scheme-II*.

Paillier cryptosystem is employed to guarantee the confidentiality and integrity of collected data during communications between IoT device and aggregator in *Scheme-I*, but *Scheme-II* employs AES to reduce computational cost of devices. In *Scheme-I*, the computational cost of one IoT device is $C1 = (n-1) \times 0.0034 + (n-1) \times 0.0038 + 110$ ms. Similarly, the computational cost of one IoT device in *Scheme-II* is $C2 = 0.0034$ ms. It is shown in Figure 2, which indicates that even if there are nearly 10000 IoT devices in an area, the computational cost of one IoT device is not more than 0.2 s in *Scheme-I*, and the computational cost of one IoT device is just 0.0034 ms in *Scheme-II*. Hence, the computational cost for IoT devices is very low in the proposed schemes. Moreover, the total computational costs of IoT devices are also different. It can be denoted as $S1 = n(n-1) \times 0.0034 + n(n-1) \times 0.0038 + n \times 110$ ms and $S2 = n \times 0.0034$ ms, respectively. We depict the variation of total computational costs of the two schemes in terms of device number n in Figure 3. The different value of them can be denoted as $S = S1 - S2$. It is shown in Figure 4. They illustrate that *Scheme-II* is more efficient than *Scheme-I*. Moreover, with the number of devices increasing, *Scheme-II* is more efficient than *Scheme-I*. Nonetheless, because each device has different secret keys to communicate with aggregator in *Scheme-II*, that is, the aggregator has to store n secret key, which may lead to key management issues. When the number of devices is very large in a residential area, it is

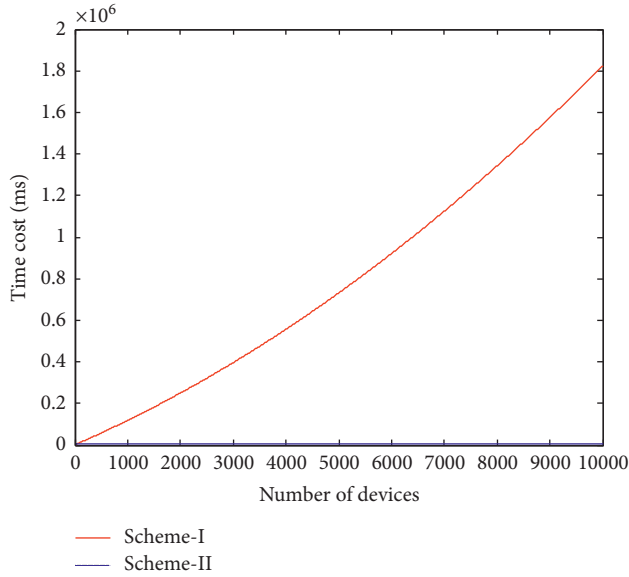


FIGURE 3: Comparison of total computational cost of all IoT devices between *Scheme-I* and *Scheme-II*.

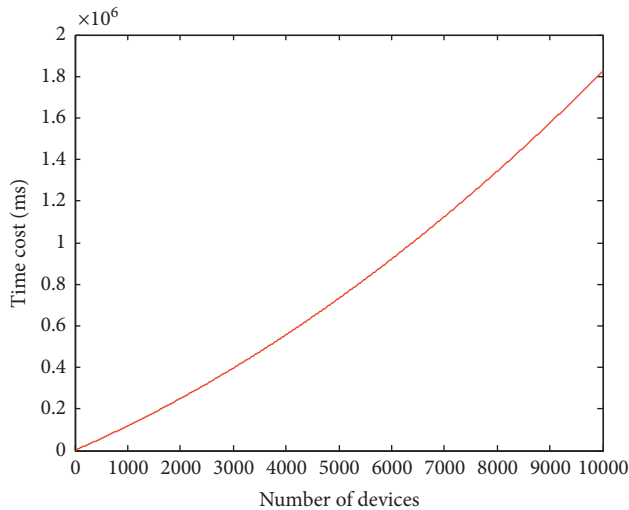


FIGURE 4: The difference value of total computational cost of *Scheme-I* and *Scheme-II*.

difficult to manage so many keys for aggregator. However, aggregator only needs to store a group key and the corresponding parameter in *Scheme-I*.

8. Conclusion

In this paper, two secure and efficient data aggregation schemes are proposed for IoT devices. Both of them support “plug and play” and preserve IoT devices’ private data by blending their data before reported. However, there are also some differences between the two proposed schemes. For *Scheme-I*, AES encryption and Paillier cryptosystem are employed to guarantee the confidentiality and integrity of the collected data. For *Scheme-II*, noise data are introduced to blend the actual data of users rather than encryption

method, which can reduce computational cost of IoT devices and improve communication efficiency significantly. Moreover, we have provided security analysis to demonstrate that our schemes can resist external attack, internal attack, colluding attack, and so on. Meanwhile, we also make a comparison between the proposed schemes to demonstrate their strength and weakness. The result shows that *Scheme-I* is more secure and *Scheme-II* is more efficient. For the future work, we plan to improve the schemes by exploring more efficient and secure encryption method and further deploy them in the real-world IoT applications.

Data Availability

In this paper, we provide the detailed data in Section 6 (Performance Evaluation). Meanwhile, we also introduce the procedure of computational cost analysis. The researchers can verify our experiment results according to our introductions. We listed the key points of the experiment as follows. Key points of data statement: 1. The experiments running in Python on a 3.7 GHz-processor 16 GB-memory computing machine 2. The experimental results indicate an AES encryption operation with 256-bit key and AES decryption operation with 256-bit key 3. The Paillier cryptosystem is 256-bit 4. AES encryption operation costs 0.0034 ms, and AES decryption operation cost 0.0038 ms 5. The encryption operation of Paillier cryptosystem costs 110 ms and an decryption operation almost costs 0.9 ms. The researcher can verify the above experimental results in the same running environment.

Disclosure

The previous work [44] was published in International Conference on Wireless Algorithms, Systems, and Applications 2018.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported partially by the Fundamental Research Funds for the Central Universities (No. 2019CDQYRJ006), National Natural Science Foundation of China (Nos. 61702062, 61672118, 61932006, and U1836114), Science and Technology Project of Guangdong Power Grid Co. Ltd. (GDKJXM20180250), Chongqing Research Program of Basic Research and Frontier Technology (Grant No. cstc2018jcyjAX0334), Key Project of Technology Innovation and Application Development of Chongqing (CSTC2019jcsx-mbdx0151), and Overseas Returnees Innovation and Entrepreneurship Support Program of Chongqing (cx2018015).

References

- [1] C. Cecchini, M. Jimenez, S. Mosser, and M. Riveill, “An architecture to support the collection of big data in the internet of things,” in *Proceedings of the 2014 IEEE World*

- Congress on Services*, pp. 442–449, IEEE, Anchorage, AK, USA, June 2014.
- [2] M. Abu-Elkheir, M. Hayajneh, and N. Ali, “Data management for the internet of things: design primitives and solution,” *Sensors*, vol. 13, no. 11, pp. 15582–15612, 2013.
 - [3] C. Perera, R. Ranjan, L. Wang, S. U. Khan, and A. Y. Zomaya, “Big data privacy in the internet of things era,” *IT Professional*, vol. 17, no. 3, pp. 32–39, 2015.
 - [4] C. Hu, H. Li, Y. Huo, T. Xiang, and X. Liao, “Secure and efficient data communication protocol for wireless body area networks,” *IEEE Transactions on Multi-Scale Computing Systems*, vol. 2, no. 2, pp. 94–107, 2016.
 - [5] T. K. Dasaklis, F. Casino, and C. Patsakis, “Blockchain meets smart health: towards next generation healthcare services,” in *Proceedings of the 2018 9th International Conference on Information, Intelligence, Systems and Applications (IISA)*, pp. 1–8, IEEE, Zakynthos, Greece, July 2018.
 - [6] C. Hu, X. Cheng, J. Yu, Z. Tian, W. Lv, and X. Chen, “Achieving privacy preservation and billing via delayed information release,” *submitted to IEEE/ACM Transactions on Networking*, 2019.
 - [7] A. Alkhamisi, M. S. H. Nazmudeen, and S. M. Buhari, “A cross-layer framework for sensor data aggregation for iot applications in smart cities,” in *Proceedings of the IEEE International Smart Cities Conference (ISC2)*, pp. 1–6, Trento, Italy, September 2016.
 - [8] A. Alrawaiis, A. Althothaily, C. Hu, and X. Cheng, “Fog computing for the internet of things: security and privacy issues,” *IEEE Internet Computing*, vol. 21, no. 2, pp. 34–42, 2017.
 - [9] Z. Cai and X. Zheng, “A private and efficient mechanism for data uploading in smart cyber-physical systems,” *IEEE Transactions on Network Science and Engineering*, p. 1, 2018.
 - [10] C. Hu, H. Liu, L. Ma et al., “A secure and scalable data communication scheme in smart grids,” *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 5816765, 17 pages, 2018.
 - [11] T. Song, R. Li, B. Mei, J. Yu, X. Xing, and X. Cheng, “A privacy preserving communication protocol for iot applications in smart homes,” *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1844–1852, 2017.
 - [12] C. Hu, R. Li, W. Li, J. Yu, Z. Tian, and R. Bie, “Efficient privacy-preserving schemes for dot-product computation in mobile computing,” in *Proceedings of the 1st ACM Workshop on Privacy-Aware Mobile Computing*, pp. 51–59, ACM, Paderborn, Germany, July 2016.
 - [13] Z. Cai, Z. He, X. Guan, and Y. Li, “Collective data-sanitization for preventing sensitive information inference attacks in social networks,” *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.
 - [14] Z. He, Z. Cai, and J. Yu, “Latent-data privacy preserving with customized data utility for social network data,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 665–673, 2018.
 - [15] L. Huang, X. Fan, Y. Huo, C. Hu, Y. Tian, and J. Qian, “A novel cooperative jamming scheme for wireless social networks without known csi,” *IEEE Access*, vol. 5, pp. 26476–26486, 2017.
 - [16] S. Egelman, A. P. Felt, and D. Wagner, “Choice architecture and smartphone privacy: there’s a price for that,” in *The Economics of Information Security and Privacy*, pp. 211–236, Springer, Berlin, Germany, 2013.
 - [17] G. G. Dagher, J. Mohler, M. Milojkovic, and P. B. Marella, “Ancile: privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology,” *Sustainable Cities and Society*, vol. 39, pp. 283–297, 2018.
 - [18] J. C. Sipior, B. T. Ward, and L. Volonino, “Privacy concerns associated with smartphone use,” *Journal of Internet Commerce*, vol. 13, no. 3-4, pp. 177–193, 2014.
 - [19] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, “Blockchain for Iot security and privacy: the case study of a smart home,” in *Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom workshops)*, pp. 618–623, IEEE, Kona, France, March 2017.
 - [20] E. Fernandes, J. Jung, and A. Prakash, “Security analysis of emerging smart home applications,” in *Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP)*, pp. 636–654, IEEE, San Jose, CA, USA, May 2016.
 - [21] Y. Huo, Y. Tian, L. Ma, X. Cheng, and T. Jing, “Jamming strategies for physical layer security,” *IEEE Wireless Communications*, vol. 25, no. 1, pp. 148–153, 2018.
 - [22] X. Zheng, Z. Cai, and Y. Li, “Data linkage in smart iot systems: a consideration from privacy perspective,” *IEEE Communications Magazine*, vol. 10, no. 2, pp. 12–20, 2018.
 - [23] C. Hu, W. Li, X. Cheng, J. Yu, S. Wang, and R. Bie, “A secure and verifiable access control scheme for big data storage in clouds,” *IEEE Transactions on Big Data*, vol. 4, no. 3, pp. 341–355, 2018.
 - [24] Y. Huo, C. Hu, X. Qi, and T. Jing, “LoDPD: a location difference-based proximity detection protocol for fog computing,” *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1117–1124, 2017.
 - [25] C. Hu, N. Zhang, H. Li, X. Cheng, and X. Liao, “Body area network security: a fuzzy attribute-based signcryption scheme,” *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 9, pp. 37–46, 2013.
 - [26] Y. Huo, W. Dong, J. Qian, and T. Jing, “Coalition game-based secure and effective clustering communication in vehicular cyber-physical system (vcps),” *Sensors*, vol. 17, no. 3, p. 475, 2017.
 - [27] Y. Lu, Z. Zhao, B. Zhang, L. Ma, Y. Huo, and G. Jing, “A context-aware budget-constrained targeted advertising system for vehicular networks,” *IEEE Access*, vol. 6, pp. 8704–8713, 2018.
 - [28] S. Sharma, K. Chen, and A. Sheth, “Toward practical privacy-preserving analytics for iot and cloud-based healthcare systems,” *IEEE Internet Computing*, vol. 22, no. 2, pp. 42–51, 2018.
 - [29] Z. Wang, “An identity-based data aggregation protocol for the smart grid,” *IEEE Transactions on Industrial Informatics*, vol. 13, no. 5, pp. 2428–2435, 2017.
 - [30] W. Jia, H. Zhu, Z. Cao, X. Dong, and C. Xiao, “Human-factor-aware privacy-preserving aggregation in smart grid,” *IEEE Systems Journal*, vol. 8, no. 2, pp. 598–607, 2014.
 - [31] C.-I. Fan, S.-Y. Huang, and Y.-L. Lai, “Privacy-enhanced data aggregation scheme against internal attackers in smart grid,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 666–675, 2014.
 - [32] H. Zhu, R. Lu, C. Huang, L. Chen, and H. Li, “An efficient privacy-preserving location-based services query scheme in outsourced cloud,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7729–7739, 2016.
 - [33] B. Niu, X. Zhu, X. Lei, W. Zhang, and H. Li, “Eps: encounter-based privacy-preserving scheme for location-based services,” in *Proceedings of the Global Communications Conference*

- (*GLOBECOM*), 2013, pp. 2139–2144, IEEE, Atlanta, GA, USA, December 2013.
- [34] C. Hu, Y. Huo, L. Ma, H. Liu, S. Deng, and L. Feng, “An attribute-based secure and scalable scheme for data communications in smart grids,” in *Wireless Algorithms, Systems, and Applications (WASA)*, pp. 469–482, Springer, Berlin, Germany, 2017.
- [35] H. Bao and R. Lu, “Comment on privacy-enhanced data aggregation scheme against internal attackers in smart grid,” *IEEE Transactions on Industrial Informatics*, vol. 12, no. 1, pp. 2–5, 2016.
- [36] N. Saputro and K. Akkaya, “Performance evaluation of smart grid data aggregation via homomorphic encryption,” in *Proceedings of the Wireless Communications And Networking Conference (WCNC)*, pp. 2945–2950, IEEE, Paris, France, April 2012.
- [37] Z. Erkin, J. R. Troncoso-Pastoriza, R. L. Lagendijk, and F. Perez-Gonzalez, “Privacy-preserving data aggregation in smart metering systems: an overview,” *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 75–86, 2013.
- [38] R. Lu, K. Heung, A. H. Lashkari, and A. A. Ghorbani, “A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced iot,” *IEEE Access*, vol. 5, pp. 3302–3312, 2017.
- [39] A. Alghamdi, M. Alshamrani, A. Alqahtani, S. S. A. Al Ghamdi, and R. Harrathi, “Secure data aggregation scheme in wireless sensor networks for iot,” in *Proceedings of the 2016 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–5, IEEE, Hammamet, Tunisia, May 2016.
- [40] W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. Abdelzaher, “Pda: privacy-preserving data aggregation in wireless sensor networks,” in *Proceedings of the INFOCOM 2007 26th IEEE International Conference on Computer Communications*, pp. 2045–2053, IEEE, Anchorage, AK, USA, May 2007.
- [41] C. Gosman, C. Dobre, and F. Pop, “Privacy-preserving data aggregation in intelligent transportation systems,” in *Proceedings of the 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pp. 1059–1064, IEEE, Lisbon, Portugal, May 2017.
- [42] H. Li, X. Lin, H. Yang, X. Liang, R. Lu, and X. Shen, “Eppdr: an efficient privacy-preserving demand response scheme with adaptive key evolution in smart grid,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2053–2064, 2014.
- [43] K. Karamitsios and T. Orphanoudakis, “Efficient iot data aggregation for connected health applications,” in *Proceedings of the 2017 IEEE Symposium on Computers and Communications (ISCC)*, pp. 1182–1185, IEEE, Heraklion, Greece, July 2017.
- [44] C. Hu, J. Luo, Y. Pu et al., “An efficient privacy-preserving data aggregation scheme for IoT,” in *Proceedings of the International Conference on Wireless Algorithms, Systems, and Applications*, pp. 164–176, Springer, Tianjin, China, June 2018.
- [45] C. Fontaine and F. Galand, “A survey of homomorphic encryption for nonspecialists,” *EURASIP Journal on Information Security*, vol. 2007, no. 1, pp. 1–10, 2007.
- [46] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in *Advances in Cryptology-EUROCRYPT’99*, pp. 223–238, Springer, Berlin, Germany, 1999.

Research Article

Distributed Link Scheduling Algorithm Based on Successive Interference Cancellation in MIMO Wireless Networks

Junhua Wu, Dandan Lin, Guangshun Li , Yuncui Liu, and Yanmin Yin

School of Information Science and Engineering, Qufu Normal University, Rizhao, China

Correspondence should be addressed to Guangshun Li; 30752585@qq.com

Received 6 March 2019; Accepted 15 May 2019; Published 19 June 2019

Guest Editor: Zaobo He

Copyright © 2019 Junhua Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The performance of multiple input multiple output (MIMO) wireless networks is limited mainly by concurrent interference among sensor nodes. Effective link scheduling algorithms with the technology of successive interference cancellation (SIC) can maximize throughput in MIMO wireless networks. Most previous works on link scheduling in MIMO wireless networks did not consider SIC. In this paper, we propose a MIMO-SIC (MSIC) algorithm under the SINR model. First, a mathematical framework is established for the cross-layer optimization of routing and scheduling, with constraints of traffic balance and link capacity. Second, the interference regions are divided to characterize the level of interference between links. Finally, we propose a distributed link scheduling algorithm based on MSIC to eliminate the interference between competing links in the MIMO network. Experimental results show that the MSIC algorithm can increase the end-to-end throughput per unit by approximately 73% on average compared with non-SIC algorithms.

1. Introduction

Due to the explosiveness of big data, many new high-performance requirements of network throughput, real-time performance, security privacy, and bandwidth have been put forward [1–4]. It is consequently challenging to design efficient link scheduling algorithms to improve communication efficiency in wireless communication. Wireless network multiple input multiple output (MIMO) technology can transmit multiple data streams simultaneously without increasing bandwidth and enhance data throughput; MIMO has therefore attracted increasing attention [5, 6].

MIMO refers to the technology of using multiple transmitting antennas and multiple receiving antennas in wireless transmission, which is a major technology of smart antennas in wireless communication networks. The main idea of MIMO is to combine the signals of the receiving and sending antennas to increase transmission reliability and data throughput [7, 8]. The architecture of MIMO wireless communication networks is shown in Figure 1.

However, concurrent links also generate some problems in communication interference, which reduces the success

probability of communication links [9]. The reason is that, due to transmission interference of adjacent channels, mixed superimposed signals will reach the receiver node [10, 11]. The cumulative interference effect of the link depends not only on itself but also on concurrent links. When conflict occurs between concurrent links, transmissions fail due to bad interference. In wireless networks, MIMO gain is closely related to link scheduling.

The link scheduling problem focuses on the study of capacity optimization and throughput maximization [12] and can be divided into the following two types of problems. One is the maximum independent set link (MISL) problem [13], also known as the capacity maximization problem or single-slot link scheduling problem. In this problem, given a set of communication links, the largest subset of concurrent links that can be transmitted simultaneously in the same time slot must be identified. The other type of problem is the shortest link scheduling (SLS) problem [14, 15], also known as the delay minimization problem. This problem refers to scheduling a given set of links with a minimum of time slots. This paper studies the former problem, namely, designing a

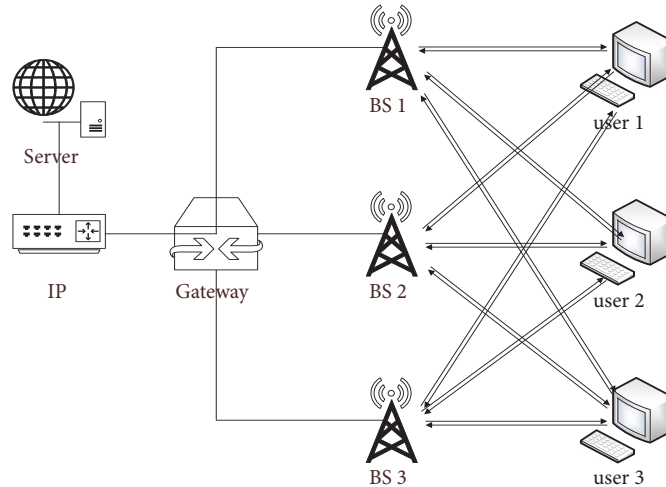


FIGURE 1: The architecture of a MIMO wireless communication network.

link algorithm that schedules as many links as possible in the same time slot.

There are many factors that affect the link scheduling problem, such as different choices of centralized [16] or distributed algorithms [17, 18]. Before the centralized algorithm is executed, various information of the network node is broadcast to the execution node, such as the set of neighbors and the transmit power of the node. As the network scale increases, the time complexity of centralized algorithms will increase dramatically. However, the distributed algorithm only needs to obtain the corresponding information of the neighbors during the execution process. Such information exchange can be achieved only by a broadcast, and the running time of the algorithm is independent of the network scale.

In addition, establishing a reasonable interference model is the key to designing a correct and efficient link scheduling algorithm. The most commonly used interference models can be classified into the protocol interference model [19] and SINR (signal-to-interference-plus-noise ratio) interference model [20, 21]. Under the protocol model, the transmission of a link is deemed successful if no other links within a certain transmission range are active. Therefore, the coexistence relationship between two links is mainly determined by the geometry. Due to its simplicity, the protocol model has been widely used. By contrast, under the SINR model, the coexistence relationship depends on its own channel condition and the level of the aggregated interference. Specifically, a transmission of a link is said to be successful if its SINR value is greater than a predetermined threshold. One challenge under the SINR model is that multiple links can transmit successfully through a common channel, even if they observe some interference signal from each other, in marked contrast to the protocol model. Furthermore, the link relationship is a function of the distance to the neighboring links and their status, which may change over time. Therefore, the link coexistence relationship under the SINR model is “multilateral”

and “dynamic.” As a result, link scheduling under the SINR model is much more complicated. The literature [22] proves the robustness of SINR models with geometric path loss. The SINR model opens a new avenue for more efficient resource allocation.

SIC is an effective physical layer technology for multipacket reception to combat interference, and it allows other concurrent transmission links to decode correctly [20]. According to the descending order of the receivers’ power level, SIC regards the interference signals as a useful signal obeying a specific structure. As SIC is able to resolve the collision, more simultaneous transmission and higher performance could be expected [23, 24]. We therefore focus on link scheduling in MIMO wireless networks with SIC. However, there has been little work on exploring SIC in MIMO networks based on the SINR model.

The main contributions of this paper are as follows. We propose a distributed link scheduling algorithm in MIMO wireless networks with SIC to maximize the link throughput. First, based on the characteristics of MIMO and SIC, we propose a combination of these two technologies to establish the MSIC model, and a mathematical framework is established for cross-layer optimization of scheduling, with constraints of traffic balance and link capacity. Second, due to the global characteristic of the interference, localization of the interference and division of the interference regions are considered. Finally, a distributed link scheduling algorithm based on MSIC to generate a feasible scheduling set is proposed.

The rest of this article is organized as follows: the first and second parts describe the background and current situation. In the third part, the MSIC network model is constructed based on SIC technology, and the cross-layer optimization problem is modeled. In the fourth part, the interference regions are divided, and the detailed process of the distributed link scheduling algorithm based on the MSIC network model is given. The fifth part provides experimental results. The

last part summarizes the paper and puts forward the future development trends.

2. Related Work

The link scheduling problem has been the subject of extensive research. Moscibroda et al. first defined the link scheduling problem and introduced the concept of scheduling complexity under the SINR model [25]. Goussevkaia et al. proved that the link scheduling problem is NP-hard [26]. Dinitz considered single-slot scheduling and gave the first distributed algorithm [27]. Although his goal was to design a distributed algorithm, the technique relies mainly on the machine learning theory of the algorithm. Qian et al. [21] first developed a new “MIMO-pipe” model that captured the rate-reliability trade-off in MIMO communications. However, the “conservative scheduling” achieves a suboptimal performance. Choi et al. combined a two-segment queue structure and carrier sensing technology to design a fully distributed link scheduling algorithm based on the SINR model [17]. In [28], Chen et al. proposed a low-complexity approximate optimal scheduling algorithm based on a cross-entropy optimization framework.

Most previous work in MIMO wireless networks did not consider SIC. We therefore focus on link scheduling in MIMO wireless networks with SIC. The effectiveness of SIC has been verified recently [29]. In [30], Lv et al. studied link scheduling in a network with SIC but ignored the effects of aggregate interference. Then, in 2012, they took the lead in researching link scheduling under the SINR model in wireless networks with SIC [20]. The algorithm considers the influence of cumulative interference, but it is a greedy algorithm based on independent sets that can obtain an approximate optimal schedule. In [24], Kontik et al. proposed a heuristic algorithm based on the column generation method using SIC technology to study the problem of minimized scheduling length in single-hop wireless networks. The performance of this algorithm is very close to the optimal linear programming algorithm and has better robustness. SIC technology was shown to effectively improve network performance.

In addition, due to the degree of freedom (DoF) of MIMO, network throughput can be improved by spatial multiplexing (SM). Therefore, the problem of link scheduling based on DoF has also been extensively studied. Based on the DoF concept, Sultan et al. [31] proposed a handover standard that maximizes the capacity of the downlink channel when uplink capacity is maintained at a certain level. To further improve the overall performance of the network, the data link layer, the network layer, and the transmission layer need to be designed cooperatively for optimization. The layered protocol architecture with network adaptability has received widespread attention, such as the joint routing and scheduling optimization scheme [6] and joint power control and link scheduling optimization scheme [32, 33], but there are still many limitations in the practical applications of these optimization schemes. Therefore, a mathematical framework is established in this paper for the cross-layer optimization

of routing and scheduling, with constraints of traffic balance and link capacity.

3. The System Model

3.1. Network Model. When the links are transmitted in the SINR model, if the signal at the expected receiver is higher than a given threshold, the link is transmitted successfully [17], that is,

$$\text{SINR}_{r_i} = \frac{P_{r_i}(s_i)}{I_{r_i} + N_0} \geq \beta \quad (1)$$

where $P_{r_i}(s_i)$ is the received power at receiver r_i from sender s_i ; I_{r_i} is the aggregated interference from the active links in the neighbors of link l_i ; N_0 is the background noise; and β is the minimum SINR threshold required for receiver r_i to decode signals successfully.

Consider a set of MIMO networks consisting of n communication links $L = \{l_1, \dots, l_n\}$, where each link l_i includes a sender s_i and a receiver r_i . The Euclidean distance between s_j and r_i is $d_{ji} = d(s_j, r_i)$. Thus, the length of link l_i is d_{ii} . If s_i is the intended sender, (1) can be converted into

$$\begin{aligned} \text{SINR}_{r_i} &= \frac{P(s_i)/d(s_i, r_i)^\alpha}{N_0 + \sum_{s_j \in S_t \setminus \{s_i\}} P(s_j)/d(s_j, r_i)^\alpha} \\ &= \frac{P_i/d_{ii}^\alpha}{N_0 + \sum_{l_j \in L' \setminus \{l_i\}} P_j/d_{ji}^\alpha} \end{aligned} \quad (2)$$

where $P(s_j)$ is the transmission power from sender s_j ; α is the path-loss factor; S_t is the set of all senders that are transmitted concurrently in the same time slot t as the expected sender; L' is a set of links that are simultaneously scheduled in the same time slot.

Assume that all nodes are static and apply MIMO technology with M antennas. A node communicates with others through wireless links, and each node has an input links set L_i^{in} and an output links set L_i^{out} of node i . Suppose a time frame consists of T slots, and the state of a link subset in a time slot t ($1 \leq t \leq T$) depends on link scheduling. For a given slot t , the data flow from sender s_j to r_i can be expressed as a signal vector $\mathbf{x}_j = [x_j^1, x_j^2, \dots, x_j^M]^T$, and the MIMO signal y_{ji} received by receiver r_i from sender s_j can be expressed as the following:

$$\begin{aligned} y_{ji} &= \alpha_{ji} \mathbf{V}_{ji}^\dagger \mathbf{H}_{ji}^\dagger \mathbf{U}_j \mathbf{A}_j \mathbf{x}_j + \sum_{k \in L_i, k \neq j} \alpha_{ki} \mathbf{V}_{ji}^\dagger \mathbf{H}_{ki}^\dagger \mathbf{U}_k \mathbf{A}_k \mathbf{x}_k \\ &\quad + \mathbf{V}_{ji}^\dagger \mathbf{n}_i \end{aligned} \quad (3)$$

where $\mathbf{H}_{ki} \in C^{M \times M}$ is the channel matrix between sender s_k and receiver r_i and is normalized to mean power 1; \square^\dagger is the Hermitian operations of the corresponding matrix; $\mathbf{U}_k \in C^{M \times M}$ is the unitary transmitting precoding matrix at sender s_k ; $\mathbf{A}_j \in R^{M \times M}$ and $\mathbf{A}_j = \text{diag}\{\sqrt{p_j}, \sqrt{p_j}, \dots, \sqrt{p_j}\}$ is the real-valued diagonal transmit amplitude matrix, where p_j is the transmission power of the corresponding sender s_j ; $\mathbf{V}_{ji} \in$

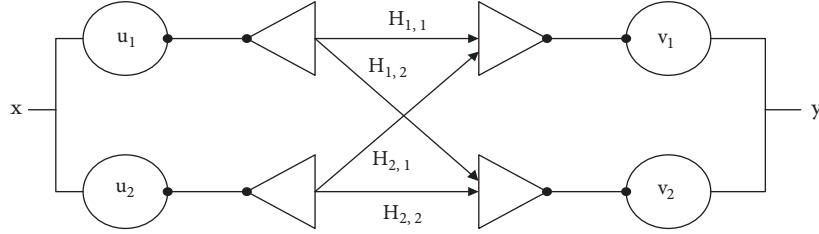


FIGURE 2: Multi-antenna Wireless Channel.

$C^{M \times M}$ is the unitary receiving decoding matrix at receiver r_i for signals from sender s_j ; $\mathbf{n}_i \in C^{M \times 1}$ is a white Gaussian noise vector with variance N_0 per element; $\|\cdot\|^2$ is the norm of the vector. Figure 2 shows that the sender and receiver 2-antenna array and MIMO channel are used at both ends. To simplify calculations, assuming that data streams are uncorrelated, the SINR for the n -th element in y_{ji} is given by the following:

$$SINR_{ji}^n = \frac{p_j \alpha_{ji}^2 \|\mathbf{v}_{ji}^{n\dagger} \mathbf{H}_{ji}^\dagger \mathbf{u}_j^n\|^2}{N_0 + \sum_{k \in I_i, k \neq j} p_k \alpha_{ki}^2 \|\mathbf{v}_{ji}^{n\dagger} \mathbf{H}_{ki}^\dagger \mathbf{u}_k\|^2} \quad (4)$$

3.2. MSIC Model. The manner in which interference is dealt with effectively affects the performance of the link scheduling algorithm in MIMO networks. The MSIC model is constructed based on SIC under the SINR model. The main idea of SIC is that the power level of the signals from K senders received at r_i are in descending order: $P_{r_i}(s_K) \geq P_{r_i}(s_{K-1}), \dots, P_{r_i}(s_i), \dots, P_{r_i}(s_2) \geq P_{r_i}(s_1)$ [20]. If the strongest signal dissatisfies the SINR constraint, the process of SIC ends. Otherwise this signal will be decoded, while other signals are considered interference and noise. If it is the expected signal, then the transmission is successful; if not, this signal is removed, and the remaining signals are decoded sequentially until the expected signal is decoded successfully.

In detail, the SIC model needs to satisfy the following constraints. Receiver r_i tries to decode signal from sender s_n in the order $k, k-1, \dots, n$. Then, the signal with received power $P_{r_i}(s_n)$ can be decoded successfully if and only if

$$\begin{aligned} \text{step 1} \quad & \frac{P_{r_i}(s_k)}{N_0 + \sum_{j \in I_i, j=1}^{k-1} P_{r_i}(s_j)} \geq \beta_{ki}, \\ \text{step 2} \quad & \frac{P_{r_i}(s_{k-1})}{N_0 + \sum_{j \in I_i, j=1}^{k-2} P_{r_i}(s_j)} \geq \beta_{(k-1)i}, \\ & \dots \\ \text{step } (k-n+1) \quad & \frac{P_{r_i}(s_n)}{N_0 + \sum_{j \in I_i, j=1}^{n-1} P_{r_i}(s_j)} \geq \beta_{ni}. \end{aligned} \quad (5)$$

If signal y_{ii} wants to be decoded correctly, it must satisfy the MSIC constraints shown in

$$SINR_{ji}^n = \frac{p_j \alpha_{ji}^2 \|\mathbf{v}_{ji}^{n\dagger} \mathbf{H}_{ji}^\dagger \mathbf{u}_j^n\|^2}{N_0 + \sum_{k \in I_i, k < j} p_k \alpha_{ki}^2 \|\mathbf{v}_{ji}^{n\dagger} \mathbf{H}_{ki}^\dagger \mathbf{u}_k\|^2} \geq \beta_{ji}, \quad \forall j, \text{ s.t. } P_{r_i}(s_i) < P_{r_i}(s_j) \quad (6)$$

$$SINR_{ii}^n = \frac{p_i \alpha_{ii}^2 \|\mathbf{v}_{ii}^{n\dagger} \mathbf{H}_{ii}^\dagger \mathbf{u}_i^n\|^2}{N_0 + \sum_{k \in I_i, k < i} p_k \alpha_{ki}^2 \|\mathbf{v}_{ii}^{n\dagger} \mathbf{H}_{ki}^\dagger \mathbf{u}_k\|^2} \geq \beta_{ii}$$

Because SIC is used, the receiver can sequentially cancel all interference signals that are stronger than its expected signal if those stronger signals satisfy (6). Therefore, it is only necessary to consider the residual interference at the sender, which is weaker than the expected signal. Specifically, the residual SINR from sender s_j to receiver r_i at time slot t is defined as follows:

$$\begin{aligned} r\text{-SINR}_{ji}[t] &= \frac{p_j \alpha_{ji}^2 \|\mathbf{v}_{ji}^{n\dagger} \mathbf{H}_{ji}^\dagger \mathbf{u}_j^n\|^2}{N_0 + \sum_{k \in I_i, k \neq j} p_k \alpha_{ki}^2 \|\mathbf{v}_{ji}^{n\dagger} \mathbf{H}_{ki}^\dagger \mathbf{u}_k\|^2 \leq p_j \alpha_{ji}^2 \|\mathbf{v}_{ji}^{n\dagger} \mathbf{H}_{ji}^\dagger \mathbf{u}_j^n\|^2} p_k \alpha_{ki}^2 \|\mathbf{v}_{ji}^{n\dagger} \mathbf{H}_{ki}^\dagger \mathbf{u}_k\|^2 \\ &\geq \beta_{ji} \end{aligned} \quad (7)$$

where sender s_k in the summation formula includes all senders' signals with weaker power than node s_j .

3.3. Problem Model. Consider the problem of throughput maximization in MIMO wireless networks. This paper transforms the cross-layer joint scheduling problem into congestion control, routing, and scheduling problems. By using local information, the congestion control problem is solved at the source node of each flow, and the routing and scheduling are transformed into the problem of flow balance and link capacity constraint, which facilitates distributed deployment and ensures network throughput.

Let F denote a set of active link session flows that describes the flow routing in the network. Denote $s(f)$ and $d(f)$ as the source and destination nodes of session flow $f \in F$, respectively. $Y(f)$ represents the reachable end-to-end throughput of session flow $f \in F$, and Y_{\min} is the minimum reachable end-to-end throughput in all sessions, that is, $Y_{\min} = \min_{f \in F} Y(f)$. $Y_l(f)$ represents the number of data rates caused by session flow $f \in F$ on link l . The goal of this paper is to maximize the minimum reachable end-to-end throughput Y_{\min} to maximize network throughput.

We assume a half-duplex node on a MIMO node. If node $i \in N$ is a sender in time slot t , the binary variable $s_i(t)$ is 1; otherwise, it is 0. Similarly, if node $i \in N$ is a receiver in time slot t , the binary variable $r_i(t)$ is 1; otherwise, it is 0. For half-duplex mode, the constraint can be written as follows:

$$s_i(t) + r_i(t) \leq 1, \quad (1 \leq i \leq N, 1 \leq t \leq T) \quad (8)$$

Denote $x_l(t)$ as the number of data streams over link l . If node i is not a sender, then there is $\sum_{l \in L_i^{out}} x_l(t) = 0$. Otherwise, in order to satisfy the DoF constraint at the sending node, the total number of outgoing data streams should be positive and cannot exceed the number of antennas it owns, that is, $1 \leq \sum_{l \in L_i^{out}} x_l(t) \leq M$. These two cases can be summarized as follows:

$$s_i(t) \leq \sum_{l \in L_i^{out}} x_l(t) \leq M s_i(t), \quad (1 \leq i \leq N, 1 \leq t \leq T) \quad (9)$$

Similarly, according to whether node i is an active receiver, we have the following constraint at the receiving node:

$$r_i(t) \leq \sum_{l \in L_i^{in}} x_l(t) \leq M r_i(t), \quad (1 \leq i \leq N, 1 \leq t \leq T) \quad (10)$$

For the congestion control problem of the transport layer, each flow f in the network determines the data rate of the next slot independently according to the local congestion queue information of the node in the current time slot. $U_i^f(t)$ represents the congestion queue length of flow f at node i in time slot t , and $\mathbf{U}(t) = [U_i^f(t)]$, ($i \in N, f \in F$) is the set of all congestion queues. According to the input and output mode of each node, the local congestion queue information is updated dynamically as follows: $U_i^f(t+1) = U_i^f(t) + \sum_{l \in L_i^{in}} x_l(t) - \sum_{l \in L_i^{out}} x_l(t) + v_f(t)$, and the initialized condition queue is satisfied with $U_i^f(0) = 0$. The source session flow f is compressed into a source rate $v_f(t)$ before being pushed into the queue. Since the local congestion queue length of each node determines the upper limit of the data sum that can be transmitted in the current time slot t , there are the following constraints:

$$\sum_{l \in L_i^{out}} x_l(t) \leq U_i^f(t), \quad (1 \leq i \leq N, 1 \leq t \leq T, f \in F) \quad (11)$$

The session flow f in the network determines the data rate of the next time slot according to the congestion queue constraint. Meanwhile, in order to guarantee the strong robustness of the network, the following inequalities must be satisfied:

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i \in N} \sum_{f \in F} E \{U_i^f(t)\} < \infty \quad (12)$$

For routing and scheduling problems, $D = \max\{\sum_{l \in L} r_l(f) * \max_{f \in F} \{u_{s_l}(f) - u_{d_l}(f)\}\}$, and each link l can use local congestion queue information to find a flow f^* that satisfies $f^* = \arg \max_{f \in F} \{u_{s_l}(f) - u_{d_l}(f)\}$. Let $w_l = u_{s_l}(f^*) - u_{d_l}(f^*)$ as the weight of link l (w_l can also

be understood as the queue length at link l with flow f^*). To solve routing and scheduling problems, we will propose a distributed algorithm in Section 4 to generate an active set L_S' of concurrent links. In each time slot, the links in set L_S' can send data to the receivers (we assume that, in each time slot, each active link transmits a packet). Let D be converted into the following form:

$$D = \max_{l \in L} \sum_{l \in L} Y_l(f^*) \cdot w_l \quad (13)$$

To achieve the goal of maximizing the minimum end-to-end throughput Y_{\min} , a feasible routing scheduling also needs to satisfy the following two constraints: flow balance constraints and the link capacity constraint. We have the following flow balance constraints:

(a) at the source node, we have

$$\sum_{l \in L_i^{out}} Y_l(f) = Y(f), \quad (i = s(f), f \in F) \quad (14)$$

(b) at each relay node, we have

$$\sum_{l \in L_i^{in}} Y_l(f) = \sum_{l \in L_i^{out}} Y_l(f), \quad (1 \leq i \leq N, i \neq s(f), i \neq d(f), f \in F) \quad (15)$$

(c) at the destination node, we have

$$\sum_{l \in L_i^{in}} Y_l(f) = Y(f), \quad (i = d(f), f \in F) \quad (16)$$

It is easy to verify that as long as any two of these equations are satisfied, the other one will also be satisfied. Therefore, it is sufficient to satisfy the first two equations.

It is assumed that a fixed modulation and encoding scheme is used for each data stream and that each data stream corresponds to one unit data rate. Since the total data rate on link l cannot exceed the average rate of the link, we have the following link capacity constraint:

$$\sum_{f \in F} Y_l(f) \leq \frac{1}{T} \sum_{t=1}^T x_l(t) \quad (17)$$

where the right side represents the average throughput on link l of a frame (T time slots). Putting all the constraints together, we have the expression for the throughput maximization problem:

max Y_{\min}

s.t. $Y_{\min} \leq Y(f), \quad (f \in F);$

r -SINR $_{ji}[t]$

$$= \frac{p_j \alpha_{ji}^2 \|\mathbf{v}_{ji}^{\dagger} \mathbf{H}_{ji}^{\dagger} \mathbf{u}_j^{\dagger}\|^2}{N_0 + \sum_{k \in L_i, k \neq j} p_k \alpha_{ki}^2 \|\mathbf{w}_{ki}^{\dagger} \mathbf{H}_{ki}^{\dagger} \mathbf{u}_k^{\dagger}\|^2 \leq p_j \alpha_{ji}^2 \|\mathbf{v}_{ji}^{\dagger} \mathbf{H}_{ji}^{\dagger} \mathbf{u}_j^{\dagger}\|^2} p_k \alpha_{ki}^2 \|\mathbf{v}_{ji}^{\dagger} \mathbf{H}_{ki}^{\dagger} \mathbf{u}_k^{\dagger}\|^2$$

$$\geq \beta_{ji};$$

$s_i(t) + r_i(t) \leq 1, \quad (1 \leq i \leq N, 1 \leq t \leq T);$

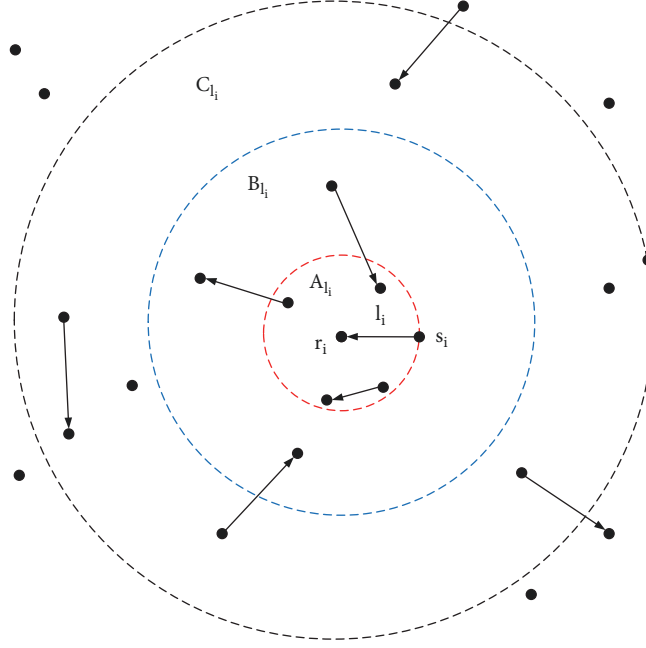


FIGURE 3: Division of Interference Regions.

$$\begin{aligned}
s_i(t) &\leq \sum_{l \in L_i^{out}} x_l(t) \leq Ms_i(t), \quad (1 \leq i \leq N, 1 \leq t \leq T); \\
r_i(t) &\leq \sum_{l \in L_i^{in}} x_l(t) \leq Mr_i(t), \quad (1 \leq i \leq N, 1 \leq t \leq T); \\
\sum_{l \in L_i^{out}} x_l(t) &\leq U_i^f(t), \quad (1 \leq i \leq N, 1 \leq t \leq T, f \in F); \\
\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i \in N} \sum_{f \in F} E \{U_i^f(t)\} &< \infty; \\
\sum_{l \in L_i^{out}} Y_l(f) &= Y(f), \quad (i = s(f), f \in F); \\
\sum_{l \in L_i^{in}} Y_l(f) &= \sum_{l \in L_i^{out}} Y_l(f), \\
(1 \leq i \leq N, i \neq s(f), i \neq d(f), f \in F); \\
\sum_{f \in F} Y_l(f) &\leq \frac{1}{T} \sum_{t=1}^T x_l(t).
\end{aligned} \tag{18}$$

4. Distributed Link Scheduling Algorithm Based on MSIC

4.1. Division of Interference Regions. Due to the characteristics of transmission loss, the receiving power $P_{r_i}(s_j)$ of different links is also different [34]. The total interference at receiver r_i is expressed as follows:

$$I_{r_i} = \sum_{s_j \in S_i \setminus \{s_i\}} \frac{P(s_j)}{d(s_j, r_i)^\alpha} = \sum_{s_j \in S_i \setminus \{s_i\}} P_{r_i}(s_j) \tag{19}$$

The scheduling problem of using SIC under the SINR model has been proved to be NP-hard [20]. If the stronger signal in the network satisfies (5), it can be decoded and removed due to the adoption of SIC. Therefore, the strength of the receiving power does not completely measure the interference generated by the link. In this paper, in order to describe the interference localization, the interference regions are defined to measure the level of interference.

Different interference regions A_{l_i} , B_{l_i} , and C_{l_i} are divided according to (19). A_{l_i} , B_{l_i} , and C_{l_i} are concentric ring (circular) regions centered on receiver r_i of link l_i , respectively, as shown in Figure 3 (the red circular region is A_{l_i} , the blue ring region is B_{l_i} , and the black dotted ring region is C_{l_i}). The interference generated by the active link in the A_{l_i} and B_{l_i} regions will interrupt the transmission of link l_i , and the set of concurrent links in these regions is denoted as L_S . The cumulative interference generated by the active link in the C_{l_i} region has a negligible effect on the receiver of link l_i .

According to (2), the maximum cumulative interference value that link l_i can tolerate is the following:

$$I_{r_i} \leq I_{l_i}^{\max} \leq \frac{P_{l_i}}{\beta d_{ii}^\alpha} \tag{20}$$

Therefore, in order to satisfy the MSIC constraints, the sum of the cumulative interference values at receiver r_i of link l_i cannot exceed $I_{l_i}^{\max}$. If the total interference in the A_{l_i} and B_{l_i} regions is $(1 - m)I_{l_i}^{\max}$ and the total interference in the C_{l_i} region is $mI_{l_i}^{\max}$, then we have

$$(1 - m)I_{l_i}^{\max} + mI_{l_i}^{\max} \leq I_{l_i}^{\max} \tag{21}$$

Assuming that the link l_j at receiver r_i from sender $s_j \in A_{l_i}$ is stronger than link l_i from sender s_i , SIC is used. That is,

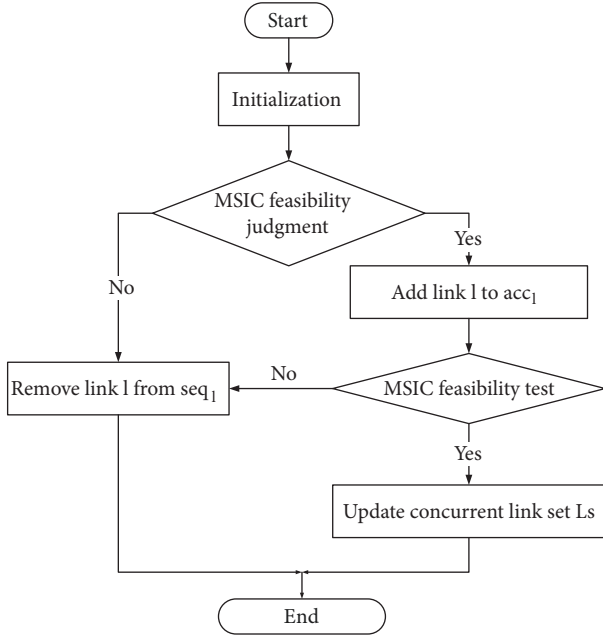


FIGURE 4: Distributed MSIC scheduling algorithm flow chart.

receiver r_i uses SIC to decode these strong signals continuously before decoding the expected signal from sender s_i . Let all links in a feasible set L_S of concurrent links be transmitted successfully. When any link l_j is added to the set $L_S' \subseteq L_S$, the link transmission will fail. Obviously, L_S' is the maximum set of concurrent links, and L_S' must satisfy the constraints in (7).

4.2. Distributed MSIC Scheduling Algorithm. In this section, based on the CSMA/CA mechanism and MSIC constraints, we design the distributed single-slot MSIC algorithm to solve the scheduling problems. If the links $l \in L'$ in the network can be transmitted concurrently, then L' can be defined as a scheduling set. In each scheduling time slot, each link will run scheduling algorithms to generate a new feasible scheduling set L_S' .

Next, considering the MSIC model and interference regions division introduced above, a distributed link scheduling algorithm is proposed under the MSIC constraints. The algorithm flow chart is shown in Figure 4. Three states of the link are defined: Active, Inactive, and Standby. The four sets correspond to different states, where sic is the active link set that satisfies the MSIC constraint and can be scheduled in the current scheduling time slot; acc is the standby link set that has not been judged by MSIC constraints; loc is the local cache link set in the standby state; and seq is the candidate link set in the inactive state. Therefore, the state of a link in the current slot can be distinguished by the set of links to which the link currently belongs.

(1) Initialization Stage. At the beginning of each scheduling cycle, each link l maintains two sets of local links: set acc_l and set seq_l . The links contained in the set acc_l are added to

a feasible set L_S of concurrent links in a certain scheduling cycle, and the links contained in set seq_l are candidate links to be added into set L_S . At the beginning of each scheduling cycle, initializing $acc_l = \emptyset$ and $seq_l = \{l, A_l \cup B_l\}$.

(2) MSIC Feasibility Judgment Stage. Each scheduling cycle consists of several slots, and in each slot each link l makes a decision about whether it should be added to acc_l or removed from seq_l . At the beginning of each scheduling slot, all links in the set seq_l are weighted for comparison. The sender of link l broadcasts its weight information w_l to all links k whose sender is located in the A_l region of link l . If link l has the largest weight among all links in seq_l , it will run Algorithm 1 to try to add itself to acc_l .

The detailed process is as follows. The sender of link l broadcasts a Request message to links in $\{A_l \cup B_l\}$ first. If there is a collision, all of the links select a random back-off time and wait for it. If there is no collision, the sender of all links in $\{A_l \cup B_l\}$ will add link l to set loc_l . When link l is added to the current schedule, any link $l' \in \{acc_l \cup loc_l\}$ will determine whether it remains feasible under the MSIC constraints. The specific judgment process is as follows. For each link $l' \in \{acc_l \cup loc_l\}$, the set of links with sender $s_{l'} \in \{\{acc_l \cup loc_l\} \cap A_{l'}\}$ and receiver $r_{l'}$ is defined as the MSIC link set $sic_{l'}$. The MSIC constraints will be satisfied when any link $l' \in \{acc_l \cup loc_l\}$ satisfies the following conditions: (i) the total interference $I_{l'}$ coming from set $B_{l'}$ does not exceed $(1 - m)I_{l'}^{\max}$; (ii) all links $k \in sic_{l'}$ are feasible; that is, the total interference I_k coming from set B_k does not exceed $(1 - m)I_k^{\max}$.

According to the above procedure, if any link $l' \in \{acc_l \cup loc_l\}$ does not satisfy the MSIC constraints, the sender of link l' will send an Error message to link l indicating that link l cannot be added to set L_S (the current scheduling is not feasible due to the strong interference caused by link l). If link $l \in loc_l$ does not receive any Error messages from its neighbors, it adds link l to set acc_l and removes link l from seq_l and loc_l . Then, the sender broadcasts a Success message to all its neighbors to update their link sets acc , seq , and loc . Otherwise, it removes link l from seq_l and loc_l and then broadcasts a Remove message to all its neighbors to update their link sets seq and loc . After the above process is performed until a new link is added to L_S , the current scheduling process is still feasible under the MSIC constraints.

(3) MSIC Feasibility Test Stage. After a new L_S' is generated in each slot, all links $l \in seq_l$ need to make the following decision about whether the MSIC constraints are satisfied. For each link $l \in seq_l$, the set of links with sender $s_l \in \{acc_l \cap A_l\}$ and receiver r_l is defined as another MSIC link set sic_l' . Similar to the first procedure, any link $l \in seq_l$ that satisfies any of the following conditions will dissatisfy the MSIC constraints: (i) the total interference I_l' coming from set B_l exceeds $(1 - m)I_l^{\max}$ or (ii) for any link $k' \in sic_l'$ is not feasible; that is, the total interference $I_{k'}$ coming from set $B_{k'}$ exceeds $(1 - m)I_{k'}^{\max}$. After the above process, if there is a link l in seq_l that does not satisfy the MSIC constraints, the sender will remove link l from seq_l and broadcast a Remove message to all its neighbors to update their link sets seq and acc . The

```

1 sets of int  $acc_l, seq_l, loc_l, sic_l', sic_l'$            % initialize
2  $acc_l \leftarrow \emptyset$  and  $seq_l \leftarrow \{l, A_l \cup B_l\}$ 
3 Sender of link  $l$  broadcasts Request message to links in  $\{A_l \cup B_l\}$ ;
4 for link  $l' \in \{\{A_l \cup B_l\} \cap \{acc_l \cup loc_l\}\}$  do           % MSIC feasibility judgment
5   if sender of link  $l'$  receives Request message from sender of link  $l$  then
6     sender of link  $l'$  adds link  $l$  into  $loc_l$ ;
7     sender of link  $l'$  calculates cumulative interference  $I_{l'}$ ;
8     if  $I_{l'} > (1 - m)I_{l'}^{\max}$  then
9       Sender of link  $l'$  broadcasts Error message to sender of link  $l$ ;
10    else
11      Generate  $sic_l'$ ;
12      for  $k \in sic_l'$  do
13        Link  $k$  calculates cumulative interference  $I_k$ ;
14        if  $I_k > (1 - m)I_k^{\max}$  then
15          Sender of link  $l'$  broadcasts Error message to sender of link  $l$ ;
16        end if
17      end for
18    end if
19  end if
20 end for
21 if sender of link  $l$  does not receive Error messages then % Generate concurrent links set
22    $acc_l \leftarrow acc_l \cup \{l\}$ ;
23    $seq_l \leftarrow seq_l / \{l\}$ ;
24    $loc_l \leftarrow loc_l / \{l\}$ ;
25   Sender of link  $l$  broadcasts Success message to all its neighbors to update their link sets  $acc$ ,  $seq$  and  $loc$ ;
26   Goto Algorithm 2;
27 else
28    $seq_l \leftarrow seq_l / \{l\}$ ;           % Does not satisfy MSIC feasibility conditions
29   Sender of link  $l$  broadcasts Remove message to all its neighbors to update their local link sets  $seq$  and  $loc$ ;
30 end if

```

ALGORITHM 1: Distributed MSIC Scheduling Algorithm (MSIC feasibility judgment).

above process ensures that each link in seq_l satisfies the MSIC constraints under the current scheduling.

In the distributed algorithm, the number of time slots in each scheduling cycle is a fixed value. In each scheduling slot, each link will run the distributed scheduling algorithm to generate a new feasible scheduling set L_S' during scheduling. Once the scheduling is completed, the selected link will transmit a packet during the transmission cycle.

4.3. Theoretical Proof of Algorithm

Theorem 1. *The set L_S of scheduling generated by the MSIC algorithm is feasible.*

Proof. Based on the MSIC algorithm under interference regions division, each link will run the algorithm independently in each scheduling slot to generate a new feasible scheduling set L_S' . From Algorithm 2, it is known that after a new set L_S' is generated in each time slot, all links $l \in seq_l$ need to be tested to determine if the MSIC constraints are still satisfied. To ensure that each link in seq_l satisfies the MSIC constraints under the current scheduling, any link l that does not satisfy the MSIC constraints will be removed from seq_l . Therefore, the interference links cannot be scheduled at the same time; that is, the links in the current set L_S' are feasible, which ensures the feasibility of the algorithm. \square

Theorem 2. *The time complexity of the MSIC algorithm is $O(k^2)$.*

Proof. In the process of MSIC feasibility judgment of Algorithm 1, it is necessary to judge the MSIC constraints formula (4) of any link $l' \in \{acc_l \cup loc_l\}$ and decide whether to add the link to the current scheduling set. This process requires two loop statements to be executed with the algorithm execution complexity of $O(k^2)$, where k represents the number of concurrently transmitted signals. After Algorithm 1 generates a new set L_S' , the MSIC feasibility test is performed in Algorithm 2. The MSIC constraints condition needs to be tested again for all links $l \in seq_l$ to ensure that any link in seq_l satisfies the MSIC constraints. This process also needs to execute two loop statements with the algorithm execution complexity of $O(k^2)$. Therefore, the total algorithm execution complexity is $O(k^2)$. \square

Theorem 3. *The messages complexity of the MSIC algorithm is $O(n)$.*

Proof. The message complexity of sending a Request broadcast message is $O(1)$. To find a set of links that can be concurrently given by a network of n links, an MSIC constraint judgment is performed for each link of $l' \in \{\{A_l \cup B_l\} \cap \{acc_l \cup loc_l\}\}$, and the number of Error or Success messages

```

1 sets of int  $I_k', I_j', sic_k'$ 
2 for link  $k \in \{\{A_l \cup B_l\} \cap seq_l\}$  do           % MSIC feasibility test
3   if the sender of link  $k$  receives a Success message from the sender of link  $l$  then
4     Link  $k$  calculates the cumulative interference  $I_k'$ ;
5     if  $I_k' > (1 - m)I_k^{\max}$  then
6       The sender of link  $k$  broadcasts a Remove message to its neighbors to update their link sets  $seq$ ;
7     else
8       Generate  $sic_k'$ ;
9       for  $j \in sic_k'$  do
10        Link  $j$  calculates the cumulative interference  $I_j'$ ;
11        if  $I_j' > (1 - m)I_j^{\max}$  then
12          The sender of link  $k$  broadcasts a Remove message to its neighbors to update their link sets  $seq$ ;
13        end if
14      end for
15    end if
16  end if
17 end for

```

ALGORITHM 2: Distributed MSIC Scheduling Algorithm (MSIC feasibility test).

sent is at most $n - 1$. For each link of $k \in \{\{A_l \cup B_l\} \cap seq_l\}$, a feasibility test is performed, and the number of Remove or Success messages is up to $n - 1$. When the execution of the algorithm ends, the total number of messages used is up to $O(n)$. \square

5. Experimental Results

In this section, the distributed single-slot scheduling problem in MIMO wireless networks is studied under the SINR model. The simulation is the average of 50 trials obtained on a network with 100 links. The network size is $600 * 600$, and the distance between the sender and the receiver is selected within the range of $[20, 40]$. The SINR parameters are set as follows: the threshold value is $\beta = 3$, the path loss index is $\alpha = 2.2$, the background noise power is $N = 4 \times 10^{-7}$, and the uniform power is $P = 2$ and $P = 10$.

First, the relationship between the size of different networks and the number of successful transmissions of links is studied. The simulation results are shown in Figure 5. Under different power allocations, the number of successful transmissions of the link increases as the network size increases, and a larger transmission power can generate a larger neighbor size, which is benefited by the gain of the transmission power. However, when the network size is large, the neighbor size reaches the upper bound, and the gain from increasing the power becomes very small. At the same time, larger power will also cause more interference to other links. At this time, the impact of the interference on transmission is greater than the benefit of the network scale increase, which makes it impossible to satisfy the SINR constraint condition, although it improves its own successful transmission probability and reduces the total size of the successful transmission of the link. Therefore, the transmission scale cannot be sought blindly by increasing the power, and a larger-scale scheduling set can be realized by controlling the network scale.

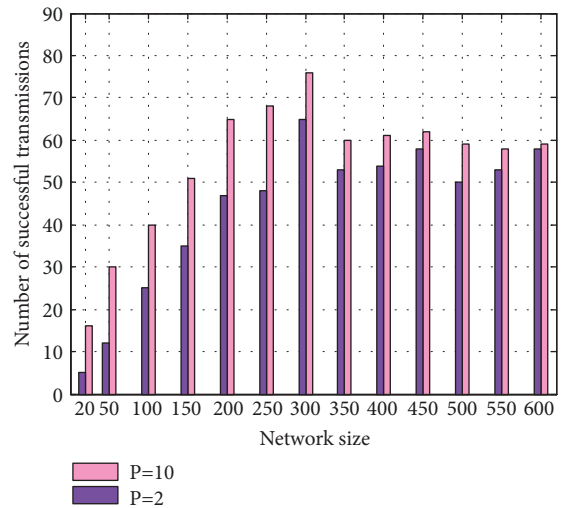


FIGURE 5: The relationship between network size and number of successful transmissions.

Next, we compare the number of concurrent link transmissions of different algorithms. At the network layer, the routing algorithm based on the least hops is adopted. For convenience, the distance, data rate, bandwidth, and transmission power of the data stream are all normalized to 1.

First, the MSIC algorithm is compared with the recursive largest first (RLF) algorithm and the smallest degree first (SDF) algorithm in [20]. The results in Figure 6 show that the MSIC algorithm can obtain a larger link scheduling scale compared with the other two algorithms. After the SIC is used, all signals with high power in the interference regions can be decoded and removed firstly, and thus the interference reached at the receiver is smaller, and a larger set of concurrent links can be obtained.

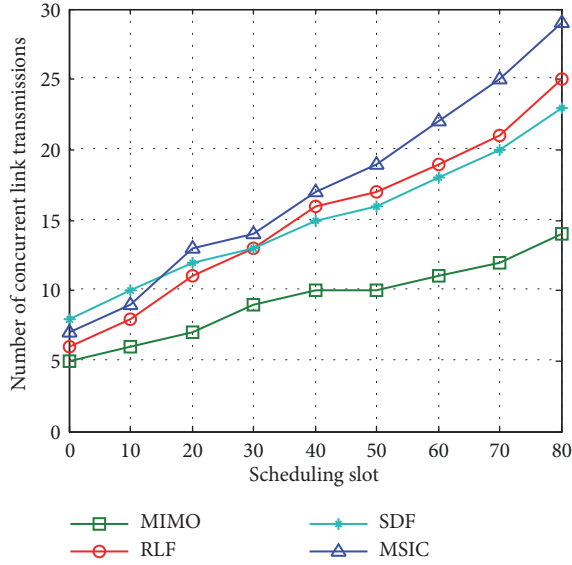


FIGURE 6: Comparison of the number of concurrent link transmissions of different algorithms.

Then, we study the effect of the parameter m on the number of concurrent link transmissions. Figure 7 shows that the number of concurrent links decreases as the parameter m increases. When m is smaller, the interference radius of each link in the network is larger, and there are more links that can participate in scheduling. The number of concurrent links in each selected scheduling configuration is larger, and the data flow rate is higher. At this time, the network has stronger interference management ability. As the parameters become larger, the interference in the network is larger, thereby allowing fewer links to be transmitted concurrently. The MSIC algorithm can reduce partial interference and increase the number of concurrent transmission links. When $m = 0.4$, there is a close interference region radius between MSIC and MIMO, so $m = 0.4$ is set here.

Next, we study the influence of the number of antennas on network performance. The number of antennas is changed from 2 to 6, and the test is repeated 100 times. We then take the average value of each flow of 100 tests. Ideally, we assume that the transmission rate of one link is equal to the capacity of the point-to-point MIMO link without other transmission interference. As shown in Figure 8, throughput is nearly linear with the number of antennas, and the MSIC algorithm can achieve a higher minimum flow throughput and a higher total throughput in the network.

To show more detail, the flow rate gains for the four flow sessions are given in Figure 9. The simulation results show that MSIC brings significant throughput gain to MIMO wireless networks. The network flow rate with SIC functionality is more than half of the network rate without interference cancellation, which means that the reachable unit end-to-end throughput is increased by approximately 73% on average due to the application of SIC. Therefore, the MSIC algorithm has obvious advantages in improving network throughput.

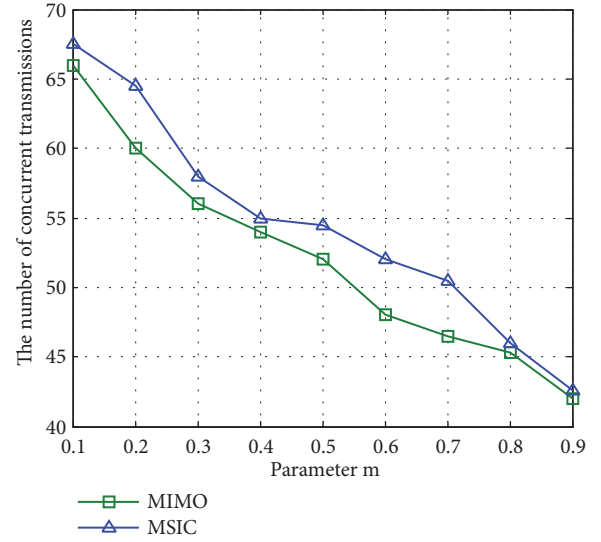


FIGURE 7: The relationship between parameter m and the number of concurrent transmissions.

6. Conclusions and Future Work

In this paper, the MSIC scheduling algorithm based on SIC in MIMO wireless networks is proposed. First, the MSIC constraints model is constructed under the SINR interference model. Then, interference region division is carried out to describe the level of interference between links. A feasible scheduling set is generated by the distributed link scheduling algorithm to coordinate the link transmission, and the interference between competing links in the MIMO network is cancelled. Experimental results show that the distributed MSIC scheduling algorithm can bring significant performance gain to wireless networks.

Since the algorithm is premised on satisfying the threshold constraint of SINR, the algorithm ends when the SINR value is less than the current threshold. Therefore, the next research goal is how to solve the link scheduling problem when the SINR value is less than threshold.

With the rapid development of the social economy, the application of modern communication technology has been continuously promoted in various fields, and the capacity requirement for networks is increasing. In addition, due to the complexity of the network environment, a single technology will not have a sufficient effect on interference cancellation. Therefore, in future wireless communication networks, a variety of joint applications of multiple interference cancellation technologies will be needed. How to combine these technologies efficiently and achieve a reasonable optimization combination is an important research direction.

Data Availability

The simulation data used to support the findings of this study are included within the article.

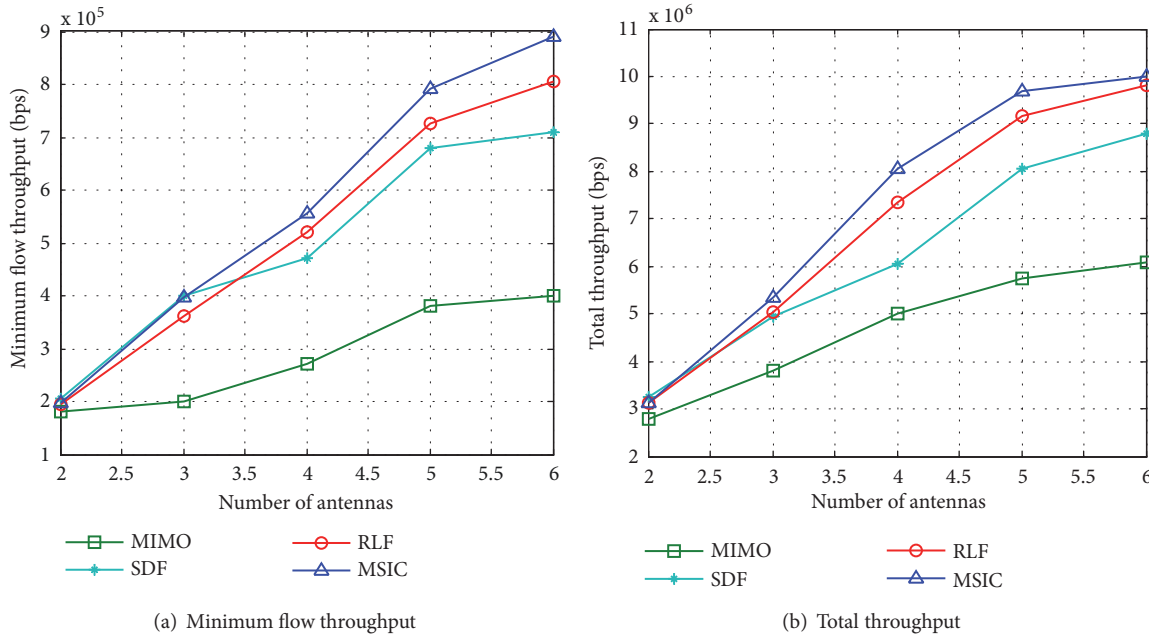


FIGURE 8: The relationship between antenna number and throughput.

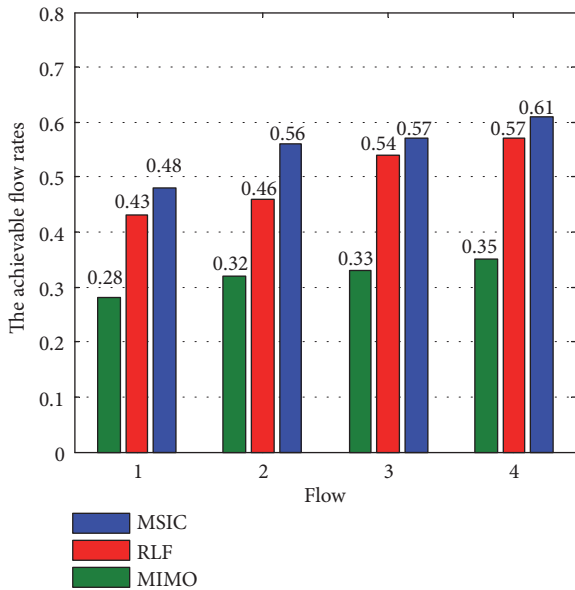


FIGURE 9: The gains of data flow rate for the four flow sessions.

Conflicts of Interest

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

Acknowledgments

This work is supported by the National Natural Science Foundation of China (61672321, 61771289, and 61832012),

Shandong provincial Graduate Education Innovation Program (SDYY14052 and SDYY15049), Qufu Normal University Science and Technology Project (xkj201525), Shandong province agricultural machinery equipment research and development innovation project (2018YZ002), Shandong Provincial Specialized Degree Postgraduate Teaching Case Library Construction Program, and Shandong Provincial Postgraduate Education Quality Curriculum Construction Program. The authors thank the School of Information Science and Engineering, Qufu Normal University.

References

- [1] S. Cheng, Z. Cai, J. Li, and H. Gao, "Extracting kernel dataset from big sensory data in wireless sensor networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 4, pp. 813–827, 2017.
- [2] S. Cheng, Z. Cai, J. Li, and X. Fang, "Drawing dominant dataset from big sensory data in wireless sensor networks," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '15)*, pp. 531–539, Hong Kong, April 2015.
- [3] X. Zheng and Z. Cai, "Real-time big data delivery in wireless networks: a case study on video delivery," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 2048–2057, 2017.
- [4] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, 2018.
- [5] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive MIMO: benefits and challenges," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 742–758, 2014.
- [6] H. Liu, L. Luo, D. Wu, J. Yu, and D. Chen, "Routing, spectrum access, and scheduling in multi-hop multi-channel wireless networks with MIMO links," *EURASIP Journal on Wireless*

- Communications and Networking*, vol. 15, no. 1, article no 65, 2015.
- [7] B. Hamdaoui and K. G. Shin, "Characterization and analysis of multi-hop wireless mimo network throughput," in *Proceedings of the Eighth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 120–129, Montreal, Quebec, Canada, September 2007.
 - [8] L. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 3, pp. 960–972, 2010.
 - [9] H. Yu, O. Bejarano, and L. Zhong, "Combating Inter-cell Interference in 802.11ac-based multi-user MIMO networks," in *Proceedings of the 20th ACM Annual International Conference on Mobile Computing and Networking (MobiCom '14)*, pp. 141–152, Maui, Hawaii, USA, September 2014.
 - [10] L. B. Le, E. Modiano, C. Joo, and N. B. Shroff, "Longest-queue-first scheduling under SINR interference model," in *Proceedings of the 11th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2010*, pp. 41–50, USA, September 2010.
 - [11] J. Li, S. Cheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Approximate holistic aggregation in wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 13, no. 2, article no. 11, 2017.
 - [12] Y. Zhou, X. Li, M. Liu, X. Mao, S. Tang, and Z. Li, "Throughput optimizing localized link scheduling for multihop wireless networks under physical interference model," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 10, pp. 2708–2720, 2014.
 - [13] G. Pei and A. K. S. Vullikanti, "Distributed approximation algorithms for maximum link scheduling and local broadcasting in the physical interference model," in *Proceedings of the 32nd IEEE Conference on Computer Communications (IEEE INFOCOM '13)*, pp. 1339–1347, Turin, Italy, April 2013.
 - [14] J. Yu, B. Huang, X. Cheng, and M. Atiquzzaman, "Shortest link scheduling algorithms in wireless networks under the SINR model," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 2643–2657, 2017.
 - [15] P. J. Wan, X. Xu, and O. Frieder, "Shortest link scheduling with power control under physical interference model," in *Proceeding of the Sixth International Conference on Mobile Ad-Hoc and Sensor Networks*, pp. 74–78, 2010.
 - [16] M. Nabli, F. Abdelkefi, W. Ajib, and M. Siala, "Efficient centralized link scheduling algorithms in wireless mesh networks," in *Proceedings of the 10th International Wireless Communications and Mobile Computing Conference, IWCMC 2014*, pp. 660–665, Cyprus, August 2014.
 - [17] J.-G. Choi, C. Joo, J. Zhang, and N. B. Shroff, "Distributed link scheduling under SINR model in multihop wireless networks," *IEEE/ACM Transactions on Networking*, vol. 22, no. 4, pp. 1204–1217, 2014.
 - [18] J. Park and S. Lee, "Distributed MIMO Ad-hoc networks: Link scheduling, power allocation, and cooperative beamforming," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 6, pp. 2586–2598, 2012.
 - [19] Y. Shi, Y. Thomas, H. J. Liu, and S. Kompella, "How to correctly use the protocol interference model for multi-hop wireless networks," in *Proceedings of the 10th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc'09*, pp. 239–248, USA, May 2009.
 - [20] S. Lv, X. Wang, and X. Zhou, "Scheduling under the SINR model in ad hoc networks with successive interference cancellation," *Computer Engineering and Science*, vol. 34, no. 2, pp. 1–5, 2012.
 - [21] D. Qian, D. Zheng, J. Zhang, and N. Shroff, "CSMA-based distributed scheduling in multi-hop MIMO networks under SINR model," in *Proceedings of the IEEE INFOCOM 2010*, pp. 1–9, USA, March 2010.
 - [22] O. Goussevskaia, M. M. Halldorsson, and R. Wattenhofer, "Algorithms for wireless capacity," *IEEE/ACM Transactions on Networking*, vol. 22, no. 3, pp. 745–755, 2014.
 - [23] O. Goussevskaia and R. Wattenhofer, "Scheduling wireless links with successive interference cancellation," in *Proceedings of the 21st International Conference on Computer Communications and Networks, ICCCN 2012*, pp. 1–7, Germany, August 2012.
 - [24] M. Kontik and S. C. Ergen, "Scheduling in single-hop multiple access wireless networks with successive interference cancellation," *IEEE Wireless Communications Letters*, vol. 3, no. 2, pp. 197–200, 2014.
 - [25] T. Moscibroda, R. Wattenhofer, and A. Zollinger, "Topology control meets SINR: the scheduling complexity of arbitrary topologies," in *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '06)*, pp. 310–321, Florence, Italy, May 2006.
 - [26] O. Goussevskaia, R. Wattenhofer, M. M. Halldorsson, and E. Welzl, "Capacity of arbitrary wireless networks," in *Proceedings of the 28th Conference on Computer Communications (IEEE INFOCOM '09)*, pp. 1872–1880, Rio de Janeiro, Brazil, April 2009.
 - [27] M. Dinitz, "Distributed algorithms for approximating wireless network capacity," in *Proceedings of the IEEE INFOCOM*, pp. 1–9, IEEE, San Diego, Calif, USA, March 2010.
 - [28] J. Chen, C. Wen, S. Jin, and K. Wong, "A low complexity pilot scheduling algorithm for massive MIMO," *IEEE Wireless Communications Letters*, vol. 6, no. 1, pp. 18–21, 2017.
 - [29] D. Halperin, T. Anderson, and D. Wetherall, "Taking the sting out of carrier sense: interference cancellation for wireless LANs," in *Proceedings of the 14th ACM Annual International Conference on Mobile Computing and Networking (MobiCom '08)*, pp. 339–350, September 2008.
 - [30] S. Lv, W. Zhuang, X. Wang, and X. Zhou, "Scheduling in wireless ad hoc networks with successive interference cancellation," in *Proceedings of the IEEE INFOCOM 2011 - IEEE Conference on Computer Communications*, pp. 1287–1295, Shanghai, China, April 2011.
 - [31] R. Sultan, L. Song, K. G. Seddik, and Z. Han, "Full-duplex meets multiuser MIMO: comparisons and analysis," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 455–467, 2017.
 - [32] X. Zheng, Z. Cai, J. Li, and H. Gao, "A study on application-aware scheduling in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 7, pp. 1787–1801, 2017.
 - [33] X. Li, Y. Shi, X. Wang, C. Xu, and M. Sheng, "Efficient link scheduling with joint power control and successive interference cancellation in wireless networks," *Science China Information Sciences*, vol. 59, no. 12, pp. 1–15, 2016.
 - [34] L. Qu, J. He, and C. Assi, "Distributed link scheduling in wireless networks with interference cancellation capabilities," in *Proceedings of the 15th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM 2014*, pp. 1–7, Australia, 2014.

Research Article

A Lightweight Fine-Grained Searchable Encryption Scheme in Fog-Based Healthcare IoT Networks

Hui Li ¹ and Tao Jing ²

¹*School of Computer and Information Technology, Beijing Jiaotong University, China*

²*School of Electronics and Information Engineering, Beijing Jiaotong University, China*

Correspondence should be addressed to Hui Li; huilee@bjtu.edu.cn

Received 1 March 2019; Accepted 6 May 2019; Published 23 May 2019

Guest Editor: Jun Liu

Copyright © 2019 Hui Li and Tao Jing. 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.

For a smart healthcare system, a cloud based paradigm with numerous user terminals is to support and improve more reliable, convenient, and intelligent services. Considering the resource limitation of terminals and communication overhead in cloud paradigm, we propose a hybrid IoT-Fog-Cloud framework. In this framework, we deploy a geo-distributed fog layer at the edge of networks. The fogs can provide the local storage, sufficient processing power, and appropriate network functions. For the fog-based healthcare system, data confidentiality, access control, and secure searching over ciphertext are the key issues in sensitive data. Furthermore, how to adjust the storage and computing requirements to meet the limited resource is also a great challenge for data management. To address these, we design a lightweight keyword searchable encryption scheme with fine-grained access control for our proposed healthcare related IoT-Fog-Cloud framework. Through our design, the users can achieve a fast and efficient service by delegating a majority part of the workloads and storage requirements to fogs and the cloud without extra privacy leakage. We prove our scheme satisfies the security requirements and demonstrate the excellent efficiency through experimental evaluation.

1. Introduction

Since Ashton [1] and Brock [2] firstly proposed the concept of IoT, it has been widely used in real life by combining with technologies in sensor networks, embedded system, object identifications, and wireless networks in order to tag, sense, and control things over the Internet [3–6]. With the ubiquitous nature of IoT, it makes great contribution in improving the equality of medical care by empowering remote monitoring and reducing time cost through implanting sensors or wearing mobile devices. According to the insight from [7], the healthcare system will evolve into a home-centered paradigm in 2030 from the current hospital-centered one. As more sensors are deployed in the healthcare system, the seamless data needs to be stored, processed, and transmitted. This may cause a great challenge to the traditional IoT-cloud infrastructure from the aspects of reliability, immediate response, and security [8]. This calls demand for a “mediator” between IoT devices and cloud server to support geo-distribution, storage, and computing capability, acting as an extension of the cloud, which is officially called

fog from the concept of fog computing proposed by Cisco [9].

When storing sensitive data like personal health records to cloud servers, the security and privacy of these data are still challenges in the fog computing paradigm [10–12]. To solve this problem, applying access control mechanism is an essential method to protect the sensitive data from unauthorized users. As a new type of IBE proposed by [13], attribute-based encryption (ABE) plays a great role in access control, which is classified into the key-policy attribute-based encryption (KP-ABE) and the ciphertext-policy attribute-based encryption (CP-ABE). KP-ABE associates user's private keys with the designated policies and tags ciphertexts with attributes, while CP-ABE is related to ciphertexts with the designated policies and identifies the user's private key with attributes [14, 15]. Obviously, CP-ABE is a better choice to execute access control in our model since it is the user's ability to designate an access structure and process the encryption operation under the structure.

However, most existing ABE schemes are time consuming in the key generation phase and have a large computational

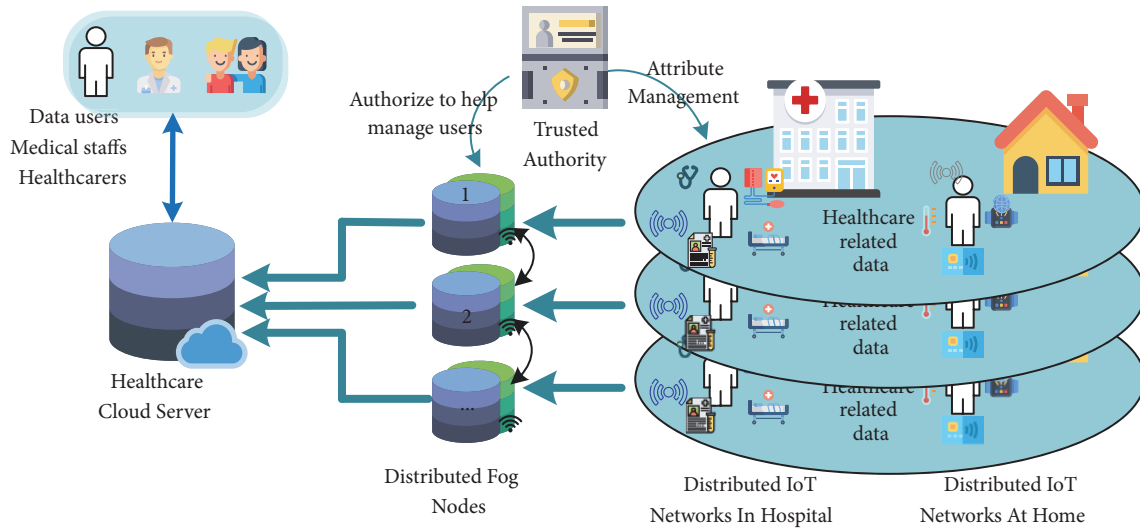


FIGURE 1: A Fog-based healthcare system.

load in the decryption phase, which leads to suffering a bad experience for users. Also, how to maintain effective search in the encrypted ciphertext is a great challenge. Searchable encryption especially searchable public encryption is an effective approach to solve the above problem. And it is important to reduce complex operations, e.g., pairing and exponential operations, for users in the searchable public encryption.

1.1. Motivation and Contribution. The IoT infrastructure, such as the monitoring devices in a traditional hospital or health management wearable devices in a smart home, continuously synchronizes data to the remote cloud. The massive sensitive data leads to a great challenge to the current healthcare-related IoT-to-cloud system due to the nature of IoT's limited storage, low power, and poor computability. In this paper, we attempt to solve the problem above as follows:

- (i) *We propose a fog-supported hybrid infrastructure as shown in Figure 1.* The distributed fogs are deployed between IoT devices and clouds, providing temporary data storage, data computation and analysis, and network services [16], so as to reduce transmission delay. Also, they help to manage users and attributes under the control of trusted authority.

With the proposed infrastructure above, we design a new scheme to implement some specific network functions to meet real world needs. We will show it by exhibiting an example as follows. A person named Wealth rarely cares about his physical condition. One day he knows his friend Bob is suffering from hyperglycemia, and then he wants to learn about it. When he searches "Hyperglycemia" in cloud service providers such as "BodyMedia", "Google Health", "CiscoHealthPresence", or "IBM Bluemix", clouds know that he or someone he knows may get hyperglycemia. Obviously, his personal health privacy is exposed to the clouds. In order to prevent privacy disclosure, we construct indexes for "Hyperglycemia" in the file encryption phase through

some secure methods. To search such a keyword, we need to generate the corresponding trapdoors with the help of the fog. Upon receiving the trapdoor, clouds return all the encrypted files associated with the specific "Hyperglycemia" if the trapdoor matches with the index. We can protect Wealth's searching privacy by this way as follows.

Further, we consider Wealth receives all files through searching "Hyperglycemia" by performing our designs. After realizing the importance of keeping healthy, he decides to start his own fitness program to monitor his health indicators such as Glycemic index through wearable sensors. Also, due to the limited storage of his own devices, he has to store his data to the cloud and shares it to some designated ones which have specific attributes. If someone without sufficient attributes attempts to search the keyword, he/she is impossible to generate a valid trapdoor matching with a keyword's index, not to mention to get Wealth's sensitive data. We help Wealth to accomplish this goal through the following designs.

In summary, Wealth could enjoy an efficient, fast, high-quality, and secure service through adopting our system.

The main contributions of this article are exhibited as follows:

- (i) *We design a keyword searchable encryption scheme in the healthcare related IoT-fog-cloud infrastructure.* The proposed scheme ensures a security requirement that both data and keywords are protected from the cloud and the fog, which is very essential to users in the health related environment.
- (ii) *With the restriction of constrained resource, IoT devices are not capable of doing complicated encryption and decryption process. In order to overcome this issue, we transfer most of heavy computation to the fog and the cloud in our scheme, while only a small part is reserved for users.*
- (iii) *On the basis of ciphertext-policy attribute-based encryption, we design a fine-grained access control*

framework. A user should obtain his query capability authorization from a trusted authority and the fog through checking his attributes. The messages are encrypted with an access policy such that *only users with the designated attributes can access them*.

- (iv) We provide formal security analysis which demonstrates that our scheme is *secure under IND-CK-CCA attack and satisfies trapdoor indistinguishability secure*. Also we make experiment comparisons with some previous research revealing that our scheme has a good efficiency.

The rest of the paper is organized as follows. In Section 2, we briefly introduce preliminaries which will be utilized in our paper. Next, in Section 3, we present two adversary models, security requirements and system functions of our lightweight fine-grained searchable encryption (LFSE) system. Our proposed system is described in Section 4. The thorough security analysis of the proposed system appears in Section 5 and the efficiency is analysed in Section 6. We conclude our paper in Section 8.

2. Preliminaries

In this section we provide a detailed description of some fundamentals of cryptography that will be used throughout this paper.

2.1. The Notations. In this section, we first give notation descriptions that will be used throughout this paper. For a prime number p , we denote the set $\{1, 2, \dots, p-1\}$ as \mathbb{Z}_p^* , where multiplication and addition modulo p are defined in the set. We use $a \leftarrow_r S$ to denote that a is uniformly chosen from all elements in S randomly. And let λ be the security parameter of our system.

2.2. Bilinear Map. \mathbb{G}_1 and \mathbb{G}_2 are two multiplicative cyclic groups of prime order p . Let g be a generator of \mathbb{G}_1 and e be a bilinear map, $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$. The bilinear map e has the following properties:

- (i) Bilinear: A map $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is bilinear if $e(aP, bQ) = e(P, Q)^{ab}$ for all $P, Q \in \mathbb{G}_1$ and all $a, b \in \mathbb{Z}_p^*$.
- (ii) Nondegenerate: $e(g, g) \neq 1$.
- (iii) Computable: There is an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in \mathbb{G}_1$.

2.3. Access Policy. An access policy defines attribute sets that are acquired to get access to private messages.

Definition 1 (monotonicity). Letting \mathcal{AT} be attributes universe, then an access policy $\mathbb{A} \subseteq 2^{\mathcal{AT}}$ means \mathbb{A} is a collection of non-empty subsets of \mathcal{AT} . We call the access policy \mathbb{A} is monotone if $\forall \Omega_1, \Omega_2 \subseteq \mathcal{AT}$ s.t.

$$\Omega_1 \subseteq \Omega_2, \quad \Omega_1 \in \mathbb{A} \implies \Omega_2 \in \mathbb{A}. \quad (1)$$

According to the monotonicity, an authorized user cannot lose his privileges if he has more attributes than required.

3. System Model

3.1. Architecture of System. The architecture of the proposed fog-based healthcare system is shown in Figure 2. It is composed of four parts, i.e., a trusted authority, cloud server providers, fog nodes, and data users (including data owners and other users).

Trusted Authority (TA). A trusted authority, such as the national health center or an entity authorized by it, is an important authority for verifying users attributes. It takes charge of generating system parameters for all entities. And it is responsible for issue, revoke, and update attribute private keys for users.

Cloud (Short for Cloud Server Providers). The cloud such as Amazon provides data storage, computational resource services, and data analysis. Apart from providing content service above, it also takes charge of the access services from the outside users to the encrypted files. We assume that the public cloud executes the searchable algorithm honestly. The cloud in our system is responsible for performing test algorithm and accomplish a part of decryption task with knowing any information about the user's keys or attributes.

Fog (Short for Fog Nodes). Fog, providing abilities of computing, storage, and mobility, is deployed at edges of networks. Because of the limited computing resources and the restricted capacity of the data owner or user's facility carried nearby, it is responsible for deploying a half-trusted fog as interface between a user and the cloud server, especially in situations with sensitive medical information. Fog in our system takes charge of managing users within its coverage, revoking users and attributes without having any information about their private keys. Further, it helps controlling users' query action through generating one part of the trapdoor without knowing the queried keyword.

Data Owner. The data owner is an entity who intends to share his files with designated receivers. The receivers' attributes should satisfy the access policy embedded in the corresponding ciphertext. It is in charge of file encryption with a specific access policy, index generation for all the keywords, and uploading to the cloud.

Data User. The data user is the entity who intends to get the encrypted files by sending a query request to cloud servers and the fog. If he has enough attributes satisfying with the required access policy, he is able to download ciphertexts and decrypt them with the help from the cloud. It takes charge of keyword selection to generate trapdoors and then ciphertext decryption.

Assumptions. We assume that the cloud and the fog are always online. They have sufficient storage capacity and computing resource. Also we assume that there exists a secure channel between data owner/user and the fog node, e.g., secure Wi-Fi networks.

We assume that the cloud and fogs are all "honest but curious" [21]. To be specific, they do not delete or modify

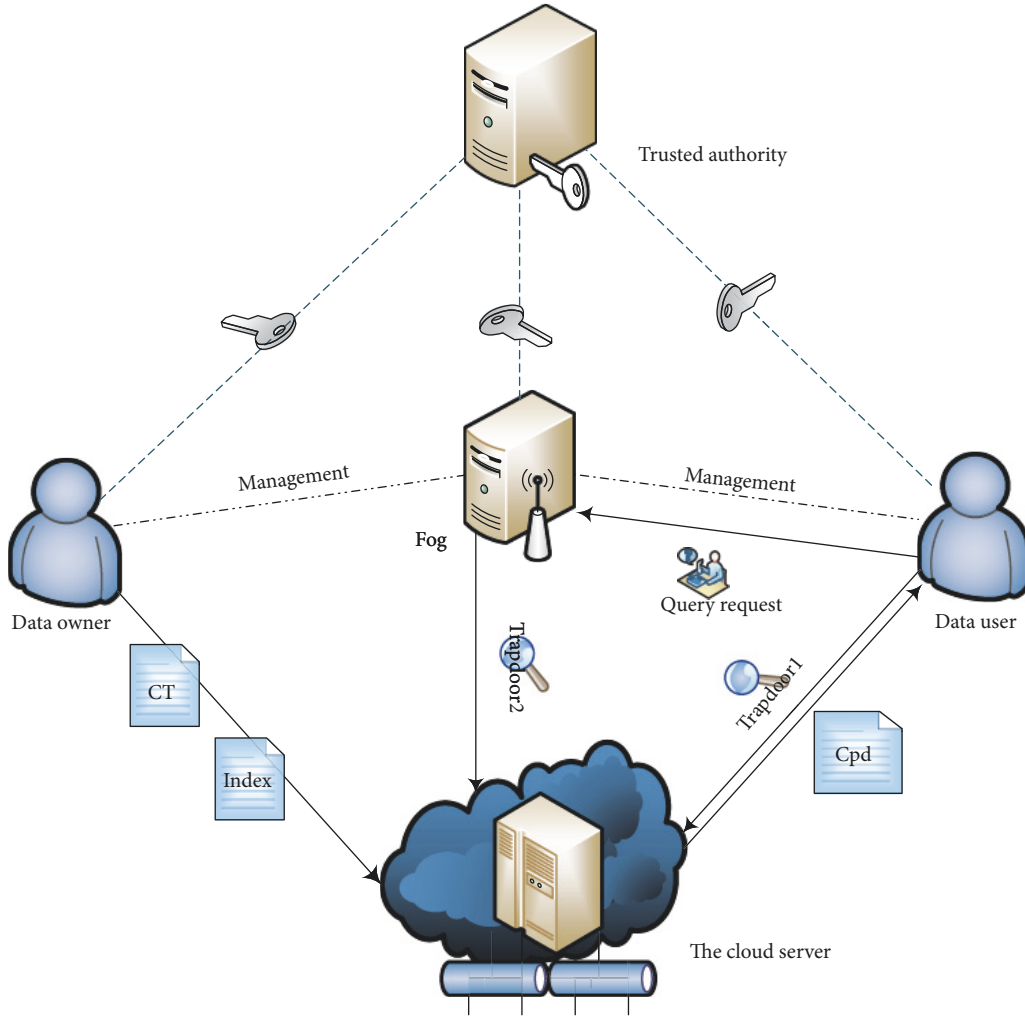


FIGURE 2: System model.

user's data and return the computing results honestly but attempt to access as much private information as possible. All the entities execute our proposed protocol and users try to access data either within or out of their privileges. And it is assumed that the cloud and the fog do not collude with each other.

Different from most existing work with only public cloud, it is a novel cloud-fog architecture. In this work, we assume that files and keywords are sensitive and should be protected from both the cloud and the fog. And attributes are semisensitive, which means attributes can only be known by the fog.

3.2. Definition of Basic Algorithms. We describe a general definition for our lightweight fine-grained searchable encryption scheme, consisting of several polynomial time algorithms.

Setup. This phase containing three subalgorithms is implemented by TA.

System.Setup (1^λ): Input the security parameter λ ; then the algorithm outputs the master key Mk , public key Pk , and other system parameters.

Fog.Setup (Pk): Input the system parameter Pk ; then the algorithm outputs the fog's public and private key pair (Pk_{F_k}, Sk_{F_k}) and the corresponding verification key vk_j for each attribute in the attribute universe.

User.Setup (Pk): For each user requesting to join the system, TA verifies the user identity and his attributes.

KeyGeneration. This phase is executed by TA, which contains two subalgorithms.

KeyGen ($Pk, Mk, User_i, \Omega_{ui}$): Input the system's keys (Pk, Sk) , the user identity, and the user's attributes; then the algorithm outputs the user's public and private key (Pk_{ui}, Sk_{ui}) . Next, Input the output of private key and user's attributes Ω_{ui} ; the algorithm outputs the secret verification key svk_j for each attribute $attr_j \in \Omega_{ui}$.

SearchKeyGen ($Mk, User_i$): Input the system's master key Mk and user's identity $User_i$; then the algorithm returns the search key \mathcal{S}_i for the user.

FogSupport. This phase is executed by the fog and users which is under the management of the fog. Three algorithms are included in this phase.

Adduser ($User_i$): Input the public parameter and the user's identity $User_i$; then the algorithm outputs a table \mathcal{T}_{user} for the fog F_k to store users' information.

ReKeyGen (Sk_{ui}, svk_j): Input the user's private key Sk_{ui} and the private verification key svk_j for $at_j \in \Omega_{ui}$, then the algorithm outputs a secret key csk_{ui} .

ReEnc (Pk_{ui}, vk_j): Input the user's public key Pk_{ui} and the verification key vk_j for $at_j \in \Omega_{ui}$, then the algorithm outputs a ciphertext cvk_{ui} .

FileEncryption. This phase is performed by the user.

Enc (F, \mathbb{A}, vk_j): Input a file F , an access policy \mathbb{A} and the verification key vk_j ; then the algorithm outputs the ciphertext C embedded with the access policy.

IndexGeneration. This phase is implemented by the user through running the algorithm *Index*.

Index ($W, \mathcal{S}_i, Pk_{ui}$): Input the user's search key \mathcal{S}_i and the keyword W ; then the algorithm outputs an index I_W for the keyword.

TrapdoorGeneration. This phase is executed by the fog and the user, including two subalgorithms.

Trapdoor (Pk_{F_k}, csk_{ui}): It is performed by the fog. Input the fog's public key Pk_{F_k} and the user's regenerated key csk_{ui} as input; then the algorithm outputs T_f that is a part of the trapdoor T .

Trapdoor2 (W, \mathcal{S}_i): This algorithm is executed by the user. Input the user's search key \mathcal{S}_i and a keyword W , then the algorithm outputs T_W that is the other part of the trapdoor T .

Test. This phase is implemented by the cloud server through running *Test*.

Test (I_W, T): Input the keyword index I_W and the trapdoor T ; then the algorithm outputs 0 if they do not match; otherwise it outputs 1.

FileDecryption. The decryption phase is implemented by the cloud server and the user, consisting two subalgorithms.

Dec (C, cvk_{ui}): Input the file's ciphertext C , the trapdoor T , and the ciphertext of user's attributes ciphertext cvk_{ui} ; then the algorithm outputs C_{pd} that is a part-decrypted version of the ciphertext.

Dec2 (C_{pd}, Sk_{ui}): Input the user's private key Sk_{ui} and the part-decrypted ciphertext C_{pd} , then the algorithm outputs the file F .

3.3. Security Requirements

- (1) Data confidentiality: The cloud and the fog are not allowed to know the encrypted data files. Unauthorized users who have no appropriate attributes matching the policy embedded in the ciphertext should not learn the content of the underlying plaintext.
- (2) Keyword privacy: The keywords should be protected from both the cloud and the fog in a secure way, such as by using a oneway hash function. The cloud server is able to perform the test operation over the

indexes but leaks no information about keywords to any unauthorized attackers.

- (3) Trapdoor privacy: One part of the trapdoor is generated by the data user by using the search key and the secret verification key for his attributes together with the keyword. The other part is generated with the help of the fog using the user's re-encrypted key. The trapdoor reveals no information about the corresponding keyword or the user's attributes to the attacker.

3.4. Adversary Model. To achieve the security requirements, we design two security models for our scheme. Firstly, we introduce a fundamental assumption in Definition 2.

Definition 2 (DBDH assumption). We say that the DBDH assumption holds if no polynomial time algorithm has a nonnegligible advantage in solving the DBDH problem.

According to the security parameter, let a group \mathbb{G}_1 of prime order p have a generator g . $a, b, c \leftarrow_{\mathcal{R}} \mathbb{Z}_p^*$ are chosen randomly. The DBDH problem states that the adversary should distinguish $e(g, g)^{abc} \in \mathbb{G}_2$ from a random element $V \in \mathbb{G}_2$ when given $g, g^a, g^b, g^c \in \mathbb{G}_1$.

Definition 3. Our LFSE scheme is trapdoor indistinguishable secure if there is no polynomial time attack can have a nonnegligible advantage in the following game.

The security model is defined as *Game 1* played between an adversary \mathcal{A} and an algorithm \mathcal{B} .

Game 1 (Trapdoor privacy). Setup: With a security parameter λ , the algorithm \mathcal{B} outputs system parameters and generates the public key Pk_{ui} , the private key Sk_{ui} , and the search key \mathcal{S}_i for the data user.

Query phase 1: The adversary \mathcal{A} adaptively makes the following queries.

\mathcal{O} .Trapdoor1: The adversary \mathcal{A} could query any keyword's one part ($T_{f_0}, T_{f_1}, T_{f_2}$) of the trapdoor.

\mathcal{O} .Trapdoor2: The adversary \mathcal{A} could query the keyword's another part (T_{W_1}, T_{W_2}) of the trapdoor.

Challenge phase: The adversary \mathcal{A} sends two keywords W_1^* and W_0^* with equal length. Then \mathcal{B} will randomly select $x \in \{0, 1\}$ and construct the trapdoor $T_f\{W_x^*\}$ for the keyword W_x^* and send it to the adversary \mathcal{A} .

Query phase 2: The adversary \mathcal{A} queries the same as phase 1 with the restriction the queried keyword $W \notin \{W_0^*, W_1^*\}$.

Guess: The adversary \mathcal{A} outputs a guess $x' \in \{0, 1\}$. If $x = x'$, \mathcal{A} wins the game and the algorithm \mathcal{B} outputs 1; otherwise \mathcal{A} fails and \mathcal{B} outputs 0.

Definition 4. Our LFSE scheme is IND-CKCCA secure if there is no polynomial time attack can have a nonnegligible advantage in the following game.

We define the indistinguishable against chosen keyword chosen ciphertext attack in our system. The security model is

defined through *Game 2* played between an adversary \mathcal{A} and a challenger \mathcal{C} as follows.

Game 2 (Ciphertext and Keyword privacy). This Initial Phase. The adversary \mathcal{A} commits to challenge \mathcal{C} .

Setup: The challenger \mathcal{C} selects a large security parameter λ and runs the setup algorithm to obtain the system master key and public key (Mk, Pk) . \mathcal{C} gives Pk to \mathcal{A} and keeps Mk .

Phase 1: The adversary \mathcal{A} makes the following queries with a polynomial number bound.

- (i) $\mathcal{O}.$ KeyGen: The oracle contains several key generation oracles executed by the challenger \mathcal{C} to generate a series of keys for \mathcal{A} .
- (ii) $\mathcal{O}.$ Trapdoor: The oracle contains two trapdoor generation oracles executed by the challenger \mathcal{C} to generate the trapdoor $T = (T_f, T_w)$ for \mathcal{A} , with the keys generated from the above steps.

Challenge: After finishing phase 1, the adversary \mathcal{A} outputs two messages m_0^*, m_1^* and two keywords W_0^*, W_1^* both with equal length to be challenged. The challenger \mathcal{C} flips a coin to choose $b_1, b_2 \in \{0, 1\}$ and then constructs ciphertext for $m_{b_1}^*$ and index for $W_{b_2}^*$. Finally, the challenger \mathcal{C} sends them to the adversary \mathcal{A} .

Phase 2: The adversary \mathcal{A} adaptively makes queries the same as phase 1, expect the restrictions that $W \notin \{W_0^*, W_1^*\}$ and the user's private key cannot be queried.

Guess: The adversary \mathcal{A} outputs guesses $b'_1, b'_2 \in \{0, 1\}$. If $b'_1 = b_1$ and $b'_2 = b_2$, \mathcal{A} wins the game.

The adversary \mathcal{A} has an advantage of $\epsilon^{Adv_{\mathcal{A}}^{LFSE}}(\lambda) = |\Pr[b'_1 = b_1, b'_2 = b_2] - 1/2|$ in breaking the DBDH assumption.

3.5. System Functions. Considering the performance-related issues, our scheme are designed to achieve the following functions.

- (1) Fine-grained access control: A data owner embeds an access policy into each file to be transmitted to the cloud. This guarantees that the data is only accessed by users with appropriate attributes and well prevented from the cloud server.
- (2) Authorization: Each data user who is authorized by the trusted attribute authority can be assigned his individual private key. These private keys can be used to search and decrypt files in our system.
- (3) Search on keywords: An authorized user can generate a query request for some keywords by using his individual private key. After the cloud server receives the query and performs the "Test" on the encrypted files, the user can obtain the matched files.
- (4) Revocability: The trusted authority should be able to revoke an user and attributes. If an authorized user is revoked, the user is no longer able to search and read files in our system. If an attribute of the user is revoked, the user is no longer able to access the files embedded with an access policy containing the attribute.

4. LFSE Scheme

4.1. Construction of LFSE Scheme. We specify the proposed LFSE scheme in fog-based healthcare system in details. In real world, we consider that all the sensors carried by the owner are continually collecting and reporting data, and the owner decides whether and when data is transmitted to the cloud.

- (1) **System setup:** Let λ be the security parameter, and then TA performs the following steps. Firstly, it chooses two cyclic groups (G, \cdot) and (G_T, \cdot) with prime order p and defines a bilinear pairing $e : G \times G \rightarrow G_T$. Let g be a generator of G , g_1, g_2 and s, v are randomly chosen from G and \mathbb{Z}_p^* , respectively. Then, it computes $g' = g^s$, $\mathcal{V} = e(g, g)^v$, and selects two hash functions: $H : \{0, 1\}^* \rightarrow \mathbb{Z}_p^*$, $H_1 : \mathbb{Z}_p^* \times \{0, 1\}^* \rightarrow \mathbb{Z}_p^*$. Ultimately, TA keeps (s, v) secret as master key Sk and publishes system parameters $Pk = \{\lambda, G, G_T, e, g, g_1, g_2, g', \mathcal{V}\}$. Afterwards, TA will initialize the attribute universe $\mathcal{AT} = \{at_1, at_2, \dots, at_m\}$ and the monotone access structure \mathbb{A} . Let $\mathbb{A}_0 = (\Omega_1, \Omega_2, \dots, \Omega_n)$ be a basis for \mathbb{A} , where each Ω_i is a minimal authorized attribute set in \mathbb{A} .
- (2) **Setup and key generation for fogs:** For each fog, TA generates its public and private keys (Pk_{Fog_k}, Sk_{Fog_k}) by running *Fog.Setup*. The algorithm picks $c_k \leftarrow_r \mathbb{Z}_p^*$ randomly and outputs $(Pk_{Fog_k}, Sk_{Fog_k}) = (c_k, g^{c_k})$. The fog maintains the private key sent from TA and initializes a table \mathcal{T}_{user} to manage all the authorized users within its coverage. Further, to authorize fogs to manage attributes, for each fog Fog_k , TA selects a $\sigma_k \leftarrow_r \mathbb{Z}_p^*$ and then computes $\theta_j = H_1(\sigma_k, at_j)$, $d_{1j} = g^{\theta_j}$, and $d_{2j} = \mathcal{V}^{\theta_j}$ for each attribute $at_j \in \mathcal{AT}$ and defines $vk_j = (d_{1j}, d_{2j})$ as a verification key. Then TA sends verification keys $\{vk_j\}_{at_j \in \mathcal{AT}}$ to the corresponding fogs as attributes information. The fogs exchange their users and attributes verification keys information to allow the authorized user to connect to our system when he moves to other fog's managing area.
- (3) **Key Generation for the user:** Assume that a new user $User_i$ with the attribute list $\Omega_{u_i} = \{\{at_j\}, j \leq m\}$ requests to join the system. First of all, TA authenticates the user's identity and his attributes. Then it returns the public and private key $(Pk_{User_i}, Sk_{User_i}) = (g^{\alpha_i}, (t_0, t_1, t_2))$ to each user, where $t_0 = g^v g_1^{\alpha_i}$, $t_1 = g^{\beta_i}$, $t_2 = \delta$, and $\alpha_i, \beta_i, \delta \leftarrow_r \mathbb{Z}_p^*$. Simultaneously, TA computes $svk_j = (Pk_{User_i})^{\theta_j} = g^{\alpha_i \theta_j}$ for each $at_j \in \Omega_{u_i}$ and returns it to the user as a secret verification key for each attribute obtained by the user. Once the phase is finished, the fog adds $User_i$ to table \mathcal{T}_{user} as a new authenticated user's information.
- (4) **Search Key Generation:** After receiving the public and private keys from TA, the user $User_i$ also needs to make a request to get a private key for searching on keywords. The user picks $\eta \leftarrow_r \mathbb{Z}_p^*$ randomly and

sends $g_1^{1/\eta}$ to TA. Then, TA computes the searching key $\mathcal{S}_i = (g_1^{1/\eta})^s g_2^{\beta_i \delta}$ and sends \mathcal{S}_i to the user.

- (5) Prepare For Fog Support: Due to the limited processing power and low computing efficiency of the user, we would like to transfer most of the computational load to the fog and cloud without leaking additional information. In our system, the user would delegate the fog to complete a part of auxiliary computation by transferring a converted secret key csk_{ui} . The user computes $T_0 = t_0^{t_2} = g^{v\delta} g_1^{\alpha_i \delta}$, $T_1 = t_1^{t_2} = g^{\beta_i \delta}$, and $T_{2j} = (svk_j)^{t_2} = g^{\alpha_i \theta_j \delta}$ and then sends $csk_{ui} = (T_0, T_1, \{T_{2j}\}_{at_j \in \Omega_{ui}})$ to the fog. With csk_{ui} , the fog can help users accomplish a part of computation tasks without knowing the user private key. For facilitating further calculations, the user can compute $V_1 = e(g, g)$ and $V_2 = e(g_1, g')$ in advance and stores them. Simultaneously, to ensure the cloud can help users do a part of computation and avoid the cloud from obtaining information from the user's attributes, the fog selects $s' \leftarrow_r \mathbb{Z}_p^*$ randomly and computes $D_1 = g_1^{s'}$, $D_{2j} = (d_{1j})^{s'} = (g^{\theta_j})^{s'} = g^{\theta_j s'}$ and sends $cvk_{ui} = (D_1, \{D_{2j}\}_{at_j \in \Omega_{ui}})$ to the cloud and the secret s' to the user through a secure channel.

- (6) Encrypt: Suppose that the data owner decides his file F . This file can be searched and acquired by users whose attributes satisfy with an access policy \mathbb{A} . Under this assumption, the user can designate different types of data to be accessed by different kind of people. For the monotone access policy \mathbb{A} , there exists a basis $\mathbb{A}_0 = (\Omega_1, \Omega_2, \dots, \Omega_n)$, where each Ω_i , a minimal set, is composed of the authorized attributes. To encrypt the file, the user picks $s_l \leftarrow_r \mathbb{Z}_p^*$ for each $1 \leq l \leq n$ computes

$$C_l = (C_{1l}, C_{2l}) = \left(F \cdot \left(\prod_{at_j \in \Omega_l} d_{2j} \right)^{s_l}, \frac{s_l}{s'} \right), \quad (2)$$

The user keeps the ciphertext as $C = (\mathbb{A}, \{C_l\}_{1 \leq l \leq n})$ embedded with the access policy \mathbb{A} .

- (7) Index: For a continuous health monitoring system, data are constantly processed and transferred to the cloud from various kind of sensors. In order to get quick access to useful files from the super large data center, we add different keywords to files.

We assume the file F contains a set of keywords \mathcal{W} which are extracted from the original health file. For each keyword $W \in \mathcal{W}$, the user picks $u \leftarrow_r \mathbb{Z}_p^*$ randomly and computes $I_W = (C_{W1}, C_{W2}, C_{W3}) = ((e(g, g)^{H(W)} e(g_1, g')^u), g^u, g_2^u)$. Subsequently, the user sends the ciphertext C together with the index I_W to the cloud. Then the cloud stores them.

- (8) Trapdoor: Generally speaking, the *Trapdoor* algorithm is used to generate a trapdoor for a certain keyword by the user who wants to search files containing

this keyword. In our system, to help the user reduce the computing burden, we delegate the fog to do a part of the trapdoor generation work without leaking any information about the queried keywords. This design has an advantage in our IoT system: confidentiality of keywords. Specifically, upon receiving the query request from the user $User_i$, the fog firstly searches the user's identity in table \mathcal{T}_{user} . If the fog does not find it in the table which means the user did not join the system, then the fog refuses to generate the part trapdoor for the user and returns a warning message. This process completed by the fog ensures that any external user who is not authenticated cannot search any keyword and guarantees no leakage of any information about keywords or encrypted files. If the fog finds the user in table \mathcal{T}_{user} , the fog randomly chooses $\rho \leftarrow_r \mathbb{Z}_p^*$, sends it to the user through a secure channel, and then computes

$$\begin{aligned} T_{f_0} &= T_0^\rho = t_0^{t_2 \rho} = g^{v\delta \rho} g_1^{\alpha_i \delta \rho}, \\ T_{f_1} &= T_1^\rho = t_1^{t_2 \rho} = g^{\beta_i \delta \rho}, \\ T_{f_{2j}} &= T_{2j}^\rho = svk_j^{t_2 \rho} = g^{\alpha_i \theta_j \delta \rho}. \end{aligned} \quad (3)$$

After finishing all the above steps, the fog uploads $T_f = (T_{f_0}, T_{f_1}, T_{f_{2j}})_{at_j \in \Omega_{ui}}$ to the cloud as a part of the trapdoor. To search files with a keyword W' , the user firstly chooses $\eta \leftarrow_r \mathbb{Z}_p^*$ and computes

$$\begin{aligned} T_{W1} &= g^{H(W')} \mathcal{S}_i^\eta = g^{H(W')} g_1^s g_2^{\beta_i \delta \eta}, \\ T_{W2} &= \frac{\eta}{\rho} \end{aligned} \quad (4)$$

with his own search key. Then, the user sends the other part of the trapdoor $T_W = (T_{W1}, T_{W2})$ to the cloud. Ultimately, the cloud gets a full trapdoor $T = (T_f, T_W)$. In this phase, if the fog has verified one user's identity and verification keys for his attributes, the fog can perform the trapdoor generation once in a while. This is available as this phase is not related to the queried keyword. As a result, the computing burden for the fog and interaction time for both the user and the fog can be reduced.

- (9) Test: Upon receiving the search request for keyword W' from the fog and the user, the cloud runs *Test* algorithm for all items which are encrypted indexes for all the keywords by computing

$$\frac{e(C_{W2}, T_{W1})}{e(C_{W3}, T_{f_1}^{T_{W2}})}. \quad (5)$$

The cloud compares the result with C_{W1} , if it equals C_{W1} , the cloud outputs 1, and performs the next step. Otherwise, the cloud outputs 0, returns a warning message, and exits the system.

(10) Decryption: If the algorithm *Test* cannot find an index for the uploaded trapdoor, the cloud would not run the *Dec1* algorithm and returns \perp . Otherwise, the cloud computes

$$C_{pd} = \left[\frac{e(D_1, \prod_{at'_j \in \Omega'_{ui}} T_{f2j})}{e(\prod_{at'_j \in \Omega'_{ui}} D_{2j}, T_{f0})} \right]^{C_{2l} T_{W2}}. \quad (6)$$

Once upon receiving the part-decrypted ciphertext C_{pd} from the cloud, the user recovers the file F by using his own private key through computing

$$F = C_{1l} C_{pd}^{1/(\delta t_2)}. \quad (7)$$

Obviously, the user only needs to do an exponential operation in the decryption, which is a great step in improving efficiency.

4.2. Consistency. Firstly, we present that the trapdoor matching is valid in our system.

$$\begin{aligned} \frac{e(C_{W2}, T_{W1})}{e(C_{W3}, T_{f1}^{T_{W2}})} &= \frac{e(g^u, g^{H(W')}) \left((g_1^{1/\eta})^s g_2^{\beta_i \delta} \right)^\eta}{e(g_2^u, (g^{\beta_i \delta \rho})^{\eta/\rho})} \\ &= \frac{e(g^u, g^{H(W')}) \cdot e(g^u, g_1^s) \cdot e(g^u, g_2^{\beta_i \delta \eta})}{e(g_2^u, g^{\beta_i \delta \eta})} \\ &= \frac{e(g, g)^{uH(W')} \cdot e(g, g_1)^{us} \cdot e(g, g_2)^{u\beta_i \delta \eta}}{e(g_2, g)^{u\beta_i \delta \eta}} \\ &= e(g, g)^{uH(W')} \cdot e(g, g_1)^{us} \\ &= e(g, g)^{uH(W')} \cdot e(g_1, g')^u. \end{aligned} \quad (8)$$

If there exists a keyword $W \in \mathcal{W}$ matching with the queried keyword which leads $H(W) = H(W')$, we can derive the conclusion $e(C_{W2}, T_{W1})/e(C_{W3}, T_{f1}^{T_{W2}}) = C_{W1}$.

Then, the file recovery can be maintained as the following two steps. If the test passes, the cloud decrypts all the related files by computing

$$\begin{aligned} C_{pd} &= \left[\frac{e(D_1, \prod_{at'_j \in \Omega'_{ui}} T_{f2j})}{e(\prod_{at'_j \in \Omega'_{ui}} D_{2j}, T_{f0})} \right]^{C_{2l} T_{W2}} \\ &= \left[\frac{e(g_1^{s'}, \prod_{at'_j \in \Omega'_{ui}} g^{\alpha_i \theta_j \delta \rho})}{e(\prod_{at'_j \in \Omega'_{ui}} g^{\theta_j s'}, (g^v g_1^{\alpha_i})^{\delta \rho})} \right]^{(s_1/s')(\eta/\rho)} \\ &= \left[\frac{e(g_1^{s'}, g^{\alpha_i \delta \rho \sum_{at'_j \in \Omega'_{ui}} \theta_j})}{e(g^{s' \sum_{at'_j \in \Omega'_{ui}} \theta_j}, g^{v \delta \rho}) \cdot e(g^{s' \sum_{at'_j \in \Omega'_{ui}} \theta_j}, g_1^{\alpha_i \delta \rho})} \right]^{(s_1/s')(\eta/\rho)} \end{aligned}$$

$$\begin{aligned} &= \left[\frac{e(g_1, g)^{s' \alpha_i \delta \rho \sum_{at'_j \in \Omega'_{ui}} \theta_j}}{e(g, g)^{s' v \delta \rho \sum_{at'_j \in \Omega'_{ui}} \theta_j} \cdot e(g, g_1)^{s' \alpha_i \delta \rho \sum_{at'_j \in \Omega'_{ui}} \theta_j}} \right]^{(s_1/s')(\eta/\rho)} \\ &= \frac{1}{(e(g, g)^{v \delta \sum_{at'_j \in \Omega'_{ui}} \theta_j})^{s_1/\eta}} = \frac{1}{e(g, g)^{v \delta s_1 \eta \sum_{at'_j \in \Omega'_{ui}} \theta_j}}. \end{aligned} \quad (9)$$

If the user's attributes Ω'_{ui} satisfy the access policy \mathbb{A} , we know there exists a basis $\mathbb{A}'_0 = (\Omega'_1, \Omega'_2, \dots, \Omega'_n)$ s.t.

$$\begin{aligned} \forall at_j \in \Omega'_{ui}, \\ \exists \Omega'_i \text{ s.t. } at_j \in \Omega'_i \subseteq \Omega'_{ui}, \end{aligned} \quad (10)$$

and we have $\sum_{at'_j \in \Omega'_{ui}} \theta_j = \sum_{at'_j \in \Omega'_i} \theta_j = \sum_1^n \theta_j = \sum_{at'_j \in \Omega'_i} \theta_j$.

According to this, the user finally recovers the file by computing

$$\begin{aligned} F &= C_{1l} C_{pd}^{1/(\delta t_2)} \\ &= \left(F \cdot \left(\prod_{at_j \in \Omega_i} d_{2j} \right)^{s_1} \right) \left(\frac{1}{e(g, g)^{v \delta s_1 \eta \sum_{at'_j \in \Omega'_{ui}} \theta_j}} \right)^{1/\delta \eta} \\ &= F \cdot e(g, g)^{v s_1 \sum_{at_j \in \Omega_i} \theta_j} \cdot \frac{1}{e(g, g)^{v s_1 \sum_{at'_j \in \Omega'_{ui}} \theta_j}} = F. \end{aligned} \quad (11)$$

4.3. User Revocation and Attribute Revocation. As mentioned above, the fog is an access interface between the cloud and users. The table \mathcal{T}_{user} is a certification to verify whether a user is in the system. The revocation of a user can be realized through rejecting the query request. To be specific, once a user submits a revocation request to the trusted authority or the trusted authority decides to revoke a user, the trusted authority deletes all keys and attributes information of the user. Then it sends the user's revocation information to the fog, and all the information about the user will be deleted in \mathcal{T}_{user} . As a result, the user cannot update his/her request to the cloud server. Furthermore, once the re-encrypted keys csk_{ui} and cvk_{ui} are revoked from the fog, the user cannot generate trapdoors for any keywords. Because the fog needs csk_{ui} and cvk_{ui} to do a part of computation to accomplish the trapdoor generation phase, the loss of csk_{ui} and cvk_{ui} leads to the user's failure to search for any files. As a result, such a user is new to the system and the fog will no longer respond to its any request.

In our system, we can achieve attribute revocation with the designation of csk_{ui} and cvk_{ui} . Once an attribute is revoked, the data owner could keep the data from the group of users who have the revoked attribute. To be specific, upon deciding to revoke an attribute at_j , the fog destroys the attribute's verification key vk_j and deletes csk_{ui} and cvk_{ui} for users containing the attribute, then sends a warning message to these users to update the related csk_{ui} and cvk_{ui} . Before users updating csk_{ui} and cvk_{ui} , the fog refuses to generate trapdoor for them, which directly leading to the failure of accessing files in the system. Although it may cause some

computational loads and transmission cost, it is acceptable when the extremely sensitive data is concerned.

5. Security Analysis

Recall that our system is concerned about three security requirements: data confidentiality, keyword privacy, and trapdoor privacy. We present our security analysis for trapdoor privacy by proofing Theorem 5, and data confidentiality and keyword privacy are exhibited through Theorem 6.

The security of our scheme is based on the complex assumption in Definition 2.

Theorem 5 (trapdoor privacy). *Under the assumption of DBDH, the trapdoor generated in our LFSE scheme is indistinguishable against the chosen keyword attack.*

Proof. Assume that an malicious adversary \mathcal{A} is able to break the trapdoor security in our LFSE scheme in a polynomial time with the advantage ϵ which is not negligible. Without loss of generality, we construct an algorithm \mathcal{B} that plays the following game with \mathcal{A} and solves DBDH using the capability of \mathcal{A} .

- (i) Setup: For a security parameter λ , the algorithm \mathcal{B} takes (g, g^a, g^b, g^c, Z) as input, where a, b, c are chosen from \mathbb{Z}_p^* by the challenger \mathcal{C} and Z is also randomly selected from G . The challenger \mathcal{C} picks a coin to denote $x \in \{0, 1\}$. If $x = 1$, computes $Z = g^{abc}$. Otherwise, Z is a random element from G . For the user u_i , the algorithm randomly chooses s, α_i, ν from \mathbb{Z}_p^* and g_1, g_2 from the group G . Then it announces the user's public and private key as $(g^{\alpha_i}, (g^\nu g^{\alpha_i}, g^b, c))$ and sets $g_2 = g, \eta = c$. Furthermore, it announces the search key for the user as $(g_1^{1/\eta})^s g_2^{ab}$.
- (ii) Query Phase 1: The adversary \mathcal{A} issues the following query.

\mathcal{O} .Query: Upon receiving the query request on keyword W from the adversary \mathcal{A} . The algorithm \mathcal{B} selects r, ρ, θ_j randomly from \mathbb{Z}_p^* and then it computes $T_{f_0} = g^{rb\rho} g_1^{\alpha_i b\rho}$, $T_{f_1} = g^{ab\rho}$, $T_{f_2} = g^{\alpha_i \theta_j b\rho}$, $T_{W_1} = g^{H(W)} ((g_1^{1/\eta})^s g_2^{ab})^\eta$, and $T_{W_2} = c/\rho$, where all the other parameters are randomly chosen in a similar way as in Theorem 5. At last, the algorithm \mathcal{B} returns $T_f = (T_{f_0}, T_{f_1}, T_{f_2}, T_{W_1}, T_{W_2})$ as the trapdoor for the keyword W' to \mathcal{A} .
- (iii) Challenge: The adversary \mathcal{A} selects two keywords W_0^* and W_1^* with equal length which are both queried for the first time. Then the algorithm \mathcal{B} flips a coin to choose a random bit of x and computes the trapdoor for the keyword W_x^* as $T_{f_x}^* = (T_{f_0}^*, T_{f_1}^*, T_{f_2}^*, T_{W_1}^*, T_{W_2}^*)$, where $T_{f_0}^* = g^{rb\rho} g_1^{\alpha_i b\rho}$, $T_{f_1}^* = g^{ab\rho}$, $T_{f_2}^* = g^{\alpha_i \theta_j b\rho}$, $T_{W_1}^* = g^{H(W)} g_1^s Z$ and $T_{W_2}^* = c/\rho$.
- (iv) Query Phase 2: The adversary \mathcal{A} does the same thing continuously for polynomial times as in Query Phase

1, but with the restriction that both W_0^* and W_1^* cannot be queried any more.

- (v) Guess Phase: The adversary \mathcal{A} returns a guess $x \in \{0, 1\}'$ to \mathcal{B} . If $x' = x$, it means the adversary \mathcal{A} wins the game, the algorithm \mathcal{B} outputs 1. Otherwise \mathcal{A} fails and \mathcal{B} outputs 0.
- (vi) Analysis: As shown above, we have $T_{W_1} = g^{H(W)} ((g_1^{1/\eta})^s g_2^{ab})^\eta = g^{H(W)} ((g_1^{1/c})^s g^{ab})^c = g^{H(W)} g_1^s g^{abc}$. Compared with $T_{f_2}^*$, we can know $Z = g^{abc}$ clearly. As a result, the adversary \mathcal{A} can win the game with the same probability of winning the DBDH assumption. That means $Adv_{\mathcal{B}(1^\lambda)}^{DBDH} = Pr[x' = x] = \epsilon$, which is contradictory to the DBDH assumption.

In summary, our scheme satisfies the trapdoor indistinguishable secure under the DBDH assumption. \square

Theorem 6 (Ciphertext privacy and keyword privacy). *The proposed scheme shown in Section 4 is IND-CK-CCA secure under the DBDH assumption.*

Proof. Suppose there is a polynomial time adversary \mathcal{A} who can break our proposed scheme with a nonnegligible advantage ϵ , then we can build an algorithm to solve the DBDH assumption. It can be described as a game between a challenger \mathcal{C} and an adversary \mathcal{A} .

Setup: The challenger \mathcal{C} receives $(G, G_T, e, g, g^x, g^y, g^z, Z)$ from the DBDH assumption, where Z is a randomly chosen element from G_T or equals $e(g, g)^{xy^z}$. The challenger \mathcal{C} chooses $s, v \rightarrow \mathbb{Z}_p^*$ and computes $g_2 = g^s, \mathcal{V} = e(g, g)^v$, and also \mathcal{C} sets $g_1 = g^x, g' = g^y$. $(g, g_1, g_2, g', \mathcal{V})$ are sent to the adversary \mathcal{A} as public parameters.

Phase 1: The adversary \mathcal{A} makes the following queries:

- (i) \mathcal{O} .Fog.KeyGen: The adversary \mathcal{A} queries keys for the fog, and the challenger \mathcal{C} picks $\sigma_F, \zeta_F \leftarrow_r \mathbb{Z}_p^*$ at random and outputs $(Pk_F, Sk_F) = (\zeta_F, g^{\zeta_F})$.
- (ii) \mathcal{O} .KeyGen: The adversary \mathcal{A} queries keys for the user $User_i$, and the challenger \mathcal{C} picks $\alpha_i, \beta_i, \delta \leftarrow_r \mathbb{Z}_p^*$ at random and computes $(Pk_{User_i}, Sk_{User_i}) = (g^{\alpha_i}, (t_0, t_1, t_2))$ to \mathcal{A} , where $t_0 = g^v g^{\alpha_i}$, $t_1 = g^{\beta_i}$, and $t_2 = \delta$. For all the attributes owned by the user, the adversary \mathcal{A} also queries the verification keys from \mathcal{O} .Fog.KeyGen and secret verification keys from \mathcal{O} .KeyGen, then \mathcal{A} obtains $vk_j = (d_{1j}, d_{2j})$, and $svk_j = d_{3j} = g^{\alpha_i \theta_j}$, where $\theta_j = H_1(\sigma_F, at_j)$, $d_{1j} = g^{\theta_j}$, and $d_{2j} = \mathcal{V}^{\theta_j}$ are computed by \mathcal{C} .
- (iii) \mathcal{O} .SearchKeyGen: After receiving a commitment $g_1^{1/\eta}$, the adversary \mathcal{A} queries the search key, and the challenger \mathcal{C} computes $\mathcal{S}_i = (g_1^{1/\eta})^s g_2^{\beta_i \delta}$ to \mathcal{A} .
- (iv) \mathcal{O} .ReKey: The adversary \mathcal{A} queries transformed key for the user, and the challenge \mathcal{C} computes $T_0 = t_0^{t_2}$, $T_1 = t_1^{t_2}$, and $T_{2j} = (d_{3j})^{t_2}$ and sends $csk_{ui} = (T_0, T_1, \{T_{2j}\}_{at_j \in \Omega_{ui}})$ to \mathcal{A} .

TABLE 1: Description of parameters.

Parameter	Description
$ S $	the size of the user's attribute set
k	the amount of attributes associated with the user's private key
$ U $	the size of the attribute universe
t	the amount of attributes associated with the ciphertext
N	the number of files to be encrypted
m	the number of keywords to be used to generate indexes
$ G , G_T $	the bit length of the elements belong to the group G, G_T
$ Z_p $	the bit length of the elements belong to the group Z_p
C_p	the computational cost of the pairing operation G, G_{e_T}
C_e, C_{e_T}	the computational cost of the exponential operation in group G, G_{e_T}

- (v) \mathcal{O} .Trapdoor: Upon getting a query on the trapdoor for the keyword W , \mathcal{E} firstly randomly chooses $\rho, \eta \leftarrow_r \mathbb{Z}_p^*$ and computes $T_{f_0} = T_0^\rho, T_{f_1} = T_1^\rho, T_{f_{2j}} = T_{2j}^\rho, T_{W_1} = g^{H(W^l)} \mathcal{S}_i^\eta$, and $T_{W_2} = \eta/\rho$ to \mathcal{A} .

Challenge: The adversary \mathcal{A} gives an access policy \mathbb{A}^* , two equal length plaintexts m_0^*, m_1^* and two keywords W_0^*, W_1^* to \mathcal{E} . Then \mathcal{E} randomly picks $b_1 \in \{0, 1\}$ and constructs the ciphertext as $((m_{b_1}^* \cdot (\prod_{at, \epsilon \in \Omega_l} d_{2j}^{\epsilon_i})^{s_i}, s_l/s^l), 1 \leq l \leq n)$. Also it constructs the index $C_{W_1}^* = e(g, g^z)^{H(W_{b_1}^*)}$. $Z, C_{W_2}^* = g^z, C_{W_3}^* = (g^z)^s$. \mathcal{E} sends the index $I_W^* = (C_{W_1}^*, C_{W_2}^*, C_{W_3}^*)$ for keyword W .

Phase 2: \mathcal{A} can ask a polynomially bounded number of queries adaptively again as Phase 1 except the queried keyword $W \notin \{W_0^*, W_1^*\}$. \mathcal{E} answers \mathcal{A} 's queries as in Phase 1.

Guess: \mathcal{A} outputs guesses b'_1, b'_2 of b_1, b_2 . \mathcal{E} outputs 0 to guess that $Z = e(g, g)^{xyz}$ if $b'_1 = b_1$ and $b'_2 = b_2$; otherwise, it outputs 1 to indicate that it believes Z is a random element.

Analysis: Assume the adversary \mathcal{A} has an advantage ϵ in attacking DBDH assumption and \mathcal{E} has an advantage ϵ' in winning the game. Through the game showed above, we can know $\epsilon' = \epsilon$ as a obvious result. \square

6. Efficiency Analysis

In this section, we analyse the efficiency of our system from both theoretical and experimental aspects. Table 1 illustrates the descriptions of notations we use in the following comparisons.

6.1. Storage and Transmission Cost Analysis. We compare our scheme with the related schemes *VKS* [17], *LHL* [19], *SYL* [18], and *ZSQM* [20] over some important features, which are illustrated in Tables 2 and 3. Though many parameters are generated, stored, and transmitted throughout the whole process, we only consider the following parameters that extremely affect the system efficiency:

- (i) PK: The size of public key PK measures how many storage is needed to store public keys of all entities

for each user to accomplish his computation. As shown in the second column in Table 2, it increases linearly with $|U|$ in [18, 19], which leads to a great amount storage demand for the user. This indicates it is difficult to adopt new attributes in [18, 19]. Because it cannot meet the demands of frequently updating attributes in rapidly changing IoT networks. Reference [20] is file-centered, so the size of PK is related to the number of all files being encrypted, which also causes a large storage requirement for each user. It is obviously that our scheme and [17] only have a small and constant storage requirement.

- (ii) SK: The private key SK is always kept by the user himself, so the size of SK only indicates the secure storage needed to store his private key for each user. The third column reveals that, in [17–20], $|SK|$ increases with the attributes with different efficiencies $k, 2k, S, |U|$, respectively, where $k < 2k < S \ll |U|$. Since the storage of users or devices in IoT networks is limited, it would be desirable if only small and constant storage is needed to store the keys. This expected goal is achieved in our scheme as shown; it is obviously better than the others with only constant storage requirement $2|G| + |Z_p|$, regardless of the attribute number' change.

- (iii) CT: The size of ciphertext CT measures the transmission cost for the user and the storage cost for the cloud server, because the ciphertext is computed by the user, transmitted to the cloud, and stored in the cloud data center. Reference [18] is concentrated in the user management such as user updating and revoking, and the encryption and decryption processes are not revealed in details, so it is empty in the fourth column. Considering that all the five schemes store an access policy in the ciphertext, we ignore this part in the ciphertext size comparison. CT in our scheme and [19] are both linearly increasing with the number of the attributes associated with the user's private key k , which is consistent with the situations in real world. Obviously, CT in [17] is much larger than our scheme and [19], owing to $|U|$ is much larger than k , which means it takes much more transmission overhead for

TABLE 2: Storage and transmission comparisons.

	PK	SK	CT	ID	TD
VSKE [17]	$6 G + G_T $	$2(S + 1) G + Z_p $	$(U + k)(G + G_T)$	$2 G + G_T $	$(2 S + 3) G $
SYL [18]	$3 U G + G_T $	$(2 U + 1) G + 2 Z_p $	–	$(U + 1) G + G_T + Z_p $	$(2 U + 1) G + 2 Z_p $
LHL [19]	$(U + 4) G $	$2k G + Z_p $	$(k + 2) G + G_T $	–	–
ZSQM [20]	$(N + 4) G $	$(S + 3) G + Z_p $	$(N + 2) G + 2 G_T $	$ G_T $	$4 G $
<i>Ours</i>	$5 G + G_T $	$2 G + Z_p $	$2k G_T $	$2 G + G_T $	$ G + Z_p $

TABLE 3: Computation cost comparisons.

	Keygen	Encrypt	Index	Trapdoor	Test	Decrypt
VSKE [17]	$(2 S + 4)C_e$	$(2m + 1)C_e$	$(2 S + 4)C_e$	$(2 U + K)C_e + kC_p$	$(2 S + 1)C_p + S C_{e_T}$	$C_p + C_{e_T}$
SYL [18]	$(2 U + 1)C_e + 2C_{e_T}$	–	$(U + 1)C_e + C_{e_T}$	$(2 U + 1)C_e$	$(U + 1)C_p + C_{e_T}$	–
LHL [19]	$2(k + 1)C_e$	$(k + 2)C_e + C_p$	–	–	–	$(2k + 3)C_p + C_{e_T}$
ZSQM [20]	$5C_e + NC_p$	$(N + 3)C_e + C_p + C_{e_T}$	C_p	$(N + 2)C_e$	$2C_e + (k + 1)C_p$	$(Nk + 1)C_p + C_e$
<i>Ours</i>	$4C_e$	kC_{e_T}	$2(C_e + C_{e_T})$	$2(S + 1)C_e$	$2C_p + C_e$	C_e

the user and more storage requirement for the cloud server. Ciphertext size in [20] is $(N + 2)|G| + 2|G_T|$, because it is file-centered; all files owned by each user are encrypted at one time; this is not convenient if only a part of files are needed to be updated or modified..

- (iv) ID and TD: The size of index ID indicates the transmission overhead for the user and the storage required to store the indexes for retrieving related files for the cloud. The size of trapdoor TD shows the transmission cost for the data user, because the trapdoor is needed to be transmitted to the cloud to accomplish the test and search processes. We do not compute ID and TD for [19] as [19] only concerns about attribute-based encryption and it does not support the function of searching on keywords. For simplicity, we consider generating the index and trapdoor for only one keyword here. We could tell that the scheme in [18] costs most for transmitting both ID and TD between the user and the cloud server. It is shown from the fifth column that [17, 20] have similar ID size with our scheme, which reveals a small and constant storage is required. For the trapdoor size, scheme in [17] is linearly with the user's attribute number, while scheme in [20] and ours is constant; furthermore, ours requires less transmission overhead than [20].

According to the above analyses, our scheme has a better performance in the storage and transmission requirement comparing with the other existing schemes.

6.2. Computational Cost Simulation and Analysis. In this section we present the analysis in terms of the computational cost and comparisons with those related works listed in Table 2. Since operations over Z_p cost much less computational time than operations over groups and the pairing operation, we just consider the latter two fundamental

cryptographic operations. The results are given in Table 3. It is obvious from the table that our scheme has significantly better efficiency than the other schemes.

By adopting the pairing-based cryptography (PBC (URL:<https://crypto.stanford.edu/pbc>)) library, we perform our experiment in C on a computer with Intel(R) Core(TM) i3-3220 CPU @ 3.30 running Ubuntu 16.04.5 with 4.00 GB system memory. This simulation environment is used to perform *Keygen* and *Test*, which are executed by the trusted authority and the cloud server with a great computational capability. In contrast, the users or devices in our system are mostly with low computational capability, to simulate *Encrypt*, *Index*, *Trapdoor*, and *Decrypt* performed by them; we execute our experiment on a client machine with Intel Core Duo CPU running Ubuntu MATE 16.04 with 2 GB system memory. To realize the security requirement of 1024-bit, we use the Type A curve, which is denoted as $E(F_q) : y^2 = x^3 + x$ with parameter $q = 512$ bits, where the order p of both the group G and group G_T is 160 bits and $|G| = |G_T| = 1024$ bits. For simplicity, we assume that the user only generates index for one keyword in our simulation. The simulation result is exhibited in Figure 3.

Once receiving a request from a user to join in the system, TA generates public and private keys for each user with only four exponential operations in our scheme. Clearly from Figure 3(a), the computation cost in our scheme is constant and the smallest among all the schemes. As the attribute number increases, the computational cost for key generation in [17–19] all grows; especially in [18] it climbs up to thousands of milliseconds. Cost in [20] is also constant and similar with ours, this is because we assume the encrypted file number $N = 1$ for simplicity in our experiment. While in reality the scheme in [20] is file-centered, the computational cost for key generation grows with the number of encrypted files increases, but in our scheme the cost for key generation is irrelevant with the encrypted file numbers.

After receiving the keys from TA, the user encrypts the files with his/her keys before updating them to the cloud

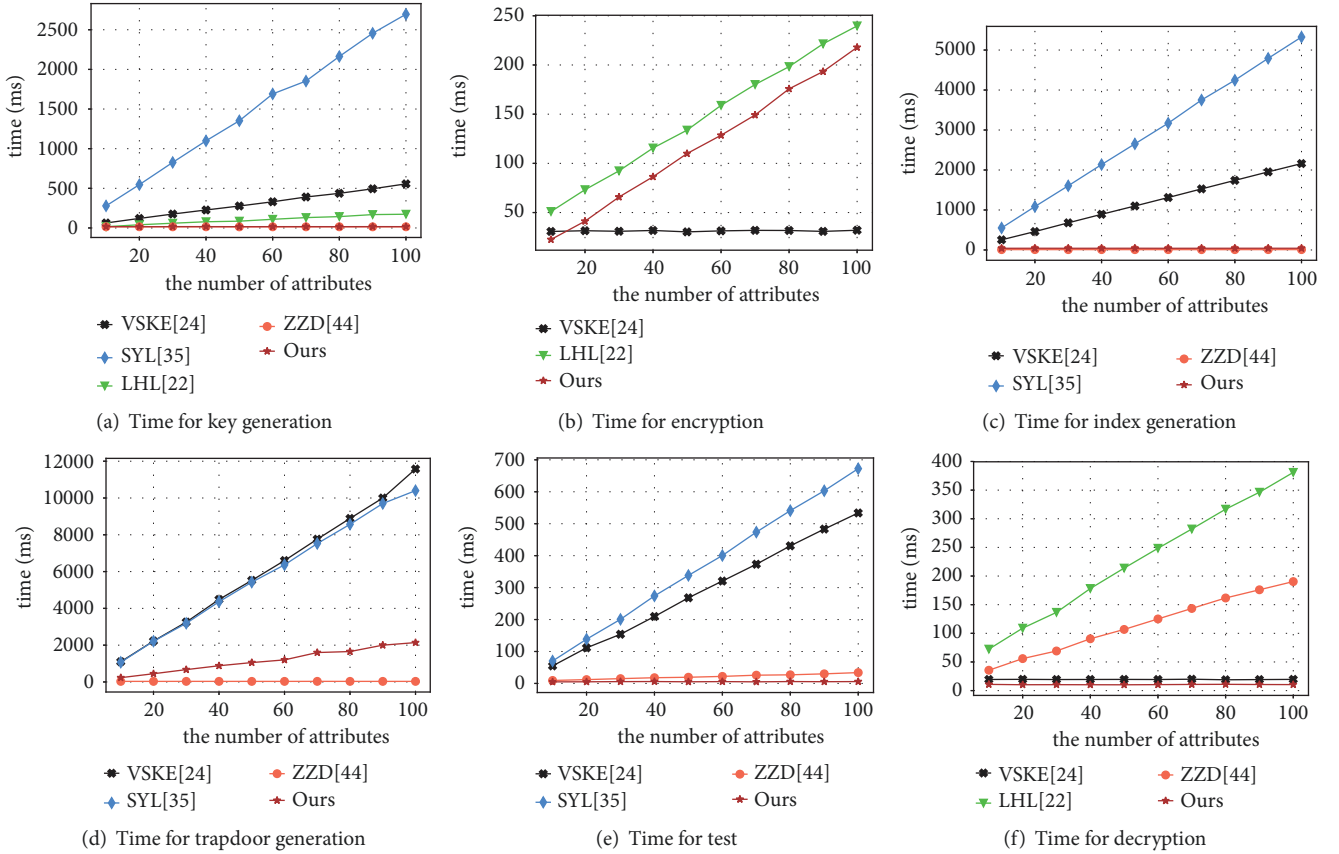


FIGURE 3: Comparison of computational cost.

server. Reference [18] focuses on attributed-based encryption to manage users; the data encryption and decryption phase are not described in details; therefore it is not considered in our encryption simulation. Reference [20] is not considered in the encryption phase because the scheme encrypts all files of one user at once, while the others encrypt one file at a time. As shown in Figure 3(b), cost for encryption in our scheme and [19] are increasing linearly with the number of attributes grows, due to the file is encrypted associated with the attributes embedded in the access policy. When the attribute number is 50, our scheme needs 109.96 milliseconds and [19] needs 133.758 milliseconds, which is lightly larger than ours. Reference [17] has the lowest computational cost because they use symmetric encryption method to encryption and the cost shown in Figure 3(b) is for access control in encryption phase.

Next is about querying on keywords, which involves the three algorithms: *Index*, *Trapdoor*, and *Test*. The computational cost for them is exhibited in Figures 3(c), 3(d), and 3(e), respectively. Because [19] has no capability for searching based on keywords, it is not considered in our comparison for these three phases. References [17, 18] have a obviously large increase in computation burden when the number of the attributes grows. When the attribute number grows up to 100, almost 15000 milliseconds are required to complement these three algorithms to achieve querying on keywords for the two schemes, which causes a long network delay. Our scheme has a similar computational cost with the scheme

in [20] proposed to speed up in the industrial IoT network, which has been proved having a good efficiency in fast query.

Last, the computational cost for the decryption phase is shown in Figure 3(f). The efficiency for the decryption algorithm is very important because one keyword is always associated with a lot of different files. To decrypt all the returned files in a short time in IoT networks is a key issue in recent researches. As shown in Figure 3(f), our scheme satisfies this demand with only less than 13 milliseconds is required, regardless of the increasing of the attribute number. And the scheme in [17] has a lightly bigger cost than ours. In contrast, the other two schemes' cost grows enormously with the attributes' number, which causes a super large computational burden because of the large amount of returned files and the user's limited computation capability.

In summary, our proposed scheme enjoys a good efficiency in storage, transmission requirement, and computational cost, which indicates it is suitable for the healthcare related IoT networks.

7. Related Work

7.1. Healthcare Related IoT Security. Security is one of the most important issues in the healthcare related IoT Networks. This is not only because the vulnerability of IoT devices themselves, which can be easily attacked or physically destroyed, but also because the data collected and processed in IoT

networks are highly sensitive and tightly related to our life. Johns Hopkins University developed an hospital-centralized patient monitoring system called MEDiSN [22]. But in this system secure communication especially data integrity and user authentication are not implemented [23]. Similar with MEDiSN, other systems such as CodeBlue [24] and MobiCare [25] are implemented in the infrastructure layer without considering the real communication security.

To achieve real communication security, encryption operations are essential. However, most of the existing encryption schemes demand complex computation operations and high process overload. How to overcome these limitations is an important issue. In [26, 27], the authors present a secure and efficient authentication and authorization framework for healthcare related IoT network but high processing power is needed. In [28], the authors implement an IoT-based health prescription assistant and achieve user authentication and access control on their system. However, the data confidentiality is not considered during the transmission process [29]. Although they have reduced some communication and computation latency in their small-scale data experiment, it is still not enough for real world network with super large amount of data [30].

7.2. ABE in Cloud Computing Paradigm. As an extension of identity-based encryption, attribute-based encryption was first introduced by Sahai and Waters [13]. It has been applied to a lot of encryption schemes to achieve fine-grained access control over encrypted data. Particularly, ABE was extended by Goyal et al. [31] to form two complementary flavors: key-policy ABE (KP-ABE) and ciphertext-policy ABE (CP-ABE). KP-ABE takes attributes to describe the ciphertexts, and policies over these attributes are associated with users' keys, while in CP-ABE it is reversed. CP-ABE makes it possible that the users can get access to the encrypted data and decrypt the data only if the access structures match attributes.

Same as the originally proposed ABE scheme in [13], the most classic architecture of ABE access control schemes apply a single central authority to take charge of enrolling, updating all attributes and managing keys for all entities. In such centralized ABE frameworks, the most difficult but important part is to achieve efficient revocation for users and attributes. In [32], the authors put forward an expiration time for each attribute to maintain revocation but it turns out having issues in backward and forward. The authors from [33, 34] succeed in overcoming the above issues through adopting the concept of proxy-based re-encryption. Also, lazy revocation [33, 35] and revocable-storage ABE [36] are designed to achieve revocation to prevent the message from unauthorized users.

As the IoT networks expand, the centralized ABE paradigm with only one single authority has a great drawback in efficiency due to the super large amount of data. Therefore, multiauthority ABE was introduced by [37], in which a global identifier was assigned by the central authority to each user as a unique ID, aiming to distinguish users without attributes by independent authorities. Furthermore, more works such as [38–40] improve the above scheme by cancelling the user's consistent GID to avoid privacy leakage and support

collusion resistance; this paradigm is called as decentralized ABE.

No matter in a centralized or decentralized ABE paradigm, a user may not withstand the financial attempt and share his attributes to other users. In order to avoid a decryption privilege leakage, works in [41, 42] provide access control schemes with traceability, where the user who leaks the decryption key to someone else can be traced and revoked by the system. As people become more concerned about personal privacy, the access policy itself can be taken as sensitive information and need to be protected from unauthorized users. Works in [43] achieve anonymity by designing three protocols together with homomorphic encryption and scrambled circuit evaluation to protect both the policies and the credentials.

7.3. Searchable Encryption with ABE in Cloud Computing Paradigm. The searchable encryption was firstly proposed by [44] and has been widely researched and used. It has indicated a new direction for operating searching on ciphertexts in cloud computing [45]. Both the notion of symmetric encryption with keyword search (SESK) and the public key encryption with keyword search (PESK) are gaining a lot of attentions. They have been developed to support different functions, such as works in [18, 46–51]. However, these schemes cannot achieve fine-grained access control on ciphertexts.

The attribute-based keyword search (ABKS) was proposed in [52], in which the cloud server checks whether the user has the capability to decrypt the required encrypted ciphertext before searching it by a signature built from the user's attributes. But this scheme cannot maintain the security of keywords. Some other works also proposed different schemes based on ABKS to support specific functions such as Checkability [19], fuzzy keyword search [53], revocation [54], and verifiability [55]. But most of these works require the users to do complex computation like pairing and exponential operations many times, which is not practical because of the user's limited computation ability. Therefore, how to transfer the heavy computation burden and reduce the times of complex computation operations without losing security requirements is the most important challenge for now.

8. Conclusion

In this paper, we design a keyword searchable encryption with fine gained access control for our proposed healthcare related IoT-fog-cloud framework. Through our design, the users could achieve a fast and efficient service by reducing the calculation overload and storage with the help of the fog and cloud, especially the data user only needs to do a exponential operation to retrieve the message. In our scheme, the fogs are capable of helping the trusted authority to manage the users and their attributes through authoring their query keys. In addition, our scheme is very efficient because only the authorized users could download the keyword-matched-part of ciphertexts by refusing unauthorized research and unauthorized users. At last, our scheme is proofed IND-CK-CCA secure and trapdoor indistinguishably secure. We

also show our scheme takes less storage and transmission consumption and much less computational cost through theoretical analysis and experimental evaluations.

We assume fogs and the cloud do not collude with each other in this paper; next we will consider in achieving the collusion resistance in our proposed IoT-Fog-Cloud system. We are also interested in the user update and the attribute replacement with a more efficient method in our future research. How to improve the efficiency of searching process by designing better structures of indexes and trapdoors for the keywords in the cloud server is also in our future consideration.

Data Availability

The PBC (Pairing-Based Cryptography) library (version: pbc-0.5.14.tar.gz) used to support the findings of this study is included as a comment within the article in Section 5. It is a free C library built on the GMP library that performs the mathematical operations underlying pairing-based cryptosystems. It can be accessed from <https://crypto.stanford.edu/pbc/> and the GMP library (version:gmp-6.1.2.tar.lz) is also a free library can be accessed from <https://gmplib.org>.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant Nos. 61571010 and 61572070).

References

- [1] K. Ashton, "That 'internet of things' thing," *RFID Journal*, vol. 22, no. 7, pp. 97–114, 2009.
- [2] D. L. Brock, "The electronic product code (epc)," *Auto-ID Center White Paper MIT-AUTOID-WH-002*, 2001.
- [3] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the 1st ACM Mobile Cloud Computing Workshop, MCC 2012*, ACM, pp. 13–16, Finland, August 2012.
- [4] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [5] Y. Ding, Y. Jin, L. Ren, and K. Hao, "An intelligent self-organization scheme for the internet of things," *IEEE Computational Intelligence Magazine*, vol. 8, no. 3, pp. 41–53, 2013.
- [6] Y. Yuehong, Y. Zeng, X. Chen, and Y. Fan, "The internet of things in healthcare: an overview," *Journal of Industrial Information Integration*, vol. 1, pp. 3–13, 2016.
- [7] C. Koop, R. Mosher, L. Kun et al., "Future delivery of health care: cybercare," *IEEE Engineering in Medicine and Biology Magazine*, vol. 27, no. 6, pp. 29–38, 2008.
- [8] Y. Huo, C. Hu, X. Qi, and T. Jing, "LoDPD: a location difference-based proximity detection protocol for fog computing," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1117–1124, 2017.
- [9] I. Stojmenovic and S. Wen, "The fog computing paradigm: scenarios and security issues," in *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS '14)*, pp. 1–8, IEEE, Warsaw, Poland, September 2014.
- [10] Y. Huo, C. Yong, and Y. Lu, "Re-ADP: real-time data aggregation with adaptive ω -event differential privacy for fog computing," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 6285719, 13 pages, 2018.
- [11] J. Mao, W. Tian, J. Jiang, Z. He, Z. Zhou, and J. Liu, "Understanding structure-based social network de-anonymization techniques via empirical analysis," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, p. 279, December 2018.
- [12] Y. Huo, Y. Tian, L. Ma, X. Cheng, and T. Jing, "Jamming strategies for physical layer security," *IEEE Wireless Communications Magazine*, vol. 25, no. 1, pp. 148–153, 2018.
- [13] A. Sahai and B. Waters, "Fuzzy identity-based encryption," *Lecture Notes in Computer Science*, vol. 3494, pp. 457–473, 2005.
- [14] M. Sookhak, F. R. Yu, M. K. Khan, Y. Xiang, and R. Buyya, "Attribute-based data access control in mobile cloud computing: taxonomy and open issues," *Future Generation Computer Systems*, vol. 72, pp. 273–287, 2017.
- [15] J. Mao, J. Bian, W. Tian et al., "Phishing page detection via learning classifiers from page layout feature," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, p. 43, 2019.
- [16] A. M. Rahmani, T. N. Gia, B. Negash et al., "Exploiting smart e-health gateways at the edge of healthcare internet-of-things: a fog computing approach," *Future Generation Computer Systems*, vol. 78, pp. 641–658, 2018.
- [17] Y. Miao, J. Ma, Q. Jiang, L. Xiong, and A. K. Sangaiah, "Verifiable keyword search over encrypted cloud data in smart city," *Computers & Electrical Engineering*, vol. 65, pp. 90–101, 2017.
- [18] W. Sun, S. Yu, W. Lou, and T. Hou, "Protecting your right: verifiable attribute-based keyword search with fine-grained-owner-enforced search authorization in the cloud," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 4, pp. 1187–1198, 2014.
- [19] J. Li, X. Huang, X. Chen, and Y. Xiang, "Securely outsourcing attribute-based encryption with checkability," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2201–2210, 2014.
- [20] R. Zhou, X. Zhang, X. Du, X. Wang, G. Yang, and M. Guizani, "File-centric multi-key aggregate keyword searchable encryption for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, pp. 3648–3658, 2018.
- [21] O. Goldreich, "Foundations of cryptography," in *Basic Applications*, vol. 2, Cambridge University Press, Cambridge, UK, 2004.
- [22] J. G. Ko, R. Musaloiu-Elefteri, J. H. Lim et al., "MEDiSN: medical emergency detection in sensor networks," in *Proceedings of the 6th ACM Conference on Embedded Networked Sensor Systems, SenSys 2008*, pp. 361–362, USA, November 2008.
- [23] J. Mao, Y. Zhang, P. Li, T. Li, Q. Wu, and J. Liu, "A position-aware Merkle tree for dynamic cloud data integrity verification," *Soft Computing*, vol. 21, no. 8, pp. 2151–2164, 2017.
- [24] G. Kambourakis, E. Klaoudatou, and S. Gritzalis, "Securing medical sensor environments: The CodeBlue framework case," in *Proceedings of the 2nd International Conference on Availability, Reliability and Security, ARES 2007*, pp. 637–643, Australia, April 2007.
- [25] R. Chakravorty, "Abstract mobicare: a programmable service architecture for mobile medical care," in *Proceedings of the Fourth Annual IEEE International Conference on Pervasive*

- Computing and Communications Workshops (PERCOMW'06)*, pp. 532–536, Pisa, Italy, 2006.
- [26] S. R. Moosavi, T. N. Gia, E. Nigussie, and A.-M. Rahmani, “Session resumption-based end-to-end security for healthcare internet-of-things,” in *Proceedings of the IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable*, pp. 581–588, UK, October 2015.
- [27] S. R. Moosavi, T. N. Gia, A. M. Rahmani et al., “Sea: A secure and efficient authentication and authorization architecture for iot-based healthcare using smart gateways,” *Procedia Computer Science*, vol. 52, no. 1, pp. 452–459, 2015.
- [28] M. Hossain, S. M. R. Islam, F. Ali, K.-S. Kwak, and R. Hasan, “An internet of things-based health prescription assistant and its security system design,” *Future Generation Computer Systems*, vol. 82, pp. 422–439, 2018.
- [29] Y. Jia, Y. Chen, X. Dong, P. Saxena, J. Mao, and Z. Liang, “Man-in-the-browser-cache: persisting HTTPS attacks via browser cache poisoning,” *Computers & Security*, vol. 55, no. 1, pp. 62–80, 2015.
- [30] J. Mao, Y. Chen, F. Shi, Y. Jia, and Z. Liang, “Toward exposing timing-based probing attacks in web applications,” *Sensors*, vol. 17, no. 3, p. 464, 2017.
- [31] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS '06)*, pp. 89–98, November 2006.
- [32] M. Pirretti, P. Traynor, P. McDaniel, and B. Waters, “Secure attribute-based systems,” *Journal of Computer Security*, vol. 18, no. 5, pp. 799–837, 2010.
- [33] K. Yang, X. Jia, and K. Ren, “Attribute-based fine-grained access control with efficient revocation in cloud storage systems,” in *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security (ASIACCS '13)*, pp. 523–528, May 2013.
- [34] P. K. Tysowski and M. A. Hasan, “Hybrid attribute-and re-encryption-based key management for secure and scalable mobile applications in clouds,” *IEEE Transactions on Cloud Computing*, vol. 1, no. 2, pp. 172–186, 2013.
- [35] A. F. Barsoum and A. Hasan, “Enabling dynamic data and indirect mutual trust for cloud computing storage systems,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 12, pp. 2375–2385, 2013.
- [36] A. Sahai, H. Seyalioglu, and B. Waters, “Dynamic credentials and ciphertext delegation for attribute-based encryption,” *Lecture Notes in Computer Science*, vol. 7417, pp. 199–217, 2012.
- [37] M. Chase, “Multi-authority attribute based encryption,” in *Proceedings of the Conference on Theory of Cryptography*, 2007.
- [38] M. Chase and S. S. M. Chow, “Improving privacy and security in multi-authority attribute-based encryption,” in *Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS '09)*, pp. 121–130, Chicago, Ill, USA, November 2009.
- [39] S. Ruj, A. Nayak, and I. Stojmenovic, “Dacc: distributed access control in clouds,” in *Proceedings of the International Joint Conference of IEEE Trustcom-11/IEEE ICESS-11/FCST*, 2011.
- [40] Y. Kan and X. Jia, “Dac-macs: Effective data access control for multi-authority cloud storage systems,” in *Proceedings of the Infocom, IEEE*, 2013.
- [41] Z. Liu, Z. Cao, and D. S. Wong, “Traceable CP-ABE: how to trace decryption devices found in the wild,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 55–68, 2015.
- [42] J. Ning, Z. Cao, X. Dong, and L. Wei, “White-box traceable CP-ABE for cloud storage service: how to catch people leaking their access credentials effectively,” *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 5, p. 1, 2016.
- [43] K. Frikken, M. Atallah, and J. Li, “Attribute-based access control with hidden policies and hidden credentials,” *IEEE Transactions on Computers*, vol. 55, no. 10, pp. 1259–1270, 2006.
- [44] D. X. Song, D. Wagner, and A. Perrig, “Practical techniques for searches on encrypted data,” in *Proceedings of the IEEE Symposium on Security and Privacy*, 2002.
- [45] J. Mao, W. Tian, Y. Zhang et al., “Co-check: collaborative outsourced data auditing in multicloud environment,” *Security and Communication Networks*, vol. 2017, Article ID 2948025, 13 pages, 2017.
- [46] Z. Yang, Z. Sheng, and R. N. Wright, “Privacy-preserving queries on encrypted data,” in *Proceedings of the European Symposium on Research in Computer Security*, 2006.
- [47] Q. Liu, C. C. Tan, J. Wu, and G. Wang, “Towards differential query services in cost-efficient clouds,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 6, pp. 1648–1658, 2014.
- [48] H. Li, D. Liu, Y. Dai, T. H. Luan, and X. S. Shen, “Enabling efficient multi-keyword ranked search over encrypted mobile cloud data through blind storage,” *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 1, pp. 127–138, 2017.
- [49] B. Dan, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, “Public key encryption with keyword search,” *Eurocrypt*, vol. 3027, no. 16, pp. 506–522, 2004.
- [50] L. Fang, W. Susilo, C. Ge, and J. Wang, “Public key encryption with keyword search secure against keyword guessing attacks without random oracle,” *Information Sciences*, vol. 238, no. 7, pp. 221–241, 2013.
- [51] C. Hui, Z. Wan, R. H. Deng, G. Wang, and Y. Li, “Efficient and expressive keyword search over encrypted data in cloud,” *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 3, p. 1, 2016.
- [52] F. Zhao, T. Nishide, and K. Sakurai, “Multi-user keyword search scheme for secure data sharing with fine-grained access control,” in *Proceedings of the International Conference on Information Security and Cryptology*, 2011.
- [53] P. Xu, H. Jin, Q. Wu, and W. Wang, “Public-key encryption with fuzzy keyword search: a provably secure scheme under keyword guessing attack,” *IEEE Transactions on Computers*, vol. 62, no. 11, pp. 2266–2277, 2013.
- [54] J. Li, Y. Shi, and Y. Zhang, “Searchable ciphertext-policy attribute-based encryption with revocation in cloud storage,” *International Journal of Communication Systems*, vol. 30, no. 1, p. e2942, 2017.
- [55] Q. Zheng, S. Xu, and G. Ateniese, “VABKS: Verifiable attribute-based keyword search over outsourced encrypted data,” in *Proceedings of the 33rd IEEE Conference on Computer Communications, IEEE INFOCOM 2014*, IEEE, pp. 522–530, Canada, May 2014.

Research Article

Replication-Based Data Dissemination in Connected Internet of Vehicles

Xiying Fan ^{1,2} **Chuanhe Huang** ^{1,2} **Junyu Zhu**³ and **Bin Fu**⁴

¹*School of Computer Science, Wuhan University, China*

²*Collaborative Innovation Center of Geospatial Technology, China*

³*Research Center for Computer and Microelectronics Industry Development, MIIT (China Software Testing Center), China*

⁴*Department of Computer Science, The University of Texas Rio Grande Valley, Edinburg, USA*

Correspondence should be addressed to Chuanhe Huang; huangch@whu.edu.cn

Received 24 October 2018; Accepted 12 March 2019; Published 4 April 2019

Guest Editor: Zaobo He

Copyright © 2019 Xiying Fan 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.

Due to the dynamically changing topology of Internet of Vehicles (IoV), it is a challenging issue to achieve efficient data dissemination in IoV. This paper considers strongly connected IoV with a number of heterogenous vehicular nodes to disseminate information and studies distributed replication-based data dissemination algorithms to improve the performance of data dissemination. Accordingly, two data replication algorithms, a deterministic algorithm and a distributed randomised algorithm, are proposed. In the proposed algorithms, the number of message copies spread in the network is limited and the network will be balanced after a series of average operations among the nodes. The number of communication stages needed for network balance shows the complexity of network convergence as well as network convergence speed. It is proved that the network can achieve a balanced status after a finite number of communication stages. Meanwhile, the upper and lower bounds of the time complexity are derived when the distributed randomised algorithm is applied. Detailed mathematical results show that the network can be balanced quickly in complete graph; thus highly efficient data dissemination can be guaranteed in dense IoV. Simulation results present that the proposed randomised algorithm outperforms the present schemes in terms of transmissions and dissemination delay.

1. Introduction

As a promising branch of Internet of Things (IoT), Internet of Vehicles (IoV) mainly improves traffic efficiency and assists road safety through wireless communication technologies [1]. Interconnected by means of vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications, IoV could provide data services including road safety (such as collision warning and smart traffic management), entertainment demand services (such as advertisements and online videos), and location-based services (such as interest points and path optimization). Thus IoV plays a vital role in accident warning, traffic management, and user entertainment services [2].

To enhance on-road transportation safety and efficiency, efficient data dissemination which can enable high-rate communications and rapid data dissemination, is essential for

applications in IoV. Data replication can improve dissemination performance effectively, as all the vehicles involved in data dissemination help disseminate a certain quantity of message copies. Therefore, the process of information dissemination could be expedited and the dissemination delay could be reduced [3].

Characterised by decentralised control, emerging applications in IoV are confronted with problems, such as efficient cooperation among vehicles and network consensus [4]. Adapting to dynamically changing network, a type of algorithm based on distributed averaging, gossip algorithm, attracts lots of interest [5]. Through a series of communications, the participants could have the same value or reach the common state. However, gossip-based algorithms might lead to a significant waste of network resources (network capacity, bandwidth, and computing resources)

by transmitting redundant information. Similarly, although dynamic data replication can accelerate data dissemination in distributed ad hoc networks, replication-based methods could also meet a variety of problems; for instance, the high density may result in longer communication delay, which causes network resources wasting and scalability issues. Towards data replication, Spyropoulos et al. [6] proposed to disseminate a limited number of replicas; however they did not consider available network capacity and bandwidth. RAPID [7] solved the problem by taking data utilities into account to determine how the replication should carry out. Additionally, traditional replication-based dissemination algorithms could lead to high communication overhead as well as congestions and sometimes even broadcast storm by passing around redundant information. Considering the mentioned problems, the quantity of data replicas spread in the area should be controlled.

To accelerate information dissemination, every vehicle could carry a number of data replicas. In this way, the computational burden can be distributed among the vehicles and the network load balancing can be achieved. Accordingly, a concept of network balance is proposed.

In this study, we mainly investigate data dissemination in dense IoV, which can be abstracted as complete graph by graph theory. In the situation of complete graph, we assume that every two vehicular nodes are within each other's communication range. As a tentative study, the conference paper [8] focuses on data dissemination in the context of complete graph.

Additionally, since nodes in the network can have different capabilities in terms of computation or processing due to their heterogeneity, it would be better to carry an appropriate number of data replicas according to the vehicles' own capabilities rather than an approximately equal number of data replicas as [9]. Dissemination strategies will be adjusted according to different capabilities of nodes. However, most previous work studies homogeneous vehicles in vehicular communication. Therefore, this study considers heterogeneous vehicles such that data replication strategy should be determined by the capabilities of the vehicles.

To achieve data dissemination to a target area with reduced dissemination delay and consumed resources, a deterministic algorithm and a distributed randomised algorithm based on data replication are proposed for dense vehicular scenarios. In the proposed framework, different types of vehicles have different dissemination capabilities. Each vehicular node is allocated with a corresponding value to indicate the quantity of replicas that the node can spread. Every node selects one of its neighbours to exchange data depending on the proposed algorithms, and then the pair of nodes take proportional average operations, such that the values of the vehicular nodes could be updated. By iterating the operations among the nodes, the network can reach a balanced status; that is, the network converges to a consensus. To prove the efficiency of the algorithms, we evaluate the convergence complexity by calculating the average operations needed for network balance. Detailed theoretical analysis of convergence complexity is provided.

To summarise, the current study presents the following key contributions.

(1) We consider heterogeneous vehicles with different capabilities and propose a deterministic algorithm and a distributed randomised algorithm for dense scenarios in IoV, by utilising data replication to enhance data dissemination. In the algorithms, the quantity of data copies is bounded while a network balanced status can be achieved.

(2) Theoretical analysis is presented to illustrate the number of stages needed when the network achieves a balanced status in the deterministic algorithm. The upper and lower bounds of the distributed randomised algorithm are also derived. Simulation results show the effectiveness of the distributed randomised algorithm.

The remainder of the paper is structured as below. Section 2 introduces related data dissemination schemes in vehicular networks as well as the average consensus problem. Section 3 describes the system framework. Section 4 presents a deterministic algorithm and a distributed randomised algorithm for complete graph. Section 5 gives the upper and lower bounds of the proposed randomised algorithm. Section 6 evaluates the performance of the distributed randomised algorithm. Finally, Section 7 summarises the study and Section 8 presents the future prospect.

2. Related Work

This section mainly introduces some information dissemination schemes in IoV. Meanwhile, related work on average consensus problem is discussed.

2.1. Data Dissemination in Vehicular Networks. As multiple data replicas can be forwarded to an area of interest, many works have studied replication-based data dissemination schemes and a variety of dissemination strategies have been developed [6, 7]. As a simple data dissemination scheme, while flooding has the merits of high dissemination speed and wide coverage, it could cause serious broadcast storm. Towards the problem, improvements have been made by Torres et al. [10] to adapt to various traffic scenarios.

In the routing mechanism developed by [11], the amount of data spread in the target area mainly depended on the distance from source to the base station within its communication range. Xing et al. [12] proposed a framework of utility maximisation problem for multimedia dissemination and obtained the closed form of the network utility. Wu et al. [13] aimed to fully utilise the available network capacity and presented a distributed data replication scheme. Shen et al. [14] designed a data dissemination framework to schedule data transmission with maximum dissemination utility and took advantage of the space-time network coding to improve the network efficiency. To minimise the dissemination delay to a desired number of receivers, Yan et al. [15] converted the problem to processor scheduling problem and proposed heuristics to solve the problem. Xiang et al. [16] quantified different classes of data preferences and designed a safety data dissemination protocol. Zhao et al. [17] incorporated link quality and diversity as the sender metric, based on which

an efficient selection mechanism for bulk data dissemination was proposed. Chen et al. [18] studied the relation between content replication and RSU deployment and developed a cooperative replication scheme. Given a set of tasks to be executed in vehicular clouds, Jiang et al. [19] proposed the balanced-task-assignment (BETA) policy to minimise the probability of deadline violation. The authors in [20] focused on data dissemination in IoV with social characteristic and applied the property in the design of dissemination strategies. Fan et al. [9] considered vehicles with the same capability while Lin et al. [21] studied resource allocation in vehicular cloud computing systems with heterogeneous vehicles and proposed a semi-Markov based architecture to achieve optimal resource allocation. Ghorai et al. [22] considered that the obstacles might affect radio propagation and then proposed a forwarding node selection algorithm based on fuzzy logic. Ding et al. [23] studied the cooperation in group vehicular interactions and presented a dynamic member public goods game model and a greedy based neighbour selection scheme towards the high density vehicular networks.

2.2. Average Consensus Problem in Wireless Networks. As the average consensus problem attracts lots of interest in research areas, such as wireless networks, many researches have been done to address the problem [24]. Boyd et al. [25] studied randomised gossip algorithms. They developed a distributed subgradient method to improve the speed of gossip algorithms and designed a framework that could be applied to analyse distribute algorithms in different scenarios. The consensus studied by Fagnani et al. [26] could be achieved at some point. Different from average preserving algorithms, this consensus point might not be the same as the average by initial states. As bidirectional communication among agents was not necessary, the studied algorithms could be applied to more settings. To describe gossip periodic sequences in an undirect graph, Yu et al. [27] used transfer function of the node. Chen et al. [28] utilised probabilistic grouping in the proposed distributed random grouping algorithm to converge to the sums. Therefore, the impact of dynamically changing topology would be alleviated. Aysal et al. [29] developed a novel gossiping algorithm for deriving the average values that could simplify the process of random gossiping and described the conditions to guarantee the network convergence. To study network consensus in strongly connected networks, Wu et al. [30] presented a gossip-based algorithm and showed it could quickly reach consensus as well as reducing the consumed transmissions. Franceschelli et al. [31] studied the execution time of heterogeneous tasks in an undirected graph. They proposed randomised interaction algorithm based on gossip to let the nodes cooperatively complete the tasks to minimise the task execution time. Nedić et al. [32] studied the characteristics of weighted-averaging dynamic for network consensus.

The mentioned literature talks about the reliability and efficiency of data dissemination as well as network consensus rather than both of the problems. Also, as few of the previous works study the heterogeneity of vehicles; this study considers a scenario that a number of vehicles with different

capabilities exist, according to which the replication strategy is determined. In summary, we aim to design replication-based dissemination schemes to facilitate data dissemination in heterogeneous vehicular networks while the network convergence rate is investigated.

3. System Framework

3.1. Network Architecture. The proposed network architecture is shown in Figure 1. As it is shown, a source vehicle carries a message and aims to disseminate the message to the area that is indicated by the circle. The message dissemination is completed by pure vehicle-to-vehicle communication. In the network, two vehicular nodes would update their own values after an average operation until the network consensus is reached.

Different from previous settings, this study considers heterogeneous vehicles with different capabilities such that the number of replicas assigned to each vehicle should be determined according to the vehicle's capability. For example, there are three types of vehicles that are classified as red vehicles, yellow vehicles, and black vehicles. Assume that red vehicle could carry 100 replicas while yellow and black ones could carry 200 and 300, respectively. Assign each type of vehicles a parameter to indicate the maximum number of replicas the vehicles can spread. When two vehicles communicate, they exchange the replicas not by simply averaging their values; instead, the average operations are taken according to the vehicles' capabilities. For example, in Figure 1, let n_R and n_Y denote the values of the red vehicle and the yellow one, respectively. The following operation will be taken when they communicate with each other,

$$\begin{aligned} n'_R &= (n_R + n_Y) \times \frac{100}{100 + 200} \\ n'_Y &= (n_R + n_Y) \times \frac{200}{100 + 200} \end{aligned} \quad (1)$$

where n'_R and n'_Y denote the new values of the red vehicle and the yellow one, respectively.

3.2. Time Model. In the proposed architecture, it is allowed to let pairs of independent nodes contact and exchange information in parallel. We apply the synchronous time model [25]. As in the synchronous time model, the nodes can communicate simultaneously, while it only allows one node to communicate in each time slot in the asynchronous time model.

3.3. Assumptions and Definitions. This part presents some related assumptions and definitions for bounded number of data dissemination.

Assumption 1. The vehicular network of our concern is described as an undirected graph $G(V, E)$. Assume that every two vehicular nodes can exchange information with each other. Consider dense IoV, such as the scenario when road congestions happen or parking lots with many parked cars.

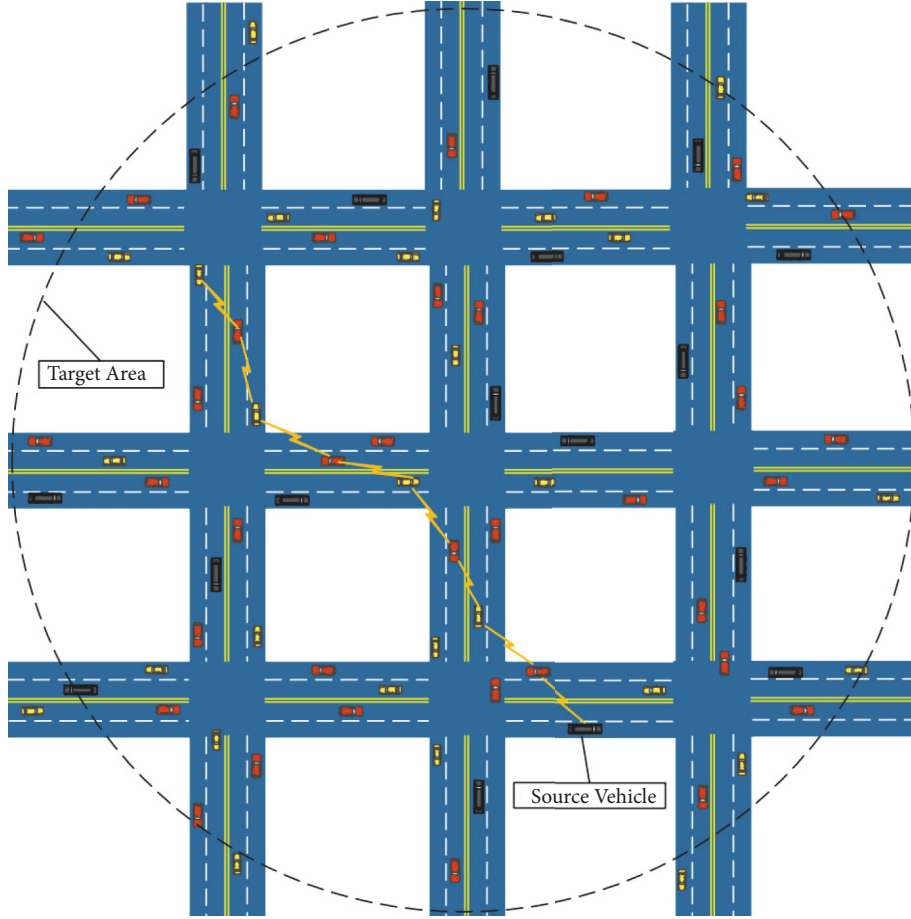


FIGURE 1: Data dissemination area.

According to graph theory, this type of network topology can be abstracted as complete graph. Every vehicle node owns a value to indicate the number of replicas it could spread. Let n_i denote the value of node i . Accordingly, graph $G(V, E)$ becomes a weighted graph.

Assumption 2. As it is stated that the number of message replicas is limited to a value, we let parameter n denote the maximum quantity of replicas. Assume there are k types of vehicles in the system, e.g., $type_1, type_2, \dots, type_k$. The corresponding capability of the vehicles are indicated as $N_{type_1}, N_{type_2}, \dots, N_{type_k}$. If the vehicle of $type_i$ (value n_i) and the vehicle of $type_j$ (value n_j) meet, the operation should be taken based on proportion, N_{type_i} or $N_{type_j}/(N_{type_i} + N_{type_j})$, such that the new values should be $n'_i = (n_i + n_j) \times N_{type_i}/(N_{type_i} + N_{type_j})$, and $n'_j = (n_i + n_j) \times N_{type_j}/(N_{type_i} + N_{type_j})$. In a system with homogeneous vehicles, $n'_i = n'_j = (n_i + n_j)/2$.

To calculate the communication stages needed to obtain network balance, the following lemmas and definitions are presented.

Definition 3. The nodes in the weighted graph are associated with corresponding nonnegative numbers. We say that an ϵ -balanced status is achieved among the nodes in the graph with the following conditions met.

- (i) For any node, the number of message replicas is not smaller than 1, that is, $n_i \geq 1$.
- (ii) For any pair of nodes with $n_i, n_j > 0$, $|n_i - n_j| \leq \epsilon$, where $\epsilon > 0$.
- (iii) If $n_i \geq 2$ and $n_j = 0$, no edge should exist between the two nodes with values n_i and n_j .

Lemma 4. Let a, b , and c, d be real numbers satisfying the condition that $a + b$ is equal to $c + d$. Then the following results can be obtained (1) $(a^2 + b^2) - (c^2 + d^2) = 2(b - d)(b - c)$, and (2) $(a^2 + b^2) - (c^2 + d^2) \geq 0$ if the inequality $a \leq c \leq d \leq b$ holds.

Lemma 4 is very easy to obtain and will be used to represent the change of potential.

Definition 5. Assume that \mathbb{R} denotes a list containing real numbers and \mathbb{N} denotes a list containing nonnegative integers. We present the following definitions.

- (i) A real average function $A(.,.)$ is a mapping $\mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R} \times \mathbb{R}$, such that for two numbers $a \leq b$, $A(a, b) = ((a + b)/2, (a + b)/2)$ if $a + b \geq 2$, or $A(a, b) = (a, b)$ if $a + b < 2$.
- (ii) An integer average function $A(.,.)$ is a mapping $\mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N} \times \mathbb{N}$ such that for two numbers $a \leq b$, $A(a, b) = (k, k)$ if $a + b = 2k \geq 2$, $A(a, b) = (k, k + 1)$ if $a + b = 2k + 1 \geq 2$, or $A(a, b) = (a, b)$ if $a + b < 2$.
- (iii) As to list $L : a_1, a_2, \dots, a_m$, potential of list L is defined as $P(L) = a_1^2 + a_2^2 + \dots + a_m^2$.
- (iv) Assume $A(a, b) = (c, d)$; we have $S_A(\langle a, b \rangle) = 2(b - d)(b - c)$, which could be indicating a small piece of length $b - d$ from the bar of length b to go down by $(b - c)$. Function $S_A(.)$ presents the potential change after an average operation (see Lemma 4).
- (v) Assume $L : a_1, a_2, \dots, a_m$ is converted into a new list $L' : a'_1, a'_2, \dots, a'_m$ after taking average operations. Let H denote all the tuples which involve in average operations and then fulfill the list transformation. Then we have $S(H) = \sum_{(a,b) \in H} S_A(a, b) = P(L) - P(L')$ to represent the sum of product.

Definition 6. A communication stage represents an average operation that happens in the connected graph. Only the nodes linked by the independent edges could exchange information and take average operations. Pairs of nodes connected by different independent edges can communicate with each other at the same time.

Through iterative average operations among the nodes, we will achieve an ϵ -balanced status. The quantity of stages needed for ϵ -balance will be analysed to reflect the complexity of network convergence.

4. Algorithm Design

We abstract the strongly connected network topology as complete graph. For data dissemination in complete graph, we propose a deterministic algorithm and analyse the stages needed for network balance in Section 4.1. Then, we present a distributed randomised algorithm in Section 4.2. Upper and lower bounds of the randomised algorithm are derived through detailed theoretical analysis in Section 5.

4.1. Deterministic Algorithm. Here, a deterministic algorithm is proposed for complete connected graph. The algorithm is described as Algorithm 1.

In the proposed deterministic algorithm, the nodes perform operations in a deterministic manner to achieve network balance. The first step is to initialise the input graph, assume the source node has a value n_1 , all other nodes have value zero. Following the parameter initialisation, the deterministic algorithm is executed in two ways, which is determined by the number of nodes with value at least two

and nodes with value zero. The operations will be iterated until the network reaches a consensus. The flowchart of the proposed deterministic algorithm is shown in Figure 2, to give a clear description of how the operations are done in a deterministic way. In the flowchart, t_1 denotes the number of nodes with value at least two while t_2 denotes the number of nodes with value zero. Finally, we have Theorem 7 to show the consumed communication stages.

Theorem 7. For complete connected graph, an algorithm exists such that after $O(\log(n/\epsilon))$ stages of real average operations, the network can reach an ϵ -balanced status. The algorithm is shown as Algorithm 1.

Proof. It is easy to see that there are at most $O(\log n)$ stages for steps (6) - (10). The upper bound for steps (11) - (13) is given below.

Let n_i be the parameter value of node i . In general, assume that $n_1 \geq n_2 \geq \dots \geq n_m$. Let n_i and n_{m-i+1} take average.

Assume that after one stage, pair n_i and n_{m-i+1} has the largest average $d_i = (n_i + n_{m-i+1})/2$, and pair n_j and n_{m-j+1} has the least average $d_j = (n_j + n_{m-j+1})/2$. We assume that $i \neq j$.

It is noted that either $(n_i - n_j)/2 \leq 0$ or $(n_{m-i+1} - n_{m-j+1})/2 \leq 0$.

$$\begin{aligned}
 d_i - d_j &= \frac{n_i + n_{m-i+1}}{2} - \frac{n_j + n_{m-j+1}}{2} \\
 &= \frac{(n_i - n_j) + (n_{m-i+1} - n_{m-j+1})}{2} \\
 &\leq \max \left(\left| \frac{n_i - n_j}{2} \right|, \left| \frac{(n_{m-i+1} - n_{m-j+1})}{2} \right| \right) \\
 &\leq \frac{n_1 - n_m}{2}.
 \end{aligned} \tag{2}$$

After t stages, the difference of the nodes is at most $(n_1 - n_m)/2^t$, such that after $O(\log(n/\epsilon))$ stages, an ϵ -balanced status can be achieved. \square

4.2. Distributed Randomised Algorithm. The following part develops a distributed randomised algorithm (DRA), shown as Algorithm 2.

The flowchart of the proposed deterministic algorithm is shown in Figure 3, to give a clear description of how the operations are done in a random manner.

During the process of initialisation, related parameters as well as the settings (including the values of nodes, the maximum number of replicas, and number of vehicles of different types) in the network are set. Then, we have to determine how the replication strategy is carried out. For each node in sending status, it randomly selects a neighbour node to send the communication request. The receiving node would choose a node with the largest gap to take specific average operations according to the capabilities of vehicles. The algorithm would execute an iterative procedure until the network reaches consensus. Finally, the algorithm returns

Input: Graph G .
Output: Graph G' , communication stages a .

- (1) Initialisation;
- (2) $a = 1$;
- (3) k nodes, values with n_1, n_2, \dots, n_k ;
- (4) Stage a (step (5) - step (13)):
- (5) sort n_1, n_2, \dots, n_k in descending order;
- (6) **if** the nodes with value no smaller than two are more than the ones with value zero **then**
- (7) let the latter ones take operations with the former ones;
- (8) **else**
- (9) take operations the other way around.
- (10) **end if**
- (11) **if** there is no node with value at least two or with value zero **then**
- (12) Let n_i and n_{m-i+1} take average for $i = 1, 2, \dots, \lfloor m/2 \rfloor$;
- (13) **end if**
- (14) Enter into the next stage, $a = a + 1$;
- (15) End of Algorithm.

ALGORITHM 1: Deterministic Algorithm.

Input: graph G ;
Output: graph G' , parameter a .

- (1) Initialisation;
- (2) Let $a = 0$;
- (3) Let i indicate a vehicular node;
- (4) Let n_i indicate the distribution task of node i ;
- (5) Let n'_i indicate the new distribution task of node i ;
- (6) **repeat**
- (7) $a = a + 1$;
- (8) Each vehicular node at sending status randomly chooses a vehicular node within its neighbourhood, then sends the communication request;
- (9) Each vehicular node at receiving status selects the node from the received request if the gap between the two nodes is the largest;
- (10) Take pairwise proportional average operations for the corresponding pairs of nodes;
- (11) Let i and j indicate the vehicles who take average with each other;
- (12) New values of node i and j are updated as $n'_i = (n_i + n_j) \times N_{type_i} / (N_{type_i} + N_{type_j})$, and $n'_j = (n_i + n_j) \times N_{type_j} / (N_{type_i} + N_{type_j})$, respectively;
- (13) **until** $(|n_i - n_j| \leq \epsilon)$
- (14) End of Algorithm.

ALGORITHM 2: Distributed randomised algorithm.

graph G' and the number of average stages when the network reaches the ϵ -balanced status.

To analyse the effectiveness of the randomised algorithm, we derive several important theorems based the famous Chernoff bounds [33]. The new theoretical results are shown as Theorems 8 and 9 and Corollary 10. The proof makes our entire study self-contained.

Theorem 8 (see [8]). *Let X_1, \dots, X_n be n independent random 0-1 variables, where X_i takes 1 with probability at least p for $i = 1, \dots, n$. Let $X = \sum_{i=1}^n X_i$, and $\mu = E[X]$. Then for any $\delta > 0$, $\Pr(X < (1 - \delta)\mu) < e^{-(1/2)\delta^2 \mu}$.*

Theorem 9 (see [8]). *Let X_1, \dots, X_n be n independent random 0-1 variables, where X_i takes 1 with probability at most p for*

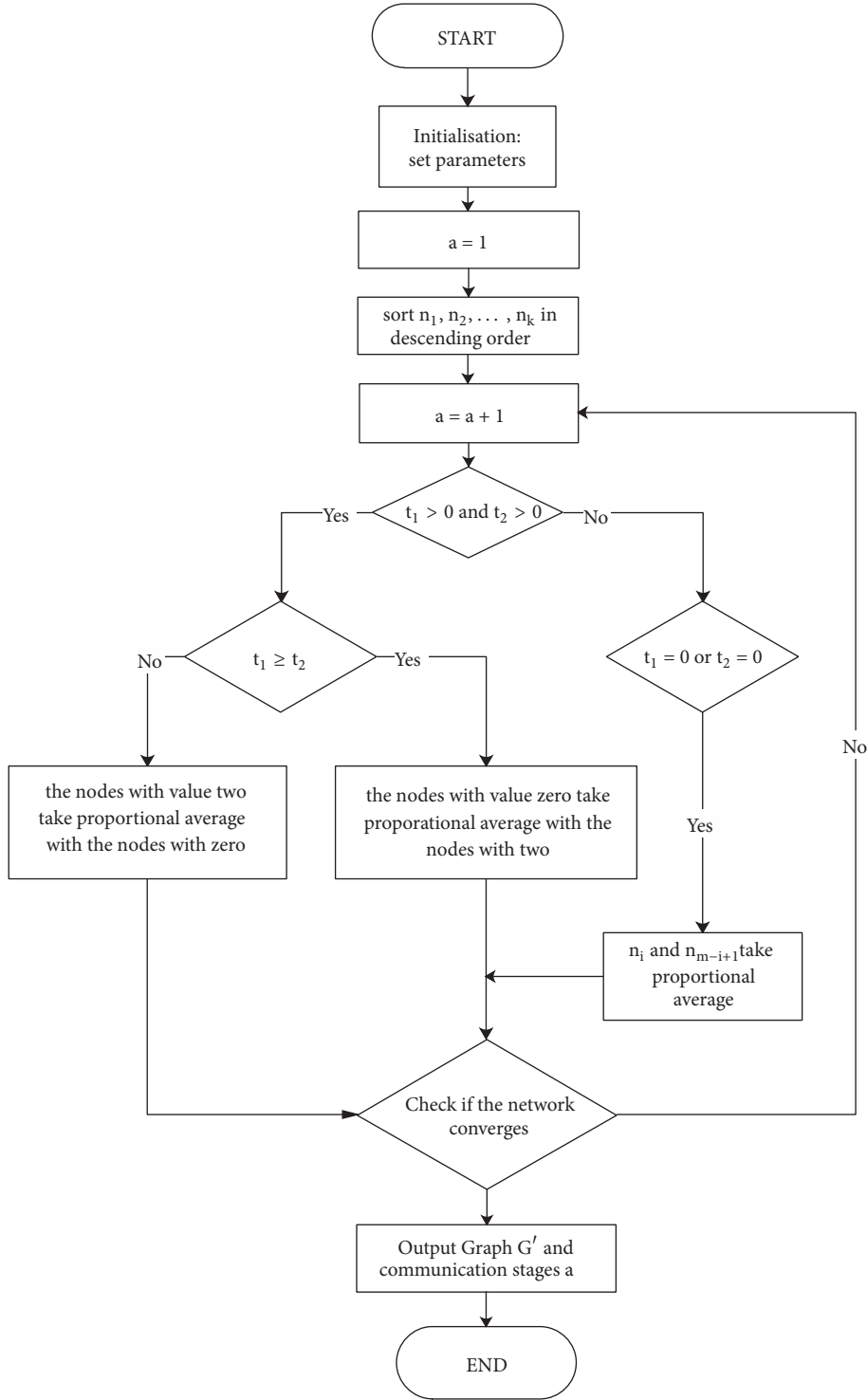


FIGURE 2: Flowchart of proposed deterministic algorithm.

$i = 1, \dots, n$. Let $X = \sum_{i=1}^n X_i$. Then for any $\delta > 0$, $\Pr(X > (1 + \delta)pn) < [e^\delta / (1 + \delta)^{(1+\delta)}]^{pn}$.

Corollary 10 (see [34]). Let X_1, \dots, X_n be n independent random 0-1 with X being the sum of $X_i, i = 1, \dots, n$.

(1) If X_i takes 1 with probability at most p for $i = 1, \dots, n$, then for any $1/3 > \epsilon > 0$, $\Pr(X > pn + \epsilon n) < e^{-(1/3)\epsilon n^2}$.

(2) If X_i takes 1 with probability at least p for $i = 1, \dots, n$, then for any $\epsilon > 0$, $\Pr(X < pn - \epsilon n) < e^{-(1/2)\epsilon n^2}$.

5. Performance Analysis

In each time step, every node becomes active with probability $1/2$ independently. Consider a node i that is active; let $d(i)$

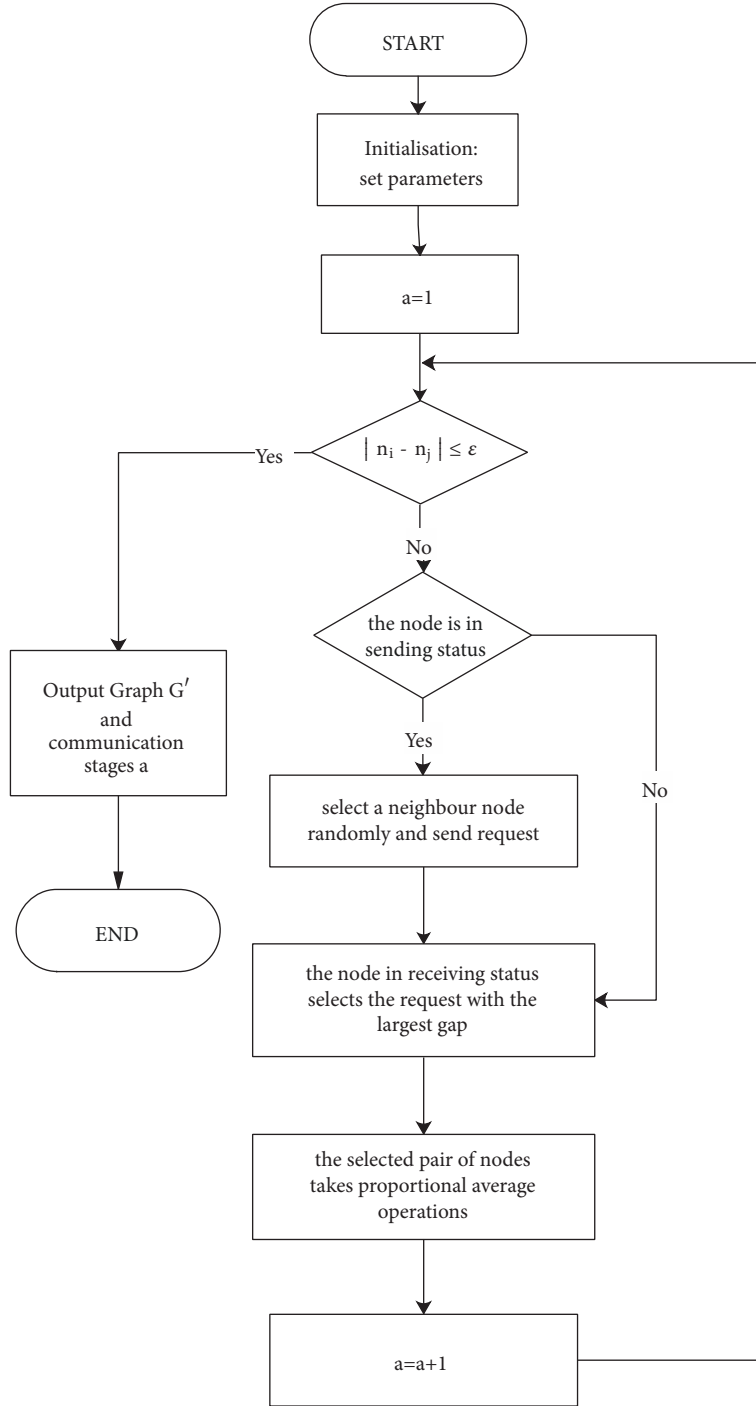


FIGURE 3: Flowchart of proposed randomised algorithm.

be its degree; that is, the number of its neighbours. Node i selects at most one of its neighbours to contact and take the proportional average operation. Each neighbour has an equal probability to be selected, i.e., $1/d(i)$. The active nodes may receive more than one contact request and they would select one of the contact requests with the largest gap. To represent the averaging time of the randomised algorithm, we have the following theorem referring to Boyd et al. [25]. Here, the averaging time means the smallest time it takes a value to be ϵ -close to the average value in the system.

Theorem 11 (see [25]). *The averaging time of the distributed randomised algorithm described above is given as*

$$\frac{0.5 \log(1/\epsilon)}{\log(1/\lambda)} \leq T(\epsilon) \leq \frac{3 \log(1/\epsilon)}{\log(1/\lambda)}, \quad (3)$$

where $\lambda = (1/2)(1 + \lambda_2(P))$.

In the following part, an upper bound and a lower bound of the communication stages consumed for network balance are derived for the proposed randomised algorithm.

5.1. Upper Bound. Before we present the upper bound of the proposed randomised algorithm, the following concepts need to be clarified.

Definition 12. For two integers a and b ; the average operation generates two new integers (a', b') , with the value of a' being equal to $\lfloor (a + b)/2 \rfloor$, and b' is equal to the remaining value.

Definition 13. Let $L = a_1, \dots, a_k$ denote a set of real numbers. $gap(L)$ is defined as $\max_{1 \leq i, j \leq k} |a_i - a_j|$.

Definition 14. Let $\alpha > 0$, and $K = a_1, \dots, a_k$ denote a list of real numbers. Assume that K is transformed into $K' = a'_1, \dots, a'_k$ after a series of communication stages. We regard K' as an α -shrink compared with K when $gap(K')$ is within a factor $(1 - \alpha)$ of $gap(K)$.

Lemma 15 is derived to show how the gap of a list of numbers shrinks via the specific average operations.

Lemma 15 (see [8]). *Let $r(\cdot)$ be a function from $S \rightarrow S$ that $r(x)$ generates a random element in S . Assume A and B are two subsets of S satisfying $|A| \leq |B|$, and $R(A) = \{x : x \in A, r(x) \in B\}$, $H(A) = \{r(x) : x \in A, r(x) \in B\}$. Then with a probability at most*

$$g(\epsilon)^{|A||B|/|S|} + ((1 - \gamma))^{(2\gamma-1)(1-\epsilon) \cdot |B|/|S| \cdot |A|}, \quad (4)$$

we have

$$|H(A)| \leq (1 - \gamma)(1 - \epsilon) \cdot \frac{|B|}{|S|} \cdot |A|, \quad (5)$$

where γ is a constant in $(0, 1)$. Furthermore, if $|B| \geq \delta|S|$ for some fixed $\delta \in (0, 1)$ then the failure probability is at most $2(1 - a)^{|A|}$ for some fixed $a \in (0, 1)$.

Proof. Let m denote the number of elements in $R(A)$ and let n denote the number of elements in B . In subset A , with probability $|B|/|S|$, each element sends its corresponding request to an element in B . Combining with Chernoff bound, the inequality $m < (1 - \epsilon) \cdot |B|/|S| \cdot |A|$ holds with a small probability

$$\zeta_1 \leq g(\epsilon)^{|A||B|/|S|}. \quad (6)$$

Assume $\gamma \in (0, 1)$ and $\epsilon(1 - \gamma) \leq 1$. The probability that $|H(A)| \leq (1 - \gamma)m$ is

$$\zeta_2 \leq \binom{n}{(1 - \gamma)m} \cdot \left(\frac{(1 - \gamma)m}{n} \right)^m \quad (7)$$

$$\leq \frac{n^{(1-\gamma)m} e^{(1-\gamma)m}}{((1 - \gamma)m)^{(1-\gamma)m}} \cdot \left(\frac{(1 - \gamma)m}{n} \right)^m \quad (8)$$

$$\leq \left(\frac{(1 - \gamma)m}{n} \right)^{(2\gamma-1)m} \quad (9)$$

$$\leq ((1 - \gamma))^{(2\gamma-1)m}. \quad (10)$$

Combining inequalities (6) and (10), the failure probability is at most $\zeta_1 + \zeta_2 \leq g(\epsilon)^{|A||B|/|S|} + ((1 - \gamma))^{(2\gamma-1)(1-\epsilon) \cdot |B|/|S| \cdot |A|}$. Thus the lemma is proved. \square

Lemma 16. *Let S be the list of all m elements that will take average operations. For some fixed $\alpha > 0$, with the failure probability not larger than $1/(\log m)^3$, the following conclusions hold.*

- (1) After $O(\log m)$ stages of average operations, there is an α -shrink.
- (2) After $O(\log m)$ stages of integer average operations, there is an α -shrink if $gap(L)$ is at least H for some H to be large enough.

Proof. Let $h = \max S - \min S$, $a = \min\{S\}$, and $b = \max\{S\}$; the median is $(a + b)/2$, which is also equal to $a + h/2$. A and B denote the sets of elements greater than and not larger than the median $(a + h/2)$ of $\min\{S\}$ and $\max\{S\}$, respectively. In general, assume $|A|$ is not larger than $|B|$. Let $A_0 = A$, $B_0 = B$, $S_0 = S$, and $j = 0$. Three periods of communication stages will be discussed in the following part.

If $|A_j| \geq (\log m)/(\log \log m)^5$, enter Period 1 below, or otherwise, enter Period 3.

Period 1. $O(1)$ communication phases will be performed as follows.

Use A_{i+1} to represent a set of elements a , which satisfies one of the following conditions:

- (1) $a \in A_i$; a does not participate in any average operation;
- (2) a is one of the elements generated by averaging elements c and d , where c and d belong to set A_i .

Use B_{i+1} to represent a set of elements a , which satisfies one of the following conditions:

- (1) $a \in B_i$; a does not participate in any average operation;
- (2) a is one of the elements generated by averaging elements c and d , where c and d belong to set B_i .

Assume that $S_{j+1}^{(1)} = A_{j+1} \cup B_{j+1}$.

Then, select three constants $\tau_1 > \tau_2 > \tau_3 > 0$ with the relation $\tau_1 = 2\tau_2 = 4\tau_3$, and constants $0 < \gamma_1, \gamma_2 < 1$. For analysis, γ_1 is set to 0.05, and γ_2 is set to 0.1. It is assumed that $\epsilon \leq \gamma_1$. According to Lemma 15, for constant $\beta \in (0, 1)$, the failure probability of $|A_{i+1}| \leq (1 - \beta)|A_i|$ is not larger than $2(1 - a)^{|A_i|} \leq 2(1 - a)^{(\log m)/(\log \log m)^5}$. Choose an integer i that makes $|A_i| \leq (1 - \beta)^i |A_0| \leq \gamma_1 |A_0|$ hold. After i stages, we have the following inequalities.

$$|A_i| \leq (1 - \beta)^i |A_0| \leq \gamma_1 |S_0|, \quad (11)$$

$$|B_i| \geq (1 - \gamma_1) |S_0| \text{ and}, \quad (12)$$

$$\max\{B_i\} \leq a + (1 - \tau_1)h \text{ with } \tau_1 \in (0, 1). \quad (13)$$

Its failure probability is at most $2i(1 - a)^{(\log m)/(\log \log m)^5}$.

D_3 is represented as the set of elements that belong to $(a + (1 - \tau_3)h, a + h)$ in set $S_j^{(1)}$. If $|D_3| \geq (\log m)/(\log \log m)^5$, the process will execute Period 2, or otherwise, Period 3 will be executed.

Period 2. $O(\log m)$ communication phases will be performed as follows.

$S_{i_1}^{(1)}$ is defined to represent the final set generated by the set S from period 1 after a series of communication operations. $S_0^{(2)}$ is defined as $S_{i_1}^{(1)}$. $S_j^{(2)}$ is used to represent the set of elements via j phases in this period.

$D_{j,3}$ is defined to represent the list of elements with values in the range of $(a + (1 - \tau_3)h, a + h]$ after j stages, while $D'_{j,3}$ indicates the elements in the range of $[a, a + (1 - \tau_3)h]$. Similarly, $D_{j,2}$ and $D'_{j,2}$ represent the elements in $(a + (1 - \tau_2)h, a + h]$ and $[a, a + (1 - \tau_2)h]$ in $S_j^{(2)}$ after j stages, respectively. As it is stated, $|D_{j,3}| \geq (\log m)/(\log \log m)^5$ always holds in Period 2.

The number of elements that belong to $[a + (1 - \tau_1)h, a + h]$ is at most $\gamma_1 m$, thus they can help up to $\gamma_1 m$ elements (the value of these elements is at most $a + (1 - \tau_1)h$) increase to at least $a + (1 - \tau_2)h$. The reason is that each element with a maximum value $a + h$ can contribute up to $\tau_2 h$. For the $\gamma_1 m$ elements with values greater than $a + (1 - \tau_1)h$, the contribution of these elements is at most $\gamma_1 m \cdot \tau_2$. Thus, the quantity of elements in the range of $[a, a + (1 - \tau_2)h]$ is at least $(1 - \gamma_2)m$. According to Lemma 15, after each phase of operation, the number of elements in set D_3 would be decreased in a fixed rate, as each element in D_3 has a greater probability to be able to do operations with the elements in set $D'_{j,2}$.

Let P_1 be the probability that the number of elements in $D_{j,3}$ that select elements in $D_{j,2}$ to take average is less than $(\gamma_2 + \epsilon)|D_{j,3}|$, and let P_2 be the probability that the number of elements in $D_{j,3}$ that take average with the ones in $D'_{j,2}$ is less than $(1 - \gamma)(1 - \epsilon) \cdot (|D'_{j,2}|/|S|)|D_{j,3}|$. According to Theorem 9 and Lemma 15, we have $P_1 \leq g(\epsilon)^{|D_{j,3}|}$ and $P_2 \leq g(\epsilon)^{|D_{j,3}|/2} + 2(1 - a)^{|D_{j,3}|}$. Thus, the elements in $D_{j+1,3}$ will be reduced.

Accordingly, with a failure probability no smaller than $2(1 - a)^{|D_{j,3}|} + g(\epsilon)^{|D_{j,3}|/2}$, that is, $o(1/(\log m)^4)$, the number of elements in $D_{j,3}$ is decreased by not smaller than

$$\begin{aligned} & (1 - \gamma_1)(1 - \epsilon) \cdot \frac{|D'_{j,2}|}{|S|} |D_{j,3}| - (\gamma_2 + \epsilon) |D_{j,3}| \\ & \geq (1 - \gamma_1)(1 - \epsilon)(1 - \gamma_1) |D_{j,3}| - (\gamma_2 + \epsilon) |D_{j,3}| \quad (14) \\ & \geq ((1 - \gamma_1)(1 - \epsilon - \gamma_1) - (2\gamma_1 - \gamma_1)) |D_{j,3}| \\ & \geq (1 - 9\gamma_1) |D_{j,3}|. \end{aligned}$$

It can be seen that $O(\log m)$ communication stages are needed to reach stage j . Once $|D_{j,3}| < (\log m)/(\log \log m)^5$, enter Period 3.

It is easy to verify the case for all integer averages when the initial gap is big enough. We just need to set up the gap such that $\tau_2 \text{gap}(K) \geq \tau_2 H \geq 3$. There is a $\tau_2 \text{gap}(K)$ gap from the new list to the old list K .

Period 3. $O((\log \log m)^2)$ communication phases will be performed as follows.

The set produced from $S_2^{(2)}$ after corresponding operations is denoted as $S_{i_2}^{(2)}$. The equalities $|D_{i_2,3}| < (\log m)/(\log \log m)^5$ and $|D'_{i_2,2}| \geq (1 - \gamma_2)|S|$ hold.

Assume that with a very small failure probability, the elements in $D'_{i_2,2}$ in sending status would select elements from B . Meanwhile, with a probability not larger than $3/4$, the elements fail to do average with those in $D'_{i_2,2}$. If an element has sent the requests for μ times, then the probability that it would not select an element from $D'_{i_2,2}$ is not larger than $(3/4)^\mu$. The probability that an element has no more than $t/4$ stages to send requests is not larger than $g(1/4)^{t/2}$.

Thus, the probability that an element in $D_{i_2,3}$ would not do average with that in $D'_{i_2,2}$ is no greater than $p_3 = g(1/4)^{t/2} + (3/4)^{t/4}$. The probability that two elements in $D_{i_2,3}$ would do average with the same element is no greater than $O((\log m)^3/m^2)$.

Consequently, the probability that the number of stages t in Period 3 is selected to be $(\log \log m)^2$ will not be larger than $|A^*|p_3 \leq 1/(\log m)^4$.

To summarise from the above analysis, the failure probability is not greater than $1/(\log m)^3$. \square

Definition 17. We treat the communication that includes c stages as α -successful when there is an α shrink in aspect of gap. For further analysis, let parameter δ indicate the probability when it fails to achieve a shrink within α .

The following Lemma 18 is derived in our previous work [8]; thus proof of the lemma is omitted here. Based on this lemma, we present Theorem 19.

Lemma 18 (see [8]). *c indicates a parameter. Partition the communication stages into groups with each containing c stages, which are denoted as G_1, G_2, \dots, G_k . Then there are k independent 0, 1 random variables r_i for each group G_i such that*

- (1) $\Pr(G_i \text{ is } \alpha\text{-successful}) \geq \Pr(r_i = 1)$
- (2) $\Pr(r_i = 1) \geq 1 - \delta$.
- (3) $\Pr(\text{there are at least } t \text{ } G_i \text{ to be } \alpha\text{-successful}) \geq \Pr(r_1 + r_2 + \dots + r_k \geq t)$.

We use m to indicate the quantity of vehicles. The corresponding value of node i is n_i^* when the network converges.

Theorem 19. *The following conclusions will be achieved after $O((\log n)(\log m)/\epsilon)$ stages by applying the proposed randomised algorithm.*

- (1) If $m \geq (1 + \epsilon)n$, then each n_i^* is either 1 or 0
- (2) If $m \leq (1 - \epsilon)n$, then each $n_i \geq 1$ and has a difference not greater than two between every two of them
- (3) If $(1 - \epsilon)n < m < (1 + \epsilon)n$, then $0 \leq n_i \leq 2$, and the numbers of value zero and two are both no greater than ϵm .

Proof. Applying Lemma 18, and Chernoff bound, it can be known that a $O(1)$ gap could be achieved after $O((\log n)(\log m))$

stages. And apply Lemma 16 $O((\log n)(\log m))$ stages if the difference is bounded by a fixed integer. The following cases are presented if the gap is $O(1)$.

- (i) $m \geq (1 + \epsilon)n$: the quantities of vehicles with value zero and two are at least ϵn in the same condition. We bound the largest elements with a constant. In this case, the number of items of 0 and 1 is at least $(1 + \epsilon)n/2$.
- (ii) $(1 - \epsilon)n < m < (1 + \epsilon)n$: when the number of vehicles with value greater than zero is at least $(1 - \epsilon)n$, the vehicles with value three will be removed via $O(\log n)$ stages. Thus, $0 \leq n_i \leq 2$ and no more than ϵn vehicles have the value zero, and no more than ϵn with two.
- (iii) $m < (1 - \epsilon)n$: after $O(\log n)$ stages, not less than γn elements are at least two as to constant $\gamma > 0$.

When the number of vehicles with value two is not less than γn , the number of zero would be decreased by θn for constant θ . And after $O(\log n)$ stages, all the zero items will be removed. When two vehicles are with gap of at least two encounters, their values will be updated such that the gap will not be greater than 1.

Assume S indicates the set of the rest of the items, and $a = \max\{x : x \in S\}$, and $b = \min\{x : x \in S\}$. $\|x\|_S$ is defined as the value of x , and it is assumed that $\|a\|_S \leq \|b\|_S$. Additionally, let $gap(S) > 2$ and $|\{x : x \in S \text{ and } x \geq a + 2\}| \geq \gamma k$. The number of a that selects value not smaller than $a + 2$ is δm . In this way, value a should disappear after $O(\log m)$ stages.

□

To show the efficiency of real average operations, Theorem 20 is proved.

Theorem 20. Assume m denotes the quantity of vehicles. Then the randomised algorithm takes $O((\log n)(\log m)/\epsilon)$ stages to enter into an ϵ -balanced status.

5.2. Lower Bound

Theorem 21. For a complete connected graph, $\Omega(\log n)$ stages are needed to reach an ϵ -balanced network status.

Proof. Referring to the proposed randomised algorithm, it may only double the number of elements via each phase. Consequently, the number of communication stages consumed is $\Omega(\log n)$ if the quantity of vehicular nodes m is more than the number of replicas n . □

6. Simulation and Analysis

We conduct our simulations by using NS-3 simulator. Consider the complexity of performance evaluation; we use the OpenStreetmap [35] to extract an area for simulation. The size of the selected area is set to 2000 m \times 2000 m, the satellite map of which is shown as Figure 4(a). Meanwhile, we use

TABLE 1: Simulation settings.

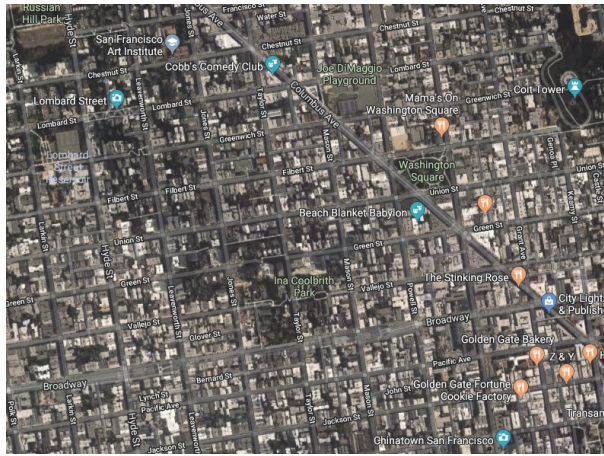
Parameters	Settings
Size of simulation area	2000m \times 2000m
Simulation time	1 hour
MAC Protocol	IEEE 802.11p
Packet size	512 bytes
Vehicle communication range	300m
Vehicle velocity	30 - 60 km/h
Number of vehicles	600 - 800
Number of data replicas	400 - 800
Shadowing model	Lognormal Shadowing model
Path loss model	Two-ray
SNR threshold	4 dBm

SUMO [36] to transform the extracted area into a simplified road network presented as Figure 4(b). We also use SUMO to generate the movement trajectories of vehicles, which will be input in the simulator to describe the vehicles' movement.

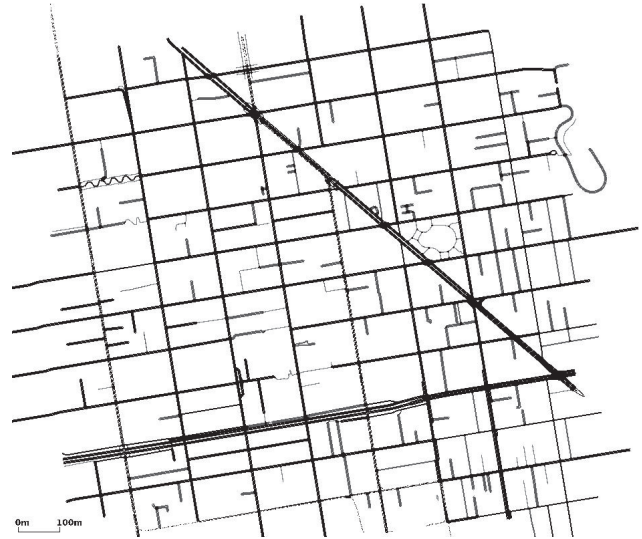
In the simulations, the number of vehicles is varied from 600 to 800 with a step length 50 to reflect a dense vehicular environment. The vehicle velocity is varied from 30 to 60 km/h that follows a normal distribution. The number of message copies is varied from 400 to 800 with a step length 100. The packet size is set to 512 bytes. The transmission range of vehicles is set to 300 m. As to the communication protocol, IEEE 802.11p is adopted to guarantee the reliability of information transmission. The two-ray path loss model is applied in the simulation as the model can calculate both the direct path and the ground reflection path. The signal-to-noise ratio (SNR) threshold is set to 4 dBm. Considering the impact of obstacles on wireless signal in urban environment as well as vehicular mobility, we apply the Lognormal Shadowing model that is suitable for the proposed scenarios. The list of simulation parameters is shown in Table 1.

6.1. Compared Algorithms. To evaluate the performance of the proposed replication-based randomised algorithm, we compared it with several data dissemination algorithms in vehicular networks, which are described below.

- (i) Constrained Capacity Replication (CCR): CCR is a distributed algorithm which can assist the vehicles to select data replication strategy autonomously according to the current network capacity.
- (ii) DOVE: DOVE controls the number of receivers in data dissemination and transforms the problem to the processor scheduling problem by utilising road layout and traffic information.
- (iii) EDDA: EDDA considers the common urban and highway scenarios. It selects all the independent pairs of nodes firstly, and then takes average operations between the corresponding nodes. The average operations will run iteratively until the network converges.
- (iv) Our proposed randomised algorithm: the proposed randomised algorithm considers the heterogeneous



(a) Google map



(b) Road topology layout

FIGURE 4: Selected area.

properties of vehicles while controlling the number of data replicas. Each node in sending status randomly selects one of its neighbours to send a contact request while the receiving node selects a request with the largest gap to take the proportional average operation.

6.2. *Performance Metrics.* We evaluate the performance of these algorithms according to the following metrics.

- (i) Number of communication stages: it indicates the average operations for the network to be balanced, which can represent the communication overhead of data dissemination to a certain extent. Meanwhile, it can also reflect the number of data transmissions for network balance as well as the network convergence complexity.
- (ii) Dissemination delay: it presents the consumed time to obtain network balance, which can be utilised to measure effectiveness of the algorithms.
- (iii) Packet delivery ratio: it can directly indicate how many vehicles could receive the replicas, as well as reflecting the dissemination performance.
- (iv) Throughput: it is used to evaluate the proposed randomised algorithm when the capabilities of vehicles are considered.

6.3. *Impact of Number of Vehicular Nodes.* To evaluate the performance of the proposed randomised algorithm with different network densities, we vary the number of vehicles from 600 to 800 to represent increasing network size.

The experimental performance analysis for the consumed communication stages is depicted in Figure 5. The proposed randomised algorithm outperforms other compared data dissemination schemes when the number of vehicles increases

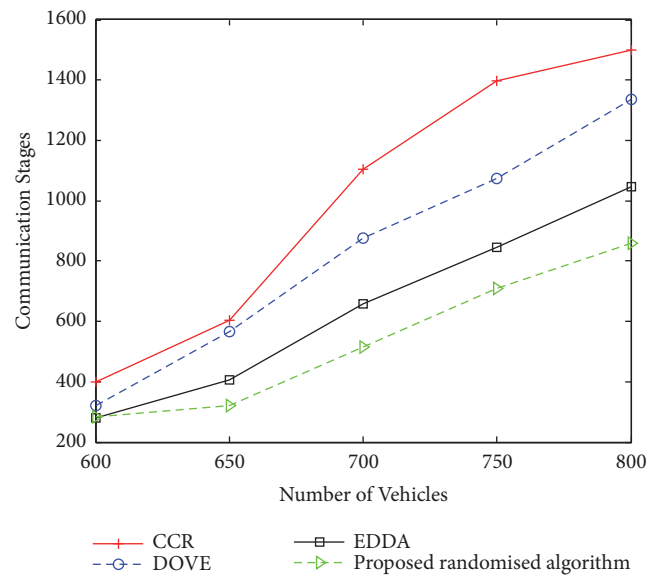


FIGURE 5: Communication stages comparison when the number of vehicles varies.

from 600 to 800, which means it needs fewer average operations to achieve network balance. CCR mainly considers network capacity to determine the replication limit while DOVE utilises road layout and traffic information to reach a desired number of vehicular receivers and minimise the dissemination delay. Our proposed randomised algorithm benefits from strong connectivity of the dense vehicular scenarios such that it can obtain network convergence with fewer communication stages than other schemes. As EDDA is developed for scenarios with normal urban density and highway, it is slightly inferior to the randomised algorithm. With the increasing number of vehicular nodes, the consumed communication stages grow for all the compared algorithms.

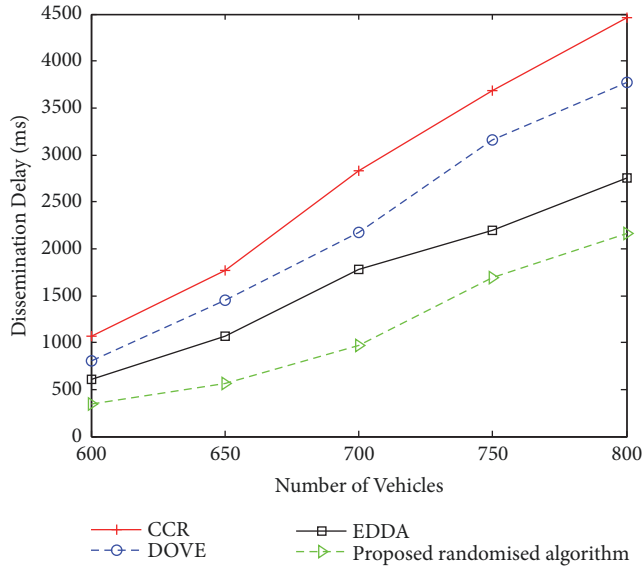


FIGURE 6: Dissemination delay comparison when the number of vehicles varies.

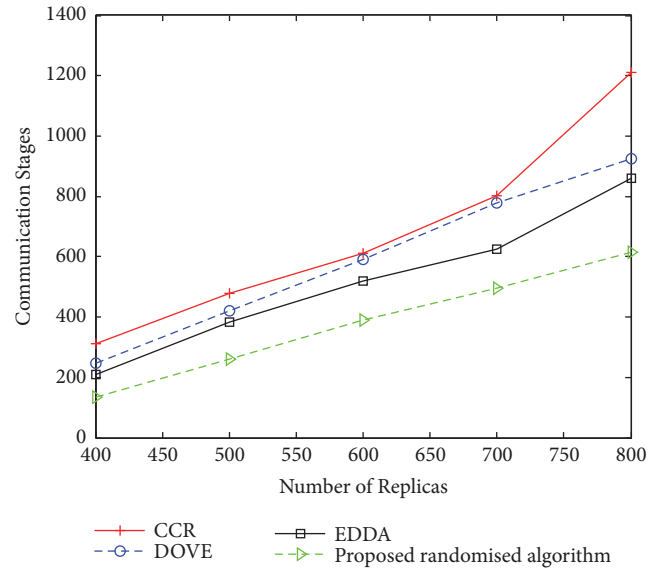


FIGURE 8: Communication stages comparison when the number of replicas varies.

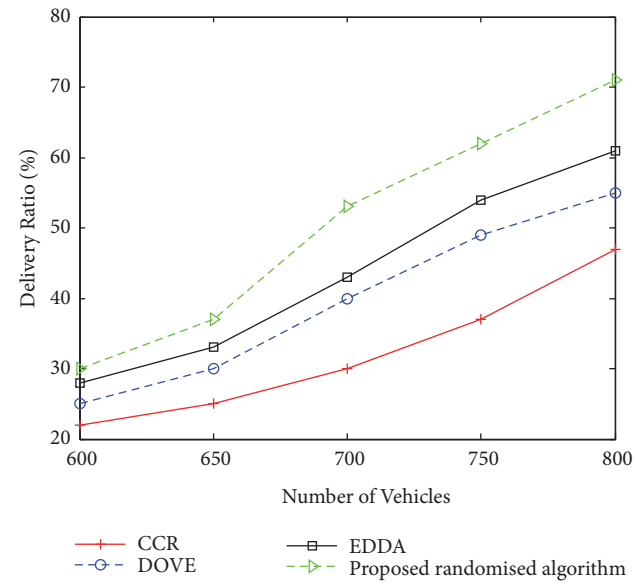


FIGURE 7: Delivery ratio comparison when the number of vehicles varies.

The variation of dissemination delay of the compared algorithms when the number of participating vehicles increases is shown in Figure 6. As is shown in the figure, when the traffic densities become higher, the dissemination delay increases for all the compared algorithms as it needs more time to fulfill data dissemination. The proposed randomised algorithm takes less time to complete data dissemination and realise network consensus compared to the other three schemes. This is because it considers the different capabilities of the vehicles in data dissemination and thus constructs a better replication strategy which could reduce data dissemination delay.

Figure 7 shows the performance of packet delivery ratio of the compared algorithms when the number of nodes involved

in data dissemination varies. In terms of delivery ratio, our proposed randomised algorithm presents an improvement compared with EDDA, DOVE, and CCR. Also, the delivery ratio of all the algorithms increases when the number of nodes increases from 600 to 800. More vehicles cooperatively participate in data dissemination such that the network connectivity would be enhanced. Accordingly, the successful packet transmissions will be improved by frequent vehicle communication instead of transmission failures caused by fewer forwarding vehicular nodes.

6.4. *Impact of Number of Data Replicas.* By varying the number of data replicas, we evaluate the impact of number of allowed data replicas on consumed communication stages and dissemination delay of the compared algorithms.

Figure 8 shows the changing trend of our proposed randomised algorithm and other compared algorithms with increasing number of data replicas. As to communication stages, the randomised algorithm performs fewer operations than EDDA profiting from the strongly connected network property of dense networks, while DOVE and CCR both need more stages for network balance. Additionally, as there are more data replicas to be disseminated in the network, it can be seen that more average operations are required to achieve network convergence. As a result, the four compared algorithms would consume increasing communication stages when more data replicas are spreading to the dissemination area.

The change of data dissemination delay under different number of data replicas is shown in Figure 9. As observed from the figure, the dissemination delay of all the compared algorithms increases when the number of replicas increases with a step length 100. The reason mainly lies in that more message replicas would take more communication stages for the network to be balanced, which leads to higher

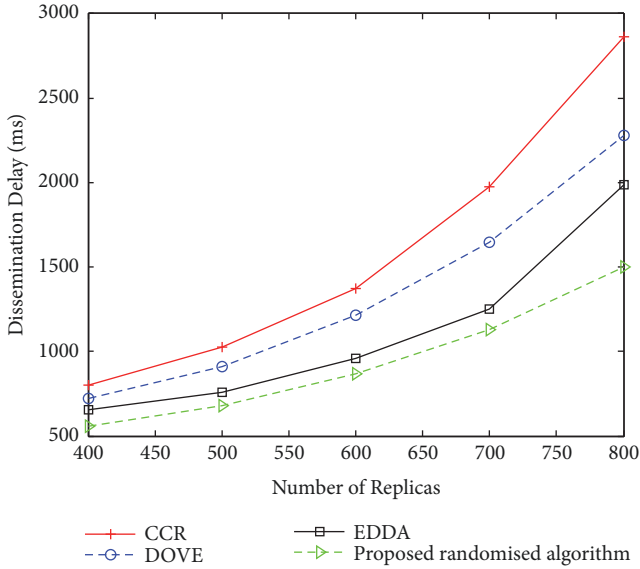


FIGURE 9: Dissemination delay comparison when the number of replicas varies.

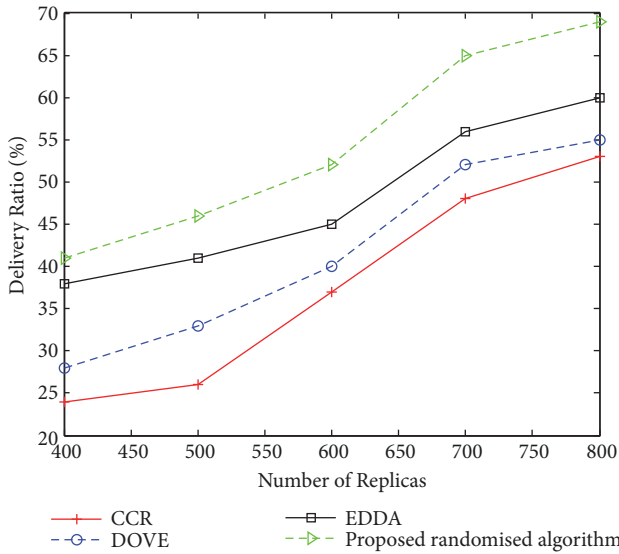


FIGURE 10: Delivery ratio comparison when the number of replicas varies.

dissemination delay. Meanwhile, data dissemination would be accelerated when the randomised algorithm is applied to the scenario as the algorithm selects pairs of nodes with the largest gap and adjusts how to do average operations according to the properties of vehicles. This is why the randomised algorithm outperforms EDDA, DOVE, and CCR.

Figure 10 depicts packet delivery ratio of the four compared algorithms varying the number of data replicas. Other than taking advantage of heterogeneous network and random average operations, the proposed randomised algorithm also benefits from better network connectivity to obtain higher delivery ratio than other algorithms. EDDA controls the number of replicas in arbitrarily connected network while

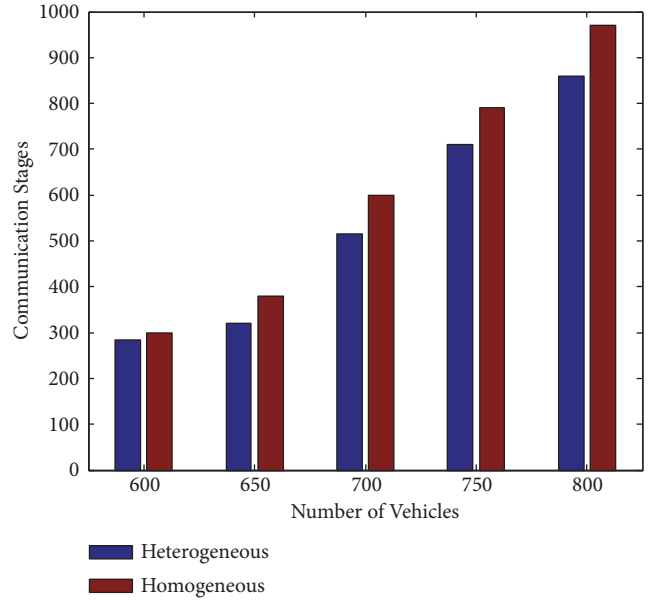


FIGURE 11: Communication stages comparison between the cases of homogeneous vehicles and heterogeneous vehicles.

CCR focuses on network capacity and DOVE wants to minimise the delay; however, they all only consider the vehicles with the same capability. The growth of delivery ratio is similar to Figure 9 that with the number of replicas changing from 400 to 800, the ratio increases for all the compared solutions. The performance comparison verifies that our proposed randomised algorithm can efficiently improve the performance of data dissemination and expedite the network convergence.

6.5. Comparing Different Versions of the Proposed Randomised Algorithm. We evaluate the communication stages and throughput of the proposed randomised algorithm in dense traffic scenario, in the case of two conditions; that is, the vehicles are homogeneous and heterogeneous. The number of vehicular nodes is varied from 600 to 800 with a step length 50. The comparison results are shown in Figures 11 and 12, respectively. In Figure 11, we can see that the number of communication stages consumed when the vehicles have the same capability is larger than the one consumed when the different capabilities of vehicles are considered. Figure 12 presents the total data sent during data dissemination. In the figure, we can see that the randomised algorithm with different capabilities achieves a 10% improvement compared with the algorithm with homogeneous vehicles. The comparison shows that the consideration of heterogeneous vehicular networks could reduce the consumed communication overhead while improving the system throughput to some extent.

7. Conclusion

To enhance data dissemination in dense vehicular scenarios, we propose a network architecture, considering heterogeneous network composed of vehicles with different capabilities. By studying data replication and network consensus

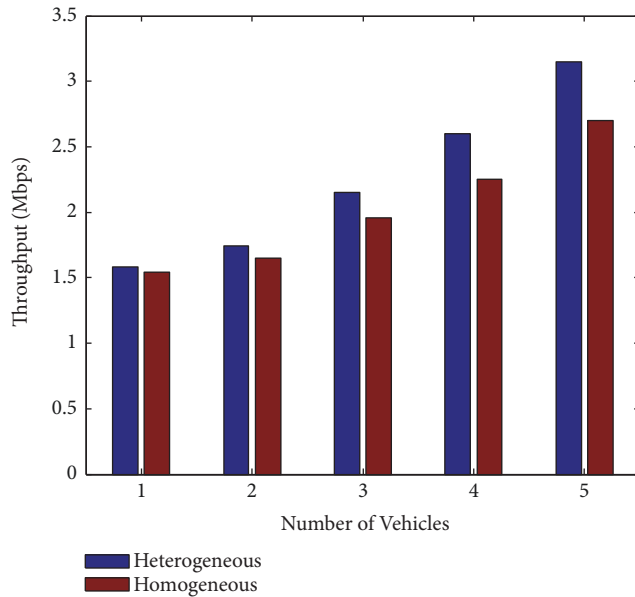


FIGURE 12: Throughput comparison between the cases of homogeneous vehicles and heterogeneous vehicles.

properties, two replication-based algorithms including a deterministic algorithm and a distributed randomised algorithm are designed. In the proposed algorithms, the vehicles take proportional average operations depending on their own capabilities. The operations will be iterated until the network converges. Mathematical analysis is derived to evaluate the complexity of network convergence. An upper bound and a lower bound of the randomised algorithm are analysed in detail. Simulation results show that the proposed randomised algorithm can reduce data dissemination delay and improve communication overhead.

8. Prospective Directions

In this study, we consider scenarios with relative ideal communication situations, which means that link interruption and other interferences are not considered. Future work should incorporate the factors that will affect data dissemination. Also, as a future prospect, we intend to enrich data dissemination mechanisms which can be adapted to complicated scenarios in IoT. Solutions that apply infrastructures such as unmanned aerial vehicles (UAVs) in cooperative networks could also be seen as a promising prospect to enhance network performance. The effectiveness of the replication-based algorithms in more IoT application scenarios needs further verification.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Disclosure

An earlier conference version of this paper [8] has been presented in 12th International Conference on WASA.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the National Science Foundation of China (no. 61772385, no. 61373040, and no. 61572370) and autonomous electric vehicle driving ability and safety evaluation technology and system development (SQ2018YFB010236-04).

References

- [1] X. Wang, Z. Ning, X. Hu et al., "A city-wide real-time traffic management system: enabling crowdsensing in social internet of vehicles," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 19–25, 2018.
- [2] R. Ghebleh, "A comparative classification of information dissemination approaches in vehicular ad hoc networks from distinctive viewpoints: A survey," *Computer Networks*, vol. 131, pp. 15–37, 2018.
- [3] Y. Li, D. Jin, P. Hui, and S. Chen, "Contact-aware data replication in roadside unit aided vehicular delay tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 2, pp. 306–321, 2016.
- [4] G. Wang, Z. Wang, and J. Wu, "A local average broadcast gossip algorithm for fast global consensus over graphs," *Journal of Parallel and Distributed Computing*, vol. 109, pp. 301–309, 2017.
- [5] R. Azimi and H. Sajedi, "A decentralized gossip based approach for data clustering in peer-to-peer networks," *Journal of Parallel and Distributed Computing*, vol. 119, pp. 64–80, 2018.
- [6] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: the single-copy case," *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 63–76, 2008.
- [7] A. Balasubramanian, B. N. Levine, and A. Venkataramani, "Replication routing in DTNs: a resource allocation approach," *IEEE/ACM Transactions on Networking*, vol. 18, no. 2, pp. 596–609, 2010.
- [8] J. Zhu, C. Huang, X. Fan, and B. Fu, "An efficient distributed randomized data replication algorithm in VANETs," in *Wireless Algorithms, Systems, and Applications*, pp. 369–380, Springer, Guilin, China, 2017.
- [9] X. Fan, C. Huang, J. Zhu, and B. Fu, "R-DRA: a replication-based distributed randomized algorithm for data dissemination in connected vehicular networks," *Wireless Networks*, pp. 1–16, 2018.
- [10] A. Torres, C. T. Calafate, J.-C. Cano, P. Manzoni, and Y. Ji, "Evaluation of flooding schemes for real-time video transmission in VANETs," *Ad Hoc Networks*, vol. 24, pp. 3–20, 2015.
- [11] A. Takahashi, H. Nishiyama, N. Kato, K. Nakahira, and T. Sugiyama, "Replication control for ensuring reliability of convergecast message delivery in infrastructure-aided DTNs," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 7, pp. 3223–3231, 2014.
- [12] M. Xing, J. He, and L. Cai, "Utility maximization for multimedia data dissemination in large-scale VANETs," *IEEE Transactions on Mobile Computing*, vol. 16, no. 4, pp. 1188–1198, 2017.

- [13] Y. Wu, Y. Zhu, H. Zhu, and B. Li, "CCR: Capacity-constrained replication for data delivery in vehicular networks," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '13)*, pp. 2580–2588, Turin, Italy, April 2013.
- [14] X. Shen, X. Cheng, L. Yang, R. Zhang, and B. Jiao, "Data dissemination in VANETs: a scheduling approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2213–2223, 2014.
- [15] T. Yan, W. Zhang, and G. Wang, "DOVE: Data dissemination to a desired number of receivers in VANET," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 4, pp. 1903–1916, 2014.
- [16] Q. Xiang, X. Chen, L. Kong, L. Rao, and X. Liu, "Data preference matters: a new perspective of safety data dissemination in vehicular ad hoc networks," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications and Networks (IEEE INFOCOM '15)*, pp. 1149–1157, Kowloon, Hang Kong, May 2015.
- [17] Z. Zhao, W. Dong, J. Bu, T. Gu, and G. Min, "Accurate and generic sender selection for bulk data dissemination in low-power wireless networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 2, pp. 948–959, 2017.
- [18] F. Chen, D. Zhang, J. Zhang et al., "Distribution-aware cache replication for cooperative road side units in VANETs," *Peer-to-Peer Networking and Applications*, vol. 11, no. 5, pp. 1075–1084, 2018.
- [19] Z. Jiang, S. Zhou, X. Guo, and Z. Niu, "Task replication for deadline-constrained vehicular cloud computing: optimal policy, performance analysis, and implications on road traffic," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 93–107, 2018.
- [20] P.-Y. Chen, S.-M. Cheng, and M.-H. Sung, "Analysis of data dissemination and control in social internet of vehicles," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2466–2477, 2018.
- [21] C. Lin, D. Deng, and C. Yao, "Resource allocation in vehicular cloud computing systems with heterogeneous vehicles and roadside units," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3692–3700, 2018.
- [22] C. Ghorai and I. Banerjee, "A robust forwarding node selection mechanism for efficient communication in urban VANETs," *Vehicular Communications*, vol. 14, pp. 109–121, 2018.
- [23] Q. Ding, X. Zeng, X. Zhang, and D. K. Sung, "A public goods game theory-based approach to cooperation in VANETs under a high vehicle density condition," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11.
- [24] G. Shi, B. Li, M. Johansson, and K. H. Johansson, "Finite-time convergent gossiping," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2782–2794, 2016.
- [25] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2508–2530, 2006.
- [26] F. Fagnani and S. Zampieri, "Randomized consensus algorithms over large scale networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 634–649, 2008.
- [27] C. Yu, B. D. Anderson, S. Mou, J. Liu, F. He, and A. S. Morse, "Distributed averaging using periodic gossiping," *Institute of Electrical and Electronics Engineers Transactions on Automatic Control*, vol. 62, no. 8, pp. 4282–4289, 2017.
- [28] J.-Y. Chen, G. Pandurangan, and D. Xu, "Robust computation of aggregates in wireless sensor networks: Distributed randomized algorithms and analysis," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 9, pp. 987–1000, 2006.
- [29] T. C. Aysal, M. E. Yildiz, A. D. Sarwate, and A. Scaglione, "Broadcast gossip algorithms for consensus," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2748–2761, 2009.
- [30] S. Wu and M. G. Rabbat, "Broadcast gossip algorithms for consensus on strongly connected digraphs," *IEEE Transactions on Signal Processing*, vol. 61, no. 16, pp. 3959–3971, 2013.
- [31] M. Franceschelli, A. Giua, and C. Seatzu, "Gossip based asynchronous and randomized distributed task assignment with guaranteed performance on heterogeneous networks," *Nonlinear Analysis: Hybrid Systems*, vol. 26, pp. 292–306, 2017.
- [32] A. Nedic and J. Liu, "On convergence rate of weighted-averaging dynamics for consensus problems," *Institute of Electrical and Electronics Engineers Transactions on Automatic Control*, vol. 62, no. 2, pp. 766–781, 2017.
- [33] R. Motwani and P. Raghavan, "Randomized algorithms," *Acm Computing Surveys*, vol. 26, pp. 48–50, 1995.
- [34] M. Li, B. Ma, and L. Wang, "On the closest string and substring problems," *Journal of the ACM*, vol. 49, no. 2, pp. 157–171, 2002.
- [35] M. Haklay and P. Weber, "Openstreetmap: user-generated street maps," *IEEE Pervasive Computing*, vol. 7, no. 4, pp. 12–18, 2008.
- [36] SUMO-Simulation of Urban Mobility, <http://sumo.sourceforge.net>.

Research Article

A Novel Task Allocation Algorithm in Mobile Crowdsensing with Spatial Privacy Preservation

Wenyi Tang,¹ Qi Jin,^{1,2} Xu Zheng,¹ Guangchun Luo ,³ Guiduo Duan,¹ and Aiguo Chen ¹

¹The School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

²Chengdu Municipal Public Security Bureau, Chengdu 610017, China

³The School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Correspondence should be addressed to Guangchun Luo; gclu.uestc@gmail.com

Received 31 January 2019; Accepted 6 March 2019; Published 1 April 2019

Guest Editor: Wei Cheng

Copyright © 2019 Wenyi Tang 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 Internet of Things (IoT) has attracted the interests of both academia and industry and enables various real-world applications. The acquirement of large amounts of sensing data is a fundamental issue in IoT. An efficient way is obtaining sufficient data by the mobile crowdsensing. It is a promising paradigm which leverages the sensing capacity of portable mobile devices. The crowdsensing platform is the key entity who allocates tasks to participants in a mobile crowdsensing system. The strategy of task allocating is crucial for the crowdsensing platform, since it affects the data requester's confidence, the participant's confidence, and its own benefit. Traditional allocating algorithms regard the privacy preservation, which may lose the confidence of participants. In this paper, we propose a novel three-step algorithm which allocates tasks to participants with privacy consideration. It maximizes the benefit of the crowdsensing platform and meanwhile preserves the privacy of participants. Evaluation results on both benefit and privacy aspects show the effectiveness of our proposed algorithm.

1. Introduction

The IoT is an efficient network that connects various devices on the Internet. It often consists of sensor-equipped devices that can sense, communicate, and react to environmental variations [1]. The developments of IoT in both academia and industry are rapid, because of its promising market value [2, 3]. A large number of IoT applications have been developed and utilized in the society, such as the smart city [4, 5], smart grid [6, 7], and smart traffic [8, 9]. Most of the IoT applications require large amounts of sensing data for monitoring and computing. Therefore, the methods of acquiring sensing data are fundamental in IoT. An efficient way to acquire large amounts of sensing data is using the mobile crowdsensing. It is a promising sensing paradigm which encourages crowds to use mobile devices to collect sensing data. Since small-sized portable mobile devices become extremely prevalent in modern society, the mobile crowdsensing reveals its high performance on the

collection of sensing data [10–12]. There is a wide range of IoT applications based on mobile crowdsensing, such as environmental monitoring [13, 14], healthcare [15], and smart cities [16, 17].

A mobile crowdsensing system typically consists of a crowdsensing platform (CSP), a set of data requesters, and a set of participants. Data requesters publish requirements of the sensing data to the CSP. The CSP segments tasks allocate segmented tasks to suitable participants and release the uploaded data from participants to requesters. The goals of the CSP are maintaining sufficient participants and maximizing its own benefit. The strategy of task allocating is crucial for the CSP, since it affects the data requester's confidence, the candidate participant's confidence and its own benefit. Specifically, participants care about their workloads and compensations. They like to collaborate with the CSP that always assigns proper sensing tasks and offers fair compensations. If a CSP always assigns improper tasks to participants, i.e., assigns participants to go far away from their

daily active regions, the CSP will lose many participants. This may decrease the CSP's capability of acquiring sensing data, lose the data requester's confidence, and reduce the CSP's benefit finally. Thus, the CSP should allocate proper tasks to participants for above reasons.

Moreover, the privacy preservation becomes important for the CSP, because the crowds care about the disclosure of their sensitive information in recent years [18–21]. Only the CSP who preserves the privacy can maintain sufficient participants. In this study, we treat the CSP as a trusted entity for participants. However, data requesters are untrusted, and we treat them as potential adversaries. They may be curious about the sensitive information of participants. For example, the requested data always associates with locations in crowdsensing. Adversaries can extract movement patterns of participants from acquired data. The movement patterns are sensitive since adversaries may be able to identify the addresses of participants' homes, schools, or working places [22]. Thus, adversaries usually choose the participants with abnormal profiles as vulnerable users and very likely execute further attacks, so that the privacy preservation is important for the task allocation as well.

Therefore, we investigate the strategy of task allocation with basic considerations and the privacy consideration. Specifically, we first formulate the problem of task allocation. This formulation carefully considers the utility of CSP and the privacy disclosure of participants. A task allocation algorithm with privacy preservation (TAPP) is proposed. It consists of three phases, allocating tasks without privacy preservation, modifying allocations with privacy consideration, and merging the allocations. Furthermore, a series of evaluations show that the proposed algorithm achieves outstanding performance on many aspects.

- (1) We first formulate the problem of task allocation with privacy preservation on the CSP's site. We utilize the relative entropy to formulate the privacy disclosure of the participants. The problem formulation is based on a series of assumptions, such as limits of the participant's total time cost and privacy disclosure.
- (2) A three-step algorithm, named TAPP, is proposed to allocate proper tasks to participants. The output allocating strategy gain a high benefit for the CSP meanwhile preserves the privacy of participants.
- (3) Extensive evaluations are executed based on real-world crowdsensing datasets. The evaluation results show TAPP performs well on maximizing the CSP's benefit and preserving the privacy of participants simultaneously.

The remaining of the paper is organized as follows. The related works are introduced in Section 2. The problem formulation is presented in Section 3. Section 4 discusses the complexity of the formulated problem and introduces the proposed algorithm. Section 5 validates the effectiveness of

the algorithm on several aspects. Section 6 concludes the paper.

2. Related Work

A large number of researchers concentrate on the task allocation in mobile crowdsensing. Traditional methods of task allocation lack privacy considerations. They make the allocation strategies according to some basic metrics, such as the quality of sensing data [23–25], the incentive cost [26, 27], the energy consumption [14, 28], and the travel distance [29–31]. The methods which focus on the quality of sensing data are mostly designed for monitoring the environment. They measure the quality of sensing data by a certain metric and attempt to maximize the data quality. The methods focus on the incentive cost allocating tasks on the site of the CSP. Xiong et al. [26] propose an incentive mechanism which minimizes the total budget of the CSP. In their study, the CSP pays according to the participant number. Zhang et al. [27] design a different incentive mechanism which assumes the CSP pays according to tasks. Xiong et al. [14] propose an energy-saving technique, named, piggybacking. It is an optimal collaborative data sensing and uploading scheme which reduces the energy consumption. The travel distance is widely considered in previous studies as well. The methods [29–31] measure the travel distances of participants by numerical values. They contain the same object to minimize the overall travel distance for all sensing tasks.

The crowds care more about the disclosure of their sensitive information in recent years [32]. The privacy preservation in mobile crowdsensing has attracted increasing research interests. Numerous preservation methods are proposed regarding to the spatial privacy, one of the important privacies. Some traditional methods [33–36] based on spatial cloaking are suitable for preserving privacy in mobile crowdsensing. These methods hide the participant spatial information by spatial transformations, generalization, or a set of dummy locations to preserve privacy. Kazemi et al. [37] propose a privacy protection method which directly applies to mobile crowdsensing. This method considers the CSP untrusted and adjusts the spatial information of a participant group. To et al. [38] adopt the differential privacy mechanism and propose a method for spatial crowdsensing task allocation. This method sets a trusted third party entity to aggregate the spatial information of participants. Wang et al. [39] propose a truthful incentive mechanism which preserves the privacy based on differential privacy and auction theory. Duan et al. [40] introduce the reverse auction to task allocation and design allocating algorithm in a novel respective.

A closely related work to ours is presented by Wang et al. [41]. They first preserve the spatial privacy on the participant's site and then allocate the tasks. This adds an extra procedure to the participants. The preservation mechanism is based on the differential privacy. In contrast, our study preserves the privacy on the CSP's site, which does not bother the participants.

TABLE I: List of main notations.

Notation	Explanation
$\{L_1, L_2, \dots, L_N\}$	the subregions
$\{W_1, W_2, \dots, W_H\}$	the participants
$\{J_1, J_2, \dots, J_N\}$	the task workloads
$\{t_{i1}, t_{i2}, \dots, t_{iN}\}$	the allocations of W_i
$P_i = \{p_{i1}, p_{i2}, \dots, p_{iN}\}$	the actual profile of W_i
$P'_i = \{p'_{i1}, p'_{i2}, \dots, p'_{iN}\}$	the observed profile of W_i

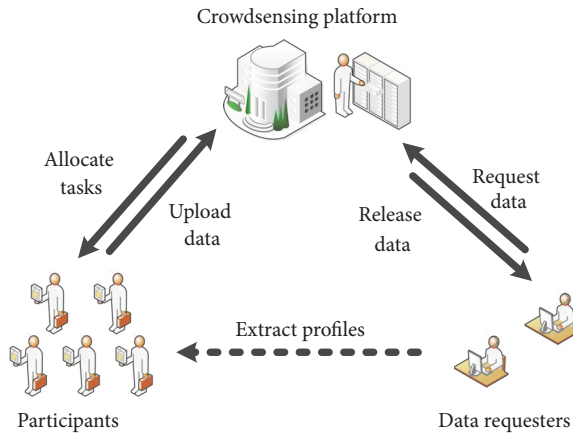


FIGURE 1: A general system model of crowdsensing.

3. Problem Formulation

This section introduces the general system model, system input, the definition of utility, and the requirements of privacy preservation. The list of main notations is shown in Table I.

3.1. System Model. A general mobile crowdsensing system consists of three main entities: participants, the CSP, and data requesters, as shown in Figure 1. In this study, we consider the task allocation problem under several specific assumptions of these entities, described as follows.

Participants. The participants receive assigned crowdsensing tasks from the CSP. Each participant actively finishes assigned tasks in time, if the tasks are not overmuch and their privacy is protected. After that, they get a reward from the CSP paid daily or monthly.

Crowdsensing Platform (CSP). The CSP is trusted by participants and knows sensitive information of participants. The CSP receives the crowdsensing requirements from the data requesters and assigns tasks to the suitable participants. The CSP releases the requested data to data requesters and gets rewards from them.

Data Requesters. The data requesters acquire data from the CSP. They may extract the sensitive information of participants from the acquired data, which leads to a privacy leakage.

Note the payment assumption of participants, we consider the scenario that each participant who finishes all assigned tasks is paid daily or monthly [42, 43]. This payment setting is helpful for the quality of acquired sensing data, since it assigns tasks regularly to fixed participants.

3.2. System Input. Assume the crowdsensing system executes the crowdsensing works in a fixed area, which consists of N subregions $\{L_1, L_2, \dots, L_N\}$. The crowdsensing system has H participants $\{W_1, W_2, \dots, W_H\}$. Each participant W_i has a personal movement pattern in real-life. We call this pattern participant's *actual profile*, defined as follows:

$$P_i = \{p_{i1}, p_{i2}, \dots, p_{iN}\}, \quad (1)$$

where $p_{ij} \in [\delta, 1)$, δ is an infinitely small quantity, and $\sum_{j=1}^N p_{ij} = 1$. We use δ instead of zero as the lower bound, because this avoids the condition $0/0$ happening when we calculate the privacy disclosure. We call L_j is the inactive subregion of W_i if $p_{ij} = \delta$, which means the participant never goes to the subregion L_j . Otherwise, we call L_j is the active subregion of W_i . The CSP acquires the actual profile of each participant, by requiring this information in the register procedure.

When the CSP receives original tasks from the data requesters, the CSP divides the original tasks into unit tasks. Each unit task requires the same time cost and associates with a subregion L_i , $i \in \{1, 2, \dots, N\}$. Set the time cost of one unit task as one for simplicity. The CSP collects all unit tasks according to associated regions; thus the workload (required time cost) in each subregion is $\{J_1, J_2, \dots, J_N\}$. $J_i = 1$ means there is only one unit task required in L_i .

The CSP allocates the unit tasks to the participants, denoted as $t_i = \{t_{i1}, t_{i2}, \dots, t_{iN}\}$. For example, t_{ij} means participant W_i needs to cost t_{ij} time to finish the allocated tasks in region L_j . Each participant has a time threshold τ_i . Assume the participants care about both the workloads and work regions. Therefore, the CSP should avoid following situations to maintain sufficient participants.

- (1) Total time cost: if the total time cost $\sum_{j=1}^N t_{ij}$ exceeds the time threshold τ_i of the participant, he/she will quit the crowdsensing system. This means the participants are assigned too much works with insufficient rewards.
- (2) Work regions: if the participant is assigned to a location where he/she never goes, i.e., $t_{ij} > \delta$, while $p_{ij} \leq \delta$, he/she will quit the crowdsensing system. This means the participants are assigned unsuitable works regarding to their actual profile.

The CSP collects the data generated by participants and sends the data to the data requester for rewards. Then, the data requesters can extract a movement pattern of each

participant from acquired data. We define this pattern as *observed profile* of each participant W_i ,

$$P'_i = \{p'_{i1}, p'_{i2}, \dots, p'_{iN}\}, \quad (2)$$

where $p'_{ij} \in [\delta, 1)$, and $\sum_{j=1}^N p'_{ij} = 1$. Each p'_{ij} is calculated as follows:

$$p'_{ij} = \frac{t_{ij}}{\sum_{k=1}^N t_{ik}}. \quad (3)$$

3.3. Utility. The utility of the CSP is its benefit. Recall the payment assumption that if a participant finishes all assigned tasks, the participant gets payment daily or monthly. If a participant has no assigned task, the participant gets no payment. The CSP gets a reward B when all the requested tasks are finished; meanwhile it pays each recruited participant $Cost$.

The utility of CSP is

$$Utility = B - Cost \left\| \left\{ W_i \mid \sum_{j=1}^N t_{ij} > \delta \right\} \right\|. \quad (4)$$

3.4. Privacy. Since the data requesters are curious and untrusted, they may be the adversaries. They infer the movement patterns of participants, which is the concerned sensitive information, from the observed profiles. The adversaries usually choose the participants with abnormal profiles as vulnerable users and very likely execute further attacks. Thus, we define the difference between an individual and its community as the privacy disclosure in this study.

Assume participant W_i is in community C_i with several other participants. The average profile of the community is

$$\bar{P}_{C_i} = \{\bar{p}_{C_i,1}, \bar{p}_{C_i,2}, \dots, \bar{p}_{C_i,N}\}, \quad (5)$$

where

$$\bar{p}_{C_{ij}} = \sum_{W_m \in C_i} \frac{p_{mj}}{|C_i|}. \quad (6)$$

Then we define the privacy disclosure of $W_i \in C_i$ as the relative entropy between P_i and \bar{P}_{C_i}

$$D(P'_i \parallel \bar{P}_{C_i}) = \sum_{i=1}^N p'_{ij} \ln \frac{p'_{ij}}{\bar{p}_{C_{ij}}}. \quad (7)$$

Furthermore, each participant may set a privacy threshold on this divergence, noted as θ_i . When CSP allocates tasks to participants, the relative entropy for each participant should be bounded by the privacy threshold to guarantee adversaries cannot learn significantly private information from the observed profiles.

3.5. Design Object. Our object is to derive an allocation scheme for the CSP, so that sufficient participants are maintained by allocating suitable tasks and preserving their privacy, meanwhile maximizing the benefit for the CSP. We formalize the problem as follows:

$$\max B - Cost \left\| \left\{ W_i \mid \sum_{j=1}^N t_{ij} > \delta \right\} \right\| \quad (8)$$

$$s.t. \quad D(P'_i \parallel \bar{P}_{C_i}) \leq \theta_i \quad \forall i \in \{1, 2, \dots, H\} \quad (9)$$

$$\sum_{j=1}^N t_{ij} \leq \tau_i \quad \forall i \in \{1, 2, \dots, H\} \quad (10)$$

$$t_{ij} = \delta \quad \forall p_{ij} = \delta \quad (11)$$

$$\sum_{i=1}^H t_{ij} \geq J_j \quad \forall j \in \{1, 2, \dots, N\} \quad (12)$$

4. Task Allocation Algorithm

In this section, we first analyze the complexity of the formulated problem. Then we introduce overview of the proposed algorithm. In the remaining parts, the main phases of the whole algorithm are presented.

4.1. Complexity. The problem of achieving the maximum benefit for the CSP following constraints (9), (10), and (11) is NP-hard.

Consider an arbitrary instance of the minimum set cover problem, consisting of an universal set $U = \{e_1, e_2, \dots, e_n\}$, series of subsets $\{S_1, S_2, \dots, S_m\}$. We construct the instance of the maximum benefit problem corresponding to the instance of the minimum set cover problem. We set the privacy threshold $\theta_i = \infty$ and time threshold $\tau_i = \sum_{j=1}^N |\{p_{ij} \mid p_{ij} > \delta\}|$ for each W_i , which means there is no privacy constraint and the time threshold equals to the number of subregions where the participant goes. We construct the workload the crowdsensing regions $L = \{L_1, L_2, \dots, L_n\}$, $J = \{J_1, J_2, \dots, J_n\}$, $\forall |J_i| = 1$ regarding the universal set U and participants $\{W_1, W_2, \dots, W_m\}$ corresponding to the subsets. Each W_i corresponding to S_i associates with the actual profile $P_i = \{p_{i1}, p_{i2}, \dots, p_{in}\}$. Set $p_{ij} = \delta$, $\forall e_j \notin S_i$; otherwise $p_{ij} = 1/|S_i|$, where $|S_i|$ is the element number of S_i . We should set the allocation of each W_i as $\{t_{i1}, t_{i2}, \dots, t_{in}\}$, $\forall p_{ij} = \delta, t_{ij} = 0$; otherwise $t_{ij} = 1$ for maximizing the benefit, for example, following our construction, given $J = \{1, 1, 1, 1, 1\}$ and a W_i with the actual profile $\{\delta, 1/3, 1/3, 1/3, \delta\}$ and $\tau_i = 3$. Since using less participants increases the benefit of the CSP, the allocation $\{0, 1, 1, 1, 0\}$ is more likely to maximize the benefit than $\{0, 3, 0, 0, 0\}$.

Thus, finding the minimum set cover equals to finding an allocation which covers all the tasks and selects the minimum number of participants, i.e., achieving the maximum benefit. As the reduction shown above, the formulated problem is NP-hard.

```

Input: participants  $\{W_1, W_2, \dots, W_H\}$ , workloads  $\{J_1, J_2, \dots, J_N\}$ , each community profile  $\{\bar{p}_{C_1}, \bar{p}_{C_2}, \dots, \bar{p}_{C_N}\}$ , each actual
profile  $\{p_{i1}, p_{i2}, \dots, p_{iN}\}$ , each time threshold  $\tau_i$ 
Output: the allocation  $\{t_{i1}, t_{i2}, \dots, t_{iN}\}$  for each  $W_i$ 
1: for each  $W_i$  do
2:   set each  $t_{ij} = 0$ ,  $j \in \{1, 2, \dots, N\}$ ,  $\Delta\tau_i = 0$ ;
3:   compute  $A_i = \{a_{i1}, a_{i2}, \dots, a_{iN}\}$  by  $P_i$ ;
4:   for each  $J_i$  do
5:     if  $J_i \leq 0$  then  $can_i = \delta'$ ;
6:     else  $can_i = \sum_{j=1}^H a_{ij} * \tau_j$ ;
7:   for each  $can_i$  do
8:     if  $can_i < J_i$  then fails;
9:   while  $\sum_{i=1}^N J_i > 0$  do
10:    update  $Can = \{can_1, can_2, \dots, can_N\}$ ;
11:     $Prio^{task} = \{can_1 - J_1, can_2 - J_2, \dots, can_N - J_N\}$ ;
12:     $k = \arg \min prio_i^{task}, prio_i^{task} \in Prio^{task}$ ;
13:    for each  $W_i$  do
14:      if  $a_{ik} == 1$  then  $\{W_i\} \rightarrow Prio_k^{part}$ ;
15:    if  $Prio_k^{part} = \emptyset$  then fails;
16:    choose  $W_l \in Prio_k^{part}$  with  $a_{lk} = 1$ , minimum  $\sum_{j=1}^N a_{lj}$ , and maximum  $\tau_l - \Delta\tau_l$ ;
17:     $t_{lk} = t_{lk} + 1$ ,  $\Delta\tau_l = \Delta\tau_l + 1$ ,  $J_k = J_k - 1$ ;
18:    if  $\Delta\tau_l == \tau_l$  then each  $a_{lk} = 0$ ;
19:    if  $\forall W_i, \Delta\tau_i == \tau_i$  and  $\exists J_i > 0$  then fails;

```

ALGORITHM 1: Task allocation without privacy preservation.

4.2. Overview of the Algorithm. Our task allocation algorithm with privacy preservation, called TAPP, runs on the site of the CSP. The CSP first receives original tasks from the data requesters. It divides the original tasks into unit tasks and merges all unit tasks according to associated subregions. Then the framework analyzes the uploaded information from participants. Then it acquires time thresholds, privacy thresholds, actual profiles, and community profiles. By collecting all the inputs, the framework makes an allocation strategy by following three phases: (i) allocating tasks without privacy preservation, as shown in Algorithm 1; (ii) modifying allocations with the privacy consideration, as shown in Algorithm 2; (iii) reducing allocated participants by merging tasks of two participants, as shown in Algorithm 3.

4.3. Task Allocation Phase. In this part, we introduce the first phase of TAPP algorithm. It iteratively picks a unit task according to the task priority and allocates the unit task to a participant according to the participant priority. The algorithm only considers the time constraints of participants and ignores the privacy constraints. Generally, the algorithm properly allocates all tasks to participants with no consideration of the privacy preservation.

Combined with Algorithm 1, the algorithm first initializes the allocations $\{t_{i1}, t_{i2}, \dots, t_{iN}\}$, the total workload $\Delta\tau_i$, and the set $A_i = \{a_{i1}, a_{i2}, \dots, a_{iN}\}$ for each W_i . Each $a_{ij} \in A$ indicates the active or inactive subregion of W_i ; i.e., if $p_{ij} > \delta$, $a_{ij} = 1$; otherwise $a_{ij} = 0$. Then the algorithm computes the set $Can = \{can_1, can_2, \dots, can_N\}$ in Lines 4~6, where δ' is a very large number. Each can_i denotes

the total available time of participants who are active in subregion L_i . It checks the worst case that the tasks are unfinished, even if all participants cost all of their time in one subregion (Lines 7 and 8). After this, it allocates one union task to a participant step by step, until all tasks are allocated.

In each step, the algorithm chooses a subregion and a participant for task allocation. Specifically, it first updates the set Can by the method shown in Lines 4~6. Then it computes the priority of tasks associated with different subregions. The priority set is $Prio^{task} = \{prio_1^{task}, prio_2^{task}, \dots, prio_N^{task}\}$, where $prio_i^{task} = can_i - J_i$. We choose $prio_i^{task}$ indicates the priority, since the tasks in a subregion with minimum excess should first be assigned. For example, assume some $J_i = 10$. The subregion L_i is the active subregion only for participants W_j and W_k , which means only $a_{ji} = 1$ and $a_{ki} = 1$. The time threshold of W_j and W_k are 5, respectively. Thus we can only allocate them spending all their time to finish tasks in L_i for a feasible allocation. So that the algorithm chooses $prio_k^{task}$ associated with L_k , the minimum one in $Prio^{task}$. For each participant who contains active subregion L_k , it chooses the participant W_l that contains minimum number of active regions and the maximum rest time. Then, the algorithm allocates a unit task from J_k to W_l (Line 17). It sets all $a_{ij} \in A_i$ for W_i , when the workload for W_i equals the time threshold. This makes the participant who has no rest time never be chosen anymore.

The allocation iterates until all the tasks are allocated. Otherwise, it fails (Line 19). The allocation loop is the main part of this algorithm, which costs $O(H \cdot \sum_{i=1}^N J_i)$ or $O(N \cdot$

Input: subregions $\{L_1, L_2, \dots, L_N\}$, participants $\{W_1, W_2, \dots, W_H\}$, each community profile $\{\bar{P}_{C_1}, \bar{P}_{C_2}, \dots, \bar{P}_{C_N}\}$, each workload $\Delta\tau_i$

Output: the allocation $\{t_{i1}, t_{i2}, \dots, t_{iN}\}$ for each W_i

- 1: update each $A_i = \{a_{i1}, a_{i2}, \dots, a_{iN}\}$ by P_i ;
- 2: **for** each W_i that $\Delta\tau_i > 0$ **do**
- 3: **if** $D(P'_i \parallel \bar{P}_{C_i}) > \theta_i$ **then** $\{W_i\} \rightarrow \text{Danger}$;
- 4: each $\{W_i \mid \Delta\tau_i < \tau_i \text{ and } W_i \notin \text{Danger}\} \rightarrow \text{Safe}$;
- 5: sort $W_i \in \text{Danger}$ by $\Delta\tau_i$ in descending order;
- 6: sort $W_j \in \text{Safe}$ by $\tau_j - \Delta\tau_j$ in ascending order;
- 7: **for** each $W_i \in \text{Danger}$ **do**
- 8: compute $\text{case}_i^- = \{c_{i1}^-, c_{i2}^-, \dots, c_{iN}^-\}$;
- 9: **while** $\exists c_{ik}^- < 0$ **do**
- 10: $l = \arg \min c_{ik}^-$, $S_1 = \emptyset$, $S_2 = \emptyset$;
- 11: **for** each $W_j \in \text{Safe}$ **do**
- 12: **if** $a_{jl} > 0$ and $\Delta\tau_j < \tau_j$
- 13: **then** compute c_{jl}^+ ;
- 14: **else continue**;
- 15: **if** $c_{jl}^+ \leq 0$ **then** $\{W_j\} \rightarrow S_1$;
- 16: **else if** $D(P'_j \parallel \bar{P}_{C_j}) + c_{jl}^+ \leq \theta_j$
- 17: **then** $\{W_j\} \rightarrow S_2$;
- 18: **if** $S_1 \neq \emptyset$ **then**
- 19: $q = \arg \max(\tau_j - \Delta\tau_j), W_j \in S_1$;
- 20: **else if** $S_2 \neq \emptyset$ **then**
- 21: $q = \arg \max(\tau_j - \Delta\tau_j), W_j \in S_2$;
- 22: **else fails**;
- 23: $t_{ql} = t_{ql} + 1$, $\Delta\tau_q = \Delta\tau_q + 1$;
- 24: $t_{il} = t_{il} - 1$, $\Delta\tau_i = \Delta\tau_i - 1$;
- 25: **if** $\Delta\tau_q == \tau_q$ **then** *Safe* deletes W_q ;
- 26: **if** $D(P'_i \parallel \bar{P}_{C_i}) \leq \theta_i$ or $\Delta\tau_i = 0$ **then break**;
- 27: compute $\text{case}_i^- = \{c_{i1}^-, c_{i2}^-, \dots, c_{iN}^-\}$;

ALGORITHM 2: Allocation modification.

Input: subregions $\{L_1, L_2, \dots, L_N\}$, participants $\{W_1, W_2, \dots, W_H\}$, workloads $\{J_1, J_2, \dots, J_N\}$, each community profile $\{\bar{P}_{C_1}, \bar{P}_{C_2}, \dots, \bar{P}_{C_N}\}$, each actual profile $\{p_{i1}, p_{i2}, \dots, p_{iN}\}$

Output: the allocation $\{t_{i1}, t_{i2}, \dots, t_{iN}\}$ for each W_i

- 1: update each $A_i = \{a_{i1}, a_{i2}, \dots, a_{iN}\}$ by P_i ;
- 2: **for** each W_i **do**
- 3: **for** each W_j that $j \neq i$ **do**
- 4: **if** $\tau_j - \Delta\tau_j > \Delta\tau_i$ and A_j covers A_i **then**
- 5: $\text{tmp} = \{t_{i1} + t_{j1}, t_{i2} + t_{j2}, \dots, t_{iN} + t_{jN}\}$;
- 6: compute P'_{tmp} by tmp ;
- 7: **if** $D(P'_{\text{tmp}} \parallel \bar{P}_{C_i}) \leq \theta_j$ **then**
- 8: $t_j = \text{tmp}$, $\Delta\tau_j = \Delta\tau_j + \Delta\tau_i$;
- 9: each $t_{ik} = 0$, $k \in \{1, 2, \dots, N\}$, $\Delta\tau_j = 0$;
- 10: **for** each W_i that $\Delta\tau_i > 0$ **do**
- 11: **if** $D(P'_i \parallel \bar{P}_{C_i}) > \theta_i$ **then fails**;

ALGORITHM 3: Allocation mergence.

$\sum_{i=1}^N J_i$) time. It finally costs $O(H \cdot \sum_{i=1}^N J_i)$ time in total, since H is always much bigger than N in practice.

4.4. Allocation Modification Phase. We introduce the second phase of TAPP in this part. We call the participant whose privacy leakage is bigger than the privacy threshold as a dangerous participant. In this phase, the algorithm modifies the allocations among participants in order to reduce dangerous participants. Specifically, the algorithm transfers some workloads from dangerous participants to safe participants, in order to make all participants safe.

Combined with Algorithm 2, the algorithm first updates set A by the actual profiles. Then it checks all allocated participants and adds dangerous participants in set *Danger* (Line 2~3). The set *Safe* contains two kinds of participants: (i) the safe participants have some allocated tasks but their workloads are less than the time threshold; (ii) the participants have no allocated task. The algorithm sorts the participants in *Danger* regarding to their workloads in descending order. It sorts each participant W_j in *Safe* regarding $\tau_j - \Delta\tau_j$ in ascending order. This means the dangerous participants first choose safe participants with allocated tasks, when they search for safe participants to transfer their workloads. After these, the algorithm transfers some allocations from dangerous participants to safe participants.

For each dangerous participant, the algorithm first computes the set $case_i^- = \{c_{i1}^-, c_{i2}^-, \dots, c_{iN}^-\}$. Each c_{ij}^- denotes the variation of the privacy leakage if W_i reduces a unit task associated with L_j . Moreover, we set $c_{ij}^- = \delta'$ if there is no candidate safe participants to allocate an unit task in L_j , where δ' is a very large number. Thus, $c_{ij}^- < 0$ means the privacy leakage will reduce if we reduce an unit task in L_j for W_i . The algorithm iteratively transfers a unit task associated with minimum c_{ij}^- to the candidate safe participant, until the privacy leakage of the dangerous participant is less than the threshold or its workload is zero (Lines 9~27). Specifically, a candidate safe participant W_j should satisfy $a_{jl} > 0$ and $\Delta\tau_j < \tau_j$, given the transferred task associated with L_l . Then it computes each c_{jl}^+ (Lines 12~14), which is similar to c_{jl}^- . The difference is that c_{jl}^+ is the variation of the privacy leakage if W_j adds an unit task associated with L_l . The algorithm chooses the candidate W_q according to the rest workload (Lines 18~22). Then the algorithm transfers an unit task in L_l from dangerous W_i to safe W_q .

The algorithm attempts to modify the allocations of all dangerous participants by the above transferring method. The best case is that there is no dangerous participant after these modifications. The time complexity of this phase is $O(H \cdot N \cdot \sum_{i=1}^N J_i)$ in total.

4.5. Allocation Mergence Phase. After the modification phase, we introduce the allocation mergence phase of TAPP in this part. The basic idea of this phase is that we can transfer all the allocations of a W_i to a W_j , if the time and privacy constraints of W_j are still satisfied. This procedure reduces the number

TABLE 2: Statistics for each city.

City	# Active Users
Cleveland, OH, USA	332
Tempe, AZ, USA	342
Calgary, AB, Canada	428

of allocated participants; thus it increases the utility of the CSP.

Combined with Algorithm 3, the algorithm first updates each set A_i by actual profiles. Then it iteratively checks each participant pair (W_i, W_j) , $i \neq j$ whether all their allocations can be merged and allocated to W_j (Lines 2~9). For each participant pair (W_i, W_j) , the algorithm first check the time constraint of W_j , where A_j covers A_i means $\forall a_{jk} = 0, a_{ik} = 1$ (Line 4). The set *tmp* is the mergence of all allocations (Line 5). Then the algorithm checks the privacy constraint of W_j , if we allocate the *tmp* to W_j . The algorithm allocates the mergence to W_j if the time and privacy constraints are satisfied and allocates no task to W_i (Line 7~9).

The algorithm checks all allocated participants at the end. If there still are some dangerous participants, the algorithm fails. The time complexity of this phase is $O(N \cdot H^2)$ in total.

5. Evaluation

We evaluate the performance of TAPP towards a real-world dataset from Yelp (<https://www.yelp.com/dataset/challenge>). Yelp is a location-based service system where reviewers publish reviews and comments for nearby businesses. In our evaluation, we consider reviewers as participants, and reviews as tasks. A review associates with a business, and the business associates with a location.

Three cities are considered in our evaluation: Cleveland, OH, USA; Tempe, AZ, USA; Calgary, AB, Canada. The user activities in each city reflect different real-world situations. Thus, these three cities are representative for evaluations. Specifically, we focus on the active participants with more than 30 reviews. The numbers of active participants in each city are shown in Table 2. The area of each city is divided into 3 by 3 grids. Since each review has a corresponding grid, the participant's actual profile is the ratio of reviews located in each grid. We consider the participants of a city belong to a community. The community profile for each city is the average value evaluated from all the participants' actual profiles. The cost for finishing a unit task is set to 1.

This evaluation focuses on two metrics: the number of dangerous participants, and the utility of the CSP. A dangerous participant is the one who has a privacy leakage more than its privacy threshold at the end of allocation. The number of dangerous participant is bigger than zero means the allocation is unfeasible. However, this metric helps us analyze the effectiveness of the allocation algorithm. Thus,

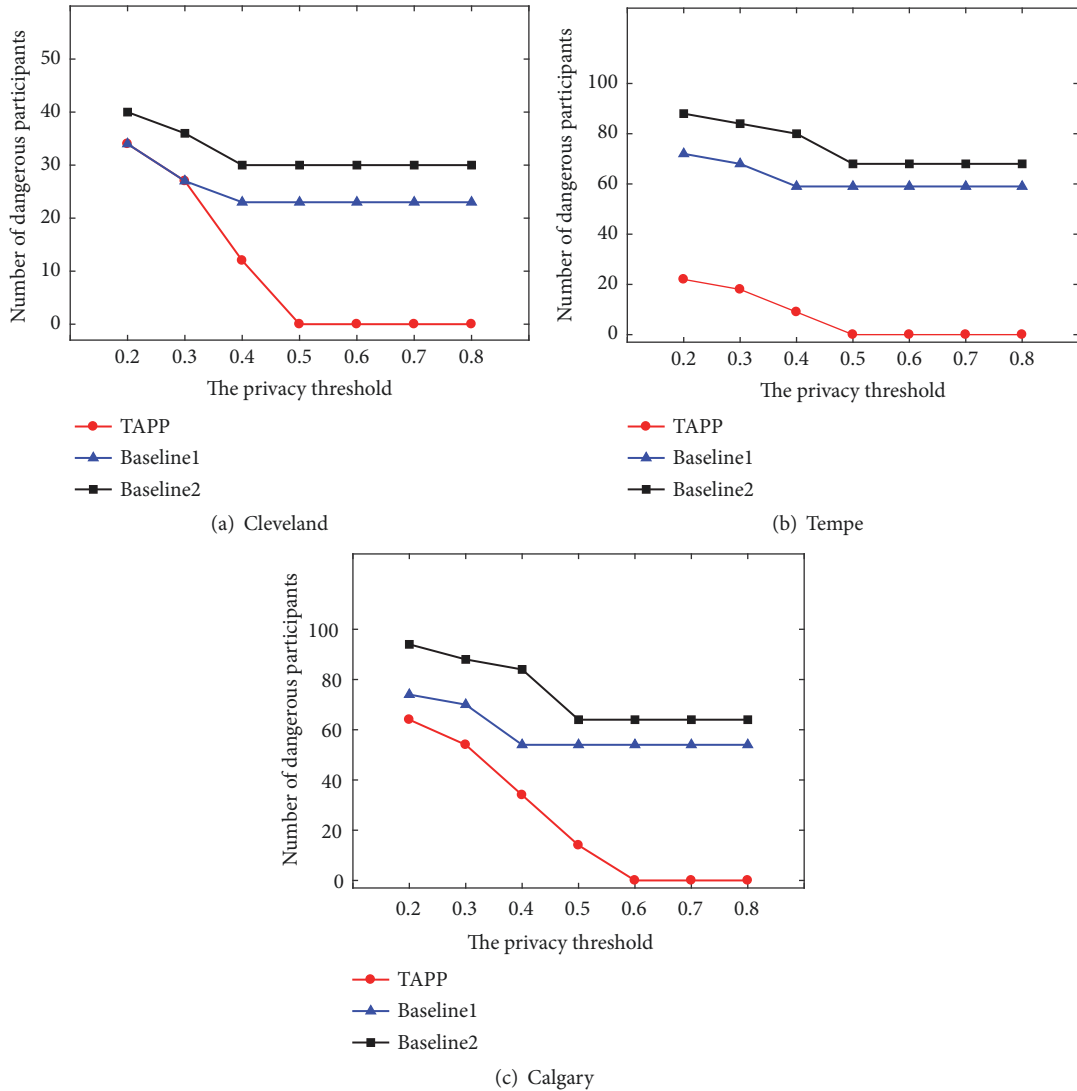


FIGURE 2: The number of dangerous participants among different allocating methods.

we change Line 22 to “else break” in Algorithm 2 for the evaluation.

The TAPP is compared with two baseline allocating methods. Baseline1 is the first phase of TAPP. Baseline2 is a greedy allocating method. It prefers to allocate the maximum J_i to participants who have the active subregion L_i and the maximum time threshold.

5.1. General Performance. We validate the effectiveness of TAPP in this part. The time threshold and the privacy threshold are set as the same value for each participant, respectively. Moreover, the privacy threshold ranges from 0.2 to 0.8.

Figure 2 shows the numbers of dangerous participants in each city. As we can see, when the privacy threshold is small, three algorithms suffer large numbers of dangerous participants. However, the number of dangerous participants in TAPP is averagely 76.23% less than Baseline1 and Baseline2,

when the privacy threshold ranges from 0.2 to 0.4, because the second and third phases of TAPP help to reduce the dangerous participants. The TAPP acquires feasible allocating solutions when the privacy threshold grows. Specifically, TAPP gets feasible solutions after the privacy threshold achieves 0.5, 0.5, and 0.6, respectively. Note that TAPP first acquires a feasible solution in Calgary with bigger privacy threshold than the other cities. It is because that the city with more population, the profiles are more heterogeneous. Then the algorithm performs relatively worse.

Figure 3 shows the utility of the CSP in each city. We treat an allocating solution as a feasible solution for comparison, even if it is unfeasible. Specifically, given H participants in total and H' participants who have allocated tasks, the utility of the CSP is $H - H'$ in this evaluation section. Because Baseline2 is based on the greedy strategy, the results of Baseline2 are close to optimal if there is no privacy constraint. The TAPP gets close to the results of Baseline2, when the privacy threshold grows bigger. By further analysis between

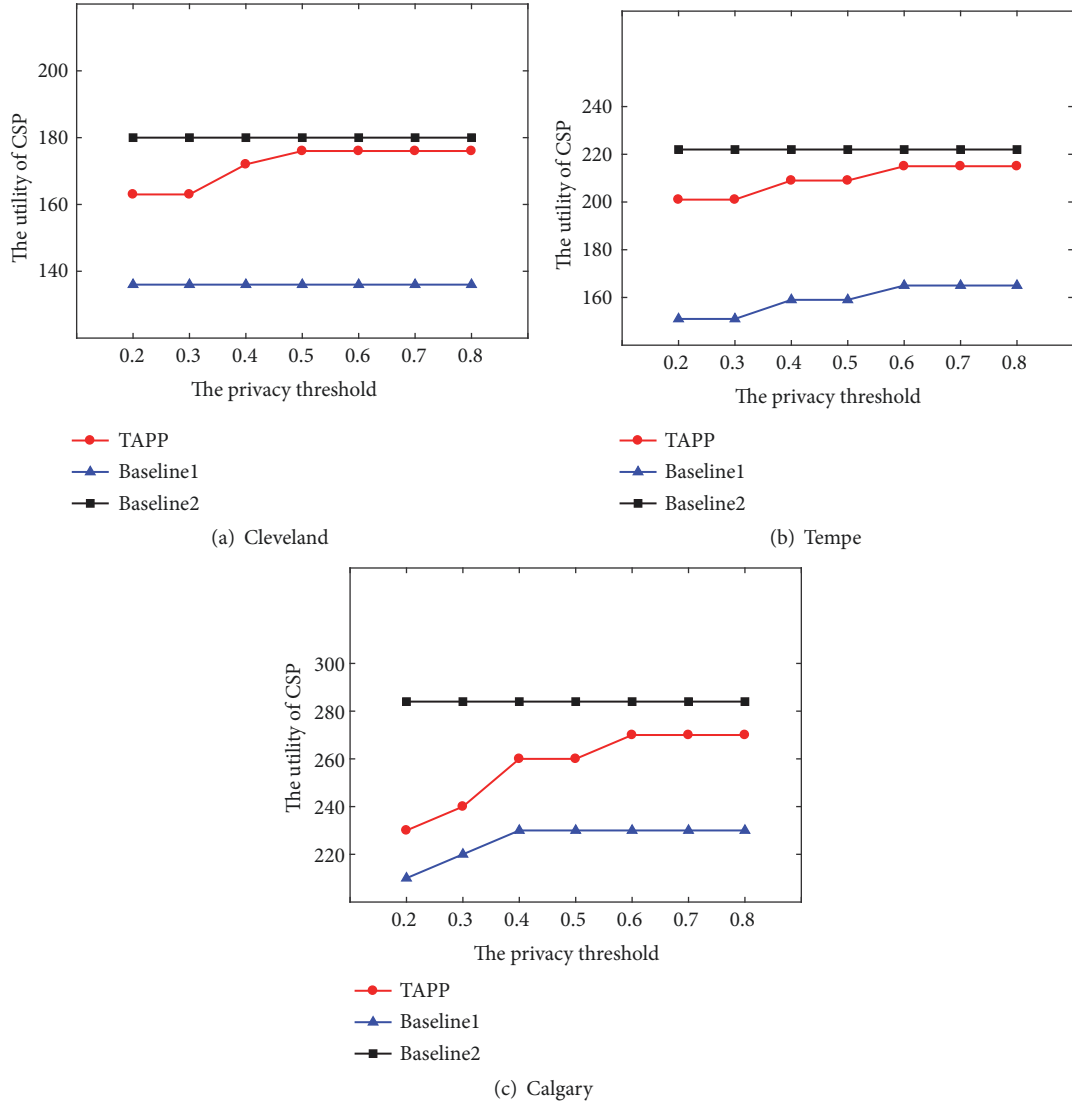


FIGURE 3: The utility of the CSP among different allocating methods.

TAPP and Baseline1, the second and third phases of TAPP averagely increase the utility 24.76% in three cities. Focusing on the results in Calgary, the heterogeneity of profiles affects the utility as well, since it affects the allocating solution.

5.2. Performance for Different Cases. In this part, we investigate the performance of TAPP for different types of task distributions. The results indicate the effectiveness of our algorithm under different task workloads. Specifically, we set the requested tasks $J^c = \{J_1^c, J_2^c, \dots, J_N^c\}$ following the distribution of the community profile, and the requested tasks $J^u = \{J_1^u, J_2^u, \dots, J_N^u\}$ following the uniform distribution. J^c and J^u satisfy $\sum_{i=1}^N J_i^c = \sum_{i=1}^N J_i^u$. The results under J^c and J^u are denoted as com-distribution and uni-distribution, respectively. The rest settings are as the same as in Section 5.1.

Figure 4 shows the numbers of dangerous participants under these two distributions. The privacy threshold under uni-distribution is bigger than com-distribution, when TAPP

first acquires a feasible solution. This is caused by the privacy constraint. The privacy constraint is based on the relative entropy between the observed profile and the community profile. Since J^c follows the distribution of the community profile, the algorithm is easier to acquire a solution which satisfies the privacy constraint. Meanwhile, TAPP acquires the first feasible solution under uni-distribution, when the privacy threshold is a little bigger than it under com-distribution. Comparing the results among three cities, the heterogeneity of profiles affects the dangerous numbers under different distributions as well.

Figure 5 shows the utilities of the CSP under these two distributions. The utilities under uni-distribution are less than com-distribution in all cases. This is because the task allocation is based on the privacy constraint, which is strongly related to the community profile. Since the com-distribution follows the same distribution with the community profile, the task allocation regarding the privacy constraint under

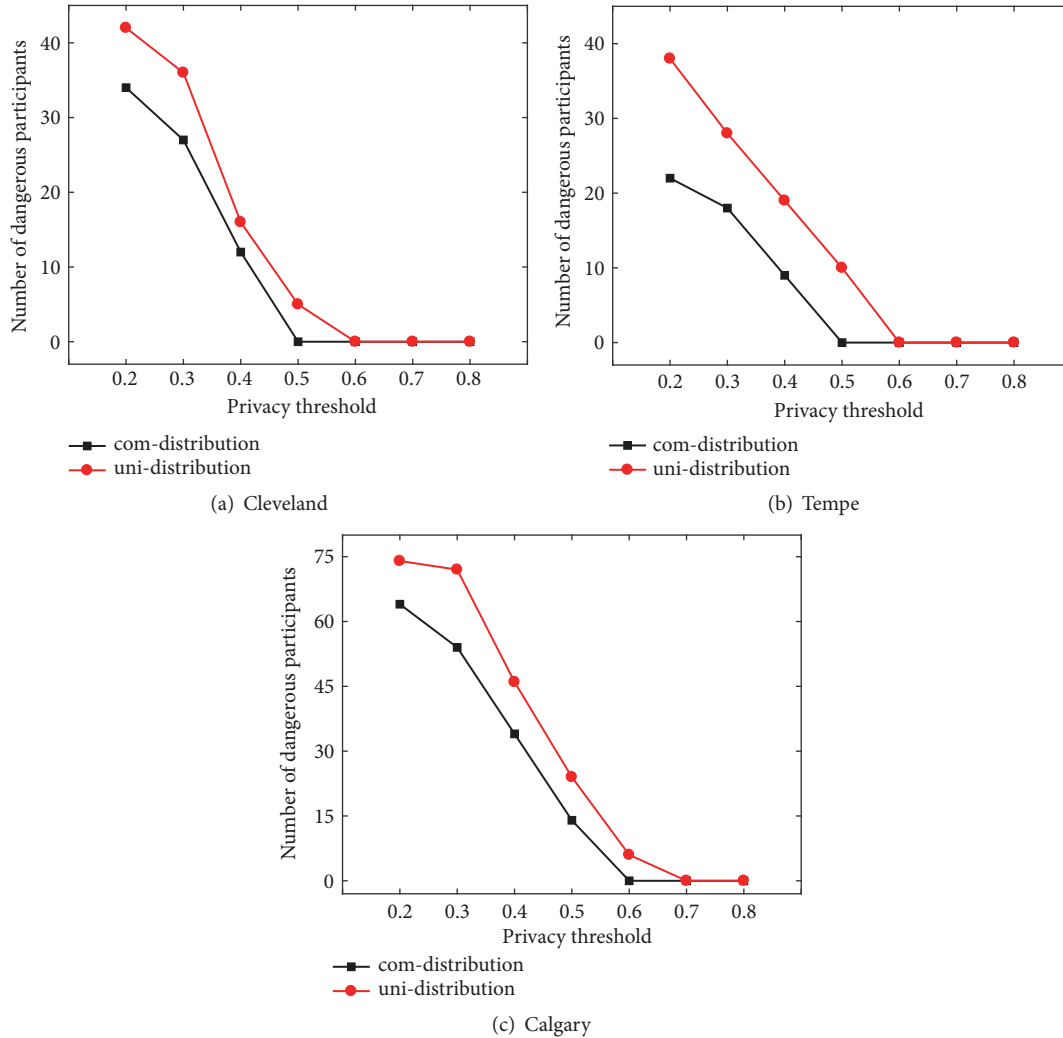


FIGURE 4: The number of dangerous participants under different task distributions.

the com-distribution is easier than the uni-distribution. The TAPP tries to make the solution satisfying the privacy constraint by allocating more participants, when given a more strict task distribution. The utility increases 6.53% and 6.15% under different distributions in Tempe and Calgary, respectively. Thus, the heterogeneity of profiles may not affect the utilities under different distributions.

6. Conclusion

Since the crowds care more about their privacy disclosure in recent years, the design of a task allocation algorithm should consider the privacy preservation. In this study, we investigate the algorithm of task allocation with basic considerations and the spatial privacy consideration. The problem formulation of task allocation is first presented. After that, we propose a task allocation algorithm on CSP's site with privacy preservation based on the formulation. It consists of three phases, allocating tasks without privacy preservation, modifying allocations with privacy consideration and merging the allocations. The

algorithm maximizes the benefit of the CSP, but meanwhile preserves the special privacy of participants. Evaluation results on utility and privacy aspects show the effectiveness of our proposed algorithm.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research work is supported by the key research plan for State Commission of Science Technology of China (2018YFC0807501, 2018YFC0807503), by the Foundation of Science & Technology Department of Sichuan province

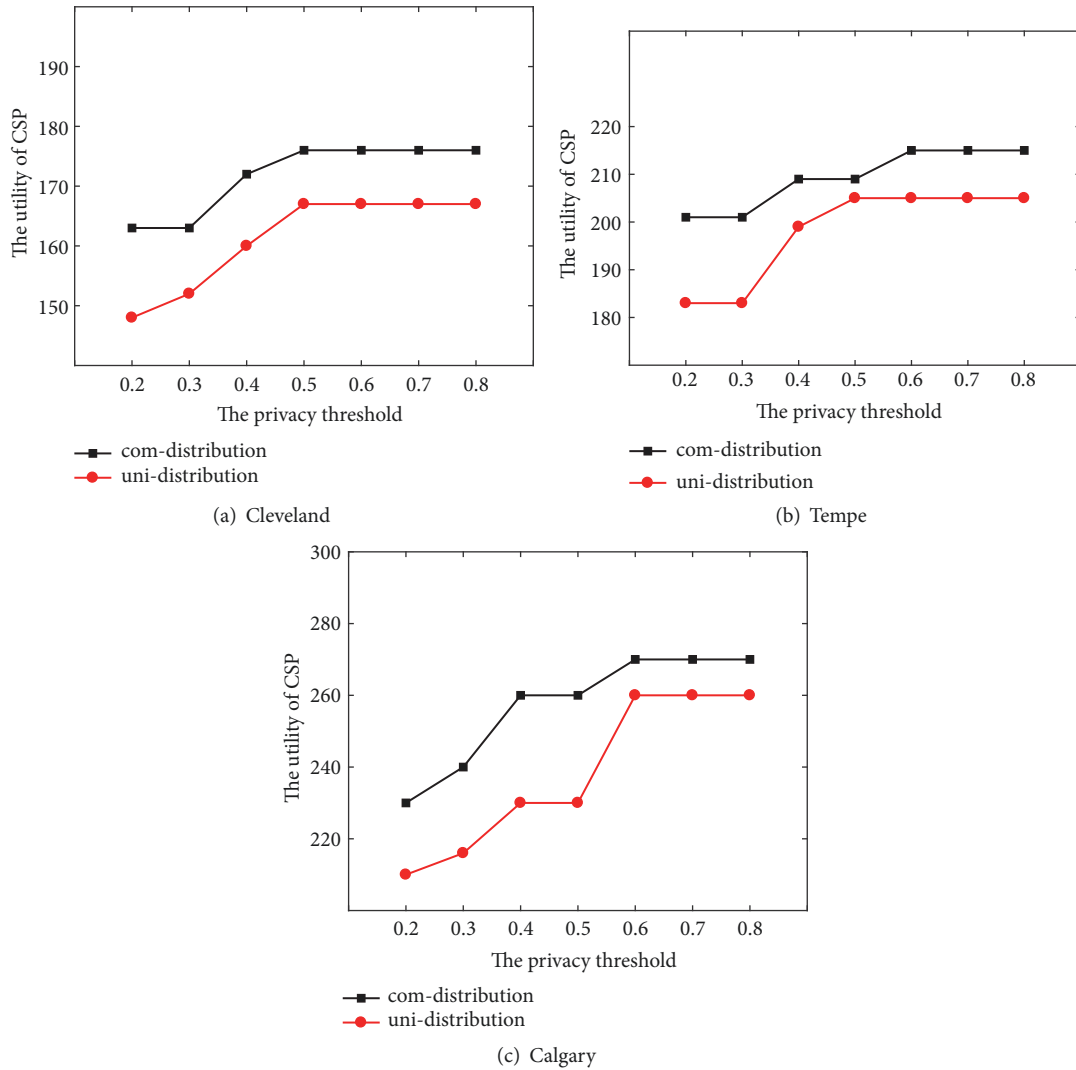


FIGURE 5: The utility of the CSP under different task distributions.

under Grants nos. 2017JY0027, 2017JY0007, 2018JY0067, 2017GFW0128, and 2016FZ0108, and by the Sichuan Provincial Economic and Information Commission (no. 2018DS010).

References

- [1] M. Seliem, K. Elgazzar, and K. Khalil, "Towards privacy preserving iot environments: a survey," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 1032761, 15 pages, 2018.
- [2] *Worldwide Internet of Things Forecast*, 2017, <https://www.idc.com/getdoc.jsp?containerId=IDC>.
- [3] R. Gartner, "Forecast: the internet of things, worldwide," in *The Internet of Things, Forecast*, 2017.
- [4] D. C. Bogatinoska, R. Malekian, J. Trengoska, and W. A. Nyako, "Advanced sensing and internet of things in smart cities," in *Proceedings of the 39th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2016*, pp. 632–637, Croatia, June 2016.
- [5] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An information framework for creating a smart city through internet of things," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 112–121, 2014.
- [6] M. Yun and B. Yuxin, "Research on the architecture and key technology of Internet of Things (IoT) applied on smart grid," in *Proceedings of the International Conference on Advances in Energy Engineering (ICAEE '10)*, pp. 69–72, June 2010.
- [7] N. Bui, A. P. Castellani, P. Casari, and M. Zorzi, "The internet of energy: a web-enabled smart grid system," *IEEE Network*, vol. 26, no. 4, pp. 39–45, 2012.
- [8] S. Sahabiswas, S. Saha, P. Mitra et al., "Drunken driving detection and prevention models using Internet of Things," in *Proceedings of the 7th IEEE Annual Information Technology, Electronics and Mobile Communication Conference, IEEE IEMCON 2016*, Canada, October 2016.
- [9] T. T. Thakur, A. Naik, S. Vatari, and M. Gogate, "Real time traffic management using Internet of Things," in *Proceedings of the 2016 International Conference on Communication and Signal Processing, ICCSP 2016*, pp. 1950–1953, India, April 2016.

- [10] J. L. Cai, M. Yan, and Y. Li, "Using crowdsourced data in location-based social networks to explore influence maximization," in *Proceedings of the IEEE INFOCOM 2016 - IEEE Conference on Computer Communications*, pp. 1–9, San Francisco, CA, USA, April 2016.
- [11] J. Li, Z. Cai, J. Wang, M. Han, and Y. Li, "Truthful incentive mechanisms for geographical position conflicting mobile crowdsensing systems," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 2, pp. 324–334, 2018.
- [12] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science & Engineering*, no. 99, pp. 1–1, 2018.
- [13] H. To, L. Fan, L. Tran, and C. Shahabi, "Real-time task assignment in hyperlocal spatial crowdsourcing under budget constraints," in *Proceedings of the 14th IEEE International Conference on Pervasive Computing and Communications, PerCom 2016*, Australia, March 2016.
- [14] H. Xiong, D. Zhang, L. Wang, and H. Chaouchi, "EMC3: energy-efficient data transfer in mobile crowdsensing under full coverage constraint," *IEEE Transactions on Mobile Computing*, vol. 14, no. 7, pp. 1355–1368, 2015.
- [15] H. Zhang, J. Liu, and N. Kato, "Threshold tuning-based wearable sensor fault detection for reliable medical monitoring using bayesian network model," *IEEE Systems Journal*, vol. 12, no. 2, pp. 1886–1896, 2018.
- [16] M. Zhang, P. Yang, C. Tian et al., "Quality-aware sensing coverage in budget-constrained mobile crowdsensing networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7698–7707, 2016.
- [17] T. G. Rodrigues, K. Suto, H. Nishiyama, and N. Kato, "Hybrid method for minimizing service delay in edge cloud computing through VM migration and transmission power control," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 66, no. 5, pp. 810–819, 2017.
- [18] X. Zheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Follow but no track: privacy preserved profile publishing in cyber-physical social systems," *IEEE Internet of Things Journal*, 2017.
- [19] Y. Huo, C. Yong, and Y. Lu, "Re-adp: Real-time data aggregation with adaptive ω -event differential privacy for fog computing," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 6285719, 13 pages, 2018.
- [20] Y. Huo, Y. Tian, L. Ma, X. Cheng, and T. Jing, "Jamming strategies for physical layer security," *IEEE Wireless Communications Magazine*, vol. 25, no. 1, pp. 148–153, 2018.
- [21] X. Zheng, Z. Cai, and Y. Li, "Data linkage in smart internet of things systems: a consideration from a privacy perspective," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 55–61, 2018.
- [22] S. Hayashida, D. Amagata, T. Hara, and X. Xie, "Dummy generation based on user-movement estimation for location privacy protection," *IEEE Access*, vol. 6, pp. 22958–22969, 2018.
- [23] L. Wang, D. Zhang, A. Pathak et al., "CCS-TA: Quality-guaranteed online task allocation in compressive crowdsensing," in *Proceedings of the 3rd ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp 2015*, pp. 683–694, Japan, September 2015.
- [24] L. Wang, D. Zhang, Y. Wang, C. Chen, X. Han, and A. M'hamed, "Sparse mobile crowdsensing: challenges and opportunities," *IEEE Communications Magazine*, vol. 54, no. 7, pp. 161–167, 2016.
- [25] Y. Zhu, Z. Li, H. Zhu, M. Li, and Q. Zhang, "A compressive sensing approach to urban traffic estimation with probe vehicles," *IEEE Transactions on Mobile Computing*, vol. 12, no. 11, pp. 2289–2302, 2013.
- [26] H. Xiong, D. Zhang, G. Chen, L. Wang, and V. Gauthier, "CrowdTasker: maximizing coverage quality in piggyback crowdsensing under budget constraint," in *Proceedings of the 13th IEEE International Conference on Pervasive Computing and Communications, PerCom 2015*, pp. 55–62, USA, March 2015.
- [27] D. Zhang, H. Xiong, L. Wang, and G. Chen, "CrowdRecruiter: Selecting participants for piggyback crowdsensing under probabilistic coverage constraint," in *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 703–714, ACM, September 2014.
- [28] X. Sheng, J. Tang, and W. Zhang, "Energy-efficient collaborative sensing with mobile phones," in *Proceedings of the IEEE Conference on Computer Communications, INFOCOM 2012*, pp. 1916–1924, USA, March 2012.
- [29] B. Guo, Y. Liu, W. Wu, Z. Yu, and Q. Han, "ActiveCrowd: a framework for optimized multitask allocation in mobile crowdsensing systems," *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 3, pp. 392–403, 2017.
- [30] S. He, D.-H. Shin, J. Zhang, and J. Chen, "Toward optimal allocation of location dependent tasks in crowdsensing," in *Proceedings of the 33rd IEEE Conference on Computer Communications (INFOCOM '14)*, pp. 745–753, IEEE, Toronto, Canada, May 2014.
- [31] Y. Liu, B. Guo, Y. Wang, W. Wu, Z. Yu, and D. Zhang, "TaskMe: multi-task allocation in mobile crowd sensing," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp 2016*, pp. 403–414, Germany, September 2016.
- [32] G. Luo, K. Yan, X. Zheng, L. Tian, and Z. Cai, "Preserving adjustable path privacy for task acquisition in mobile crowdsensing systems," *Information Sciences*, 2018.
- [33] A. Khoshgozaran and C. Shahabi, "Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy," in *Proceedings of the International Symposium on Spatial and Temporal Databases*, pp. 239–257, Springer, 2007.
- [34] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias, "Preventing location-based identity inference in anonymous spatial queries," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 12, pp. 1719–1733, 2007.
- [35] S. Wang and X. Sean Wang, "In-device spatial cloaking for mobile user privacy assisted by the cloud," in *Proceedings of the 11th IEEE International Conference on Mobile Data Management, MDM 2010*, pp. 381–386, USA, May 2010.
- [36] H. Kido, Y. Yanagisawa, and T. Satoh, "An anonymous communication technique using dummies for location-based services," in *Proceedings of the 2nd International Conference on Pervasive Services (ICPS '05)*, pp. 88–97, IEEE Press, July 2005.
- [37] L. Kazemi and C. Shahabi, "A privacy-aware framework for participatory sensing," *ACM SIGKDD Explorations Newsletter*, vol. 13, no. 1, p. 43, 2011.
- [38] H. To, G. Ghinita, and C. Shahabi, "Framework for protecting worker location privacy in spatial crowdsourcing," *Proceedings of the VLDB Endowment*, vol. 7, no. 10, pp. 919–930, 2014.
- [39] Y. Wang, Z. Cai, X. Tong, Y. Gao, and G. Yin, "Truthful incentive mechanism with location privacy-preserving for mobile crowdsourcing systems," *Computer Networks*, vol. 135, pp. 32–43, 2018.
- [40] Z. Duan, W. Li, and Z. Cai, "Distributed auctions for task assignment and scheduling in mobile crowdsensing systems," in *Proceedings of the 37th IEEE International Conference on Distributed Computing Systems, ICDCS 2017*, pp. 635–644, USA, June 2017.

- [41] L. Wang, T. Wang, D. Yang, D. Zhang, X. Han, and X. Ma, "Location privacy-preserving task allocation for mobile crowdsensing with differential geo-obfuscation," in *Proceedings of the 26th International World Wide Web Conference, WWW 2017*, pp. 627–636, Australia, April 2017.
- [42] E. Wang, Y. Yang, J. Wu, W. Liu, and X. Wang, "An efficient prediction-based user recruitment for mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 17, no. 1, pp. 16–28, 2018.
- [43] Y. Yang, W. Liu, E. Wang, and H. Wang, "Beaconing control strategy based on game theory in mobile crowdsensing," *Future Generation Computer Systems*, vol. 86, pp. 222–233, 2018.

Research Article

Energy-Efficient Broadcast Scheduling Algorithm in Duty-Cycled Multihop Wireless Networks

Quan Chen,¹ Tao Wang,¹ Lianglun Cheng,¹ Yongchao Tao ^{2,3} and Hong Gao³

¹School of Computers, Guangdong University of Technology, Guangzhou, China

²Shenzhen Academy of Aerospace Technology, China

³School of Computer Science and Technology, Harbin Institute of Technology, China

Correspondence should be addressed to Yongchao Tao; taoyongchao@chinasat.com

Received 7 November 2018; Accepted 14 January 2019; Published 6 February 2019

Guest Editor: Wei Cheng

Copyright © 2019 Quan Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Broadcasting is a fundamental function for disseminating messages in multihop wireless networks. Minimum-Transmission Broadcasting (MTB) problem aims to find a broadcast schedule with minimum number of transmissions. Previous works on MTB in duty-cycled networks exploit a rigid assumption that nodes have only active time slot per working cycle. In this paper, we investigated the MTB problem in duty-cycled networks where nodes are allowed arbitrary active time slots per working cycle (MTBDCA problem). Firstly, it is proved to be NP-hard and $o(\ln \Delta)$ -inapproximable, where Δ is the maximum degree in the network. Secondly, an auxiliary graph is proposed to integrate nodes' active time slots into the network and a novel covering problem is proposed to exploit nodes' multiple active time slots for scheduling. Then, a $\ln(\Delta + 1)$ -approximation algorithm is proposed for MTBDCA and a $(\ln(\Delta + 1) + \Delta)$ -approximation algorithm is proposed for all-to-all MTBDCA. Finally, extensive experimental results demonstrate the efficiency of the proposed algorithm.

1. Introduction

Broadcasting is a fundamental function for many services in wireless sensor networks (WSNs) [1, 2], such as data collection, code updating, and topology discovering [3–6]. Many effective broadcasting algorithms have been investigated to improve the network performance, while the total number of transmissions is often taken as a measured metric. Obviously, the smaller the number of transmissions is, the more energy will be saved. Therefore, the Minimum-Transmission Broadcasting problem (MTB), which tries to minimize the number of transmissions, has got a lot of attentions from researchers. In traditional wireless sensor networks where each node always keeps awake, many broadcasting algorithms [7–15] have been proposed. Since the MTB problem has been proved to be NP-hard [7] and the aforementioned algorithms [7–15] are approximate ones, they achieve high energy efficiency during broadcasting.

However, it is well known that the duty-cycled scheme is commonly adopted in wireless sensor networks to conserve energy [16, 17]. In such working mode, each node switches

between the active state and sleep state cyclically. To save energy, all the functional modules (such as sensing the environment, sending, and receiving the messages) are turned off in sleep state. As this working mode is much different, the MTB problem in duty-cycled wireless sensor networks needs to be reinvestigated and the following two issues should be addressed: (1) A node may need several transmissions to inform all its neighbors since the neighbors may be active at different time slots. (2) How to construct the broadcast tree to exploit nodes' multiple active time slots for scheduling is also a big challenge. In duty-cycled wireless networks, the MTB problem is mainly studied by [18–21] currently. To reduce the number of transmissions, these algorithms try to organize the nodes with the same active time slot together, and they provided a solution for the aforementioned Issue 1. However, these methods exploit a strong assumption that all nodes can be active only once in a working cycle. When each node can be active arbitrary times per working cycle [16, 17, 22], these methods cannot solve the aforementioned Issue 2 efficiently and may result in a large redundancy.

Thus, to avoid the above shortcomings, the MTB problem in duty-cycled WSNs where nodes have arbitrary active time slots per working cycle (*MTBDCA* problem) is studied in this paper. To exploit nodes' whole active time slots, an auxiliary graph is designed and a novel kind of node covering problem, *i.e.*, minimum schedule node covering problem, is proposed. Finally, a $\ln(\Delta + 1)$ -approximation algorithm is proposed for *MTBDCA*. The contributions of this paper are mainly listed as follows.

(1) The *MTBDCA* problem is firstly formulated and proved to be NP-hard and $o(\ln \Delta)$ -inapproximable, where Δ is the maximum degree in the network. An auxiliary graph is designed to integrate nodes' active time slots into the network.

(2) In order to exploit nodes' whole active time slots for scheduling, a novel kind of node covering problem, *i.e.*, minimum schedule node covering problem, is proposed and proved to be NP-hard. An approximate algorithm is proposed to solve it.

(3) An approximation algorithm with ratio of $\ln(\Delta + 1)$ is proposed for *MTBDCA*, where Δ is the maximum degree in the network. An $(\ln(\Delta + 1) + \Delta)$ -approximation algorithm is also proposed for all-to-all *MTBDCA* problem.

(4) The extensive simulations verify that the proposed algorithm can achieve high performance in terms of energy efficiency.

Compared to the conference version [23], this paper not only investigates the all-to-all *MTBDCA* problem and provide a $(\ln(\Delta + 1) + \Delta)$ -approximation algorithm for it but also provides more details on the theoretical analysis, such as the proof of the NP-hardness of the Minimum Schedule Node Covering problem and the correctness of the broadcast tree construction algorithm, which are important for the completeness of the proposed methods. Additionally, the new experiments are also conducted to show some new findings of the proposed methods.

The rest of this paper is organized as follows. The related works are introduced in Section 2. Section 3 gives the network model and problem definition. The proposed approximate algorithm for *MTBDCA* is presented in Section 4. The simulation results and conclusion are given in Sections 5 and 6.

2. Related Works

The Minimum-Transmission Broadcasting (MTB) problem has drawn a lot of attentions from researchers. When each node always keeps awake, the MTB problem in WSNs are mainly studied by [7–15]. The NP-hardness of MTB problem was first proved in [7], where a tree-based structure is used to disseminate the messages. After that, the MCDS-based (Minimum Connected Dominate Set) scheme is used to further reduce the redundancy by minimizing the number of senders in the broadcast tree [8–10]. Lou *et al.* [11, 12] proposed a Dominant Pruning scheme and a quasi-local forward-node-set-based scheme to reduce the redundancy. The multipoint relay scheme is introduced for MTB in [13–15], which can determine a small set of forwarding nodes in a localized way. Since CDS is the core of the algorithms for

MTB, Wan *et al.* proposed an 8-approximation algorithm in [24]. With a two-phase constructing approach, they reduced the approximation ratio to 6.8 in [25]. Recently, the CDS construction problem in battery-free WSNs and the weakly CDS construction problem are studied in [26–28]. However, these methods are unsuitable for duty-cycled wireless networks.

In duty-cycled wireless networks, the MTB problem are studied in [18–21]. The MTB problem in duty-cycled networks was first proved to be NP-hard in [18]. In [18], a centralized $3(\ln \Delta + 1)$ -approximation algorithm is proposed by exploiting a set covering technique. After that, the authors in [19, 20] proposed a level-based scheduling framework and constructed the backbone according to nodes' level information to reduce the number of transmissions. Recently, the authors in [21] found that the number of transmissions can be further improved if a depth-first search is conducted on the forwarding nodes. Based on this, they proposed an efficient broadcasting algorithm outperforms the previous ones. However, these methods all assume that all nodes can active only once in a working cycle, which limits their application. When each node has multiple active time slot in a working cycle, the efficient algorithms have been proposed for multicasting [16, 29], flooding [22], data aggregation [17], and beaconing [30] in the duty-cycled networks, respectively. Thus, it is very necessary and meaningful to study the *MTBDCA* problem.

3. Problem Definition

Let $G = (V, E)$ denote a duty-cycled WSN, where $V = \{1, 2, \dots, n\}$ denotes all nodes and $E = \{(u, v) \mid 1 \leq u, v \leq n \ \& \ u \neq v\}$ denotes the neighborhood relationship among nodes. As discussed before, each node owns two states in such network, *i.e.*, the sleep state and the active state, and switches between these two states cyclically. Let \mathcal{W} denote a working cycle which includes $|\mathcal{W}|$ time slots with same length, *i.e.*, $\mathcal{W} = \{0, 1, 2, \dots, |\mathcal{W}| - 1\}$. And assume $\mathcal{W}(u)$ denote the *working plan* of node u , which is defined as the set of the active time slots of u , *i.e.*, $\mathcal{W}(u) = \{t_1, t_2, \dots, t_k\} \subseteq \{0, 1, \dots, |\mathcal{W}| - 1\}$. If node u wants to receive a packet, it can only receive it when it is active (*i.e.*, the time slot $t \in \mathcal{W}(u)$). When it wants to send a message, it can choose a time slot when the receiver is awake to switch to the active state. The duty cycle is calculated as $|\mathcal{W}(u)|/|\mathcal{W}|$. Table 1 lists the major symbols used in this paper.

As for MTB problem in duty-cycled WSNs, one not only needs to construct a broadcast tree which is rooted at the source node s , but also computes the *Transmitting Schedule* of each nonleaf node to informs its children while the number of transmissions is minimized. Before giving the formal definition of *Transmitting Schedule*, some notations used in this paper should be clarified. Assume $NB(u)$ denote the set of u 's one-hop neighbors in G . And let T denote the broadcast tree; $nl(T)$ and $ch(u)$ denote the set of nonleaf nodes and the set of u 's children in the broadcast tree, respectively. Then we can have,

Definition 1 (transmitting schedule). Given any node $u \in V$, let $sch(u) = [u, t, ch(u, t)]$ denote node u 's one transmitting

TABLE 1: Symbols and notations.

Notation	Description
$G = (V, E)$	a duty-cycled WSN
n	the number of nodes in the network
\mathcal{W}	a working cycle
$\mathcal{W}(u)$	node u 's working plan
$sch(u)$	node u 's transmitting schedule
$ch(u, t)$	the set of node u 's children which can be reached at time t
$ch(u)$	the set of all child nodes of node u
$\mathcal{S}(u)$	all the transmitting schedules of node u
T	the broadcast tree of G
$\mathcal{S} = (T, \mathcal{S}(T))$	the broadcast schedules of G
$\mathcal{G} = (V', E')$	the auxiliary graph of G
$a_{u,i}$	the i -th schedule node of v
$a_{u,i} \cdot p$	the schedule node $a_{u,i}$'s primary node
$a_{u,i} \cdot t$	the schedule node $a_{u,i}$'s transmitting time slot
$R(a_{u,i})$	the set of primary nodes which can be reached by $a_{u,i}$
A_S	the set of schedule nodes for the MSNC problem
A_P	the set of all primary nodes of whose schedule node in A_S
$L(u)$	the level of node u
T_u	the subtree rooted at node u
$NB(u)$	the set of node u 's neighbors
ψ	the average number of neighbors in G
Δ	the maximum node degree of G

schedule, in which for any node $v \in ch(u, t)$, we have $(u, v) \in E$ and $t\%|\mathcal{W}| \in \mathcal{W}(v)$.

Given a transmitting schedule $sch(u)$, it means that node u can deliver the message to all the nodes in $ch(u, t)$ through only one transmission at time slot t . To receive the message, each node $v \in ch(u, t)$ is able to wake up at time slot t ($t\%|\mathcal{W}| \in \mathcal{W}(v)$). Note that a node may have several transmitting schedules, then we can find that $ch(u) = \bigcup_{sch(u) \in \mathcal{S}(u)} sch(u).ch(u, t)$, where $\mathcal{S}(u)$ is the set of all transmitting schedules of u .

Thus, the MTB problem in duty-cycled WSNs where nodes have arbitrary active time slots per working cycle (MTBDCA) is to compute a broadcast tree T and calculate one or several transmitting schedules for nonleaf nodes in the broadcast tree, *i.e.*, $\mathcal{S}(T)$, with minimum number of transmissions. The broadcast tree T and the transmitting schedules of each nonleaf node $\mathcal{S}(T)$ are called a broadcast schedule. It can be formalized as follows.

Input:

- (1) A duty-cycled network $G = (V, E)$ and a source node s .
- (2) The working plans for all nodes, *i.e.*, $\{\mathcal{W}(v) \mid \forall v \in V\}$.

Output: The broadcast schedule $\mathcal{S}_{min} = \{T, \mathcal{S}(T)\}$, where $\mathcal{S}(T) = \{\mathcal{S}(u) \mid \forall u \in nl(T)\}$, where

- (1) T is rooted at the source node s and spanning all of the nodes in V .

- (2) For any $sch(u) \in \mathcal{S}(u)$ and $\mathcal{S}(u) \in \mathcal{S}(T)$, $sch(u)$ satisfies the conditions in Definition 1.
- (3) The number of total transmissions, *i.e.*, $\sum_{\mathcal{S}(u) \in \mathcal{S}(T)} |\mathcal{S}(u)|$, is minimized.

Theorem 2. *The MTBDCA problem is NP-hard and there exists no polynomial-time approximation algorithm with ratio of $(1 - o(1)) \ln \Delta$ for MTBDCA unless $NP \subseteq DTIME(n^{O(\log \log n)})$, where Δ is the maximum degree in the network.*

Proof. We prove the NP-hardness of MTBDCA by showing that the Minimum Set Cover (MSC) problem can be mapped to one of its special cases. It has been proved that the MSC problem [29] is NP-hard and there exists no polynomial-time algorithm with ratio of $(1 - o(1)) \ln N$ unless $NP \subseteq DTIME(n^{O(\log \log n)})$, where N is the size of the MSC problem. Consider a special case of MTBDCA, where there is only one source node s along with its neighbors in the network and all the neighbors are only connected to s . Thus, source node s can choose a set of active time slots to cover all its neighbors. In this case, the MTBDCA problem is equivalent to the MSC problem. The theorem is proved. \square

4. Approximation Algorithm for MTBDCA

In this section, we will introduce an algorithm with approximation ratio of $\ln(\Delta + 1)$ for MTBDCA, which includes the following steps: (1) The auxiliary graph is designed and

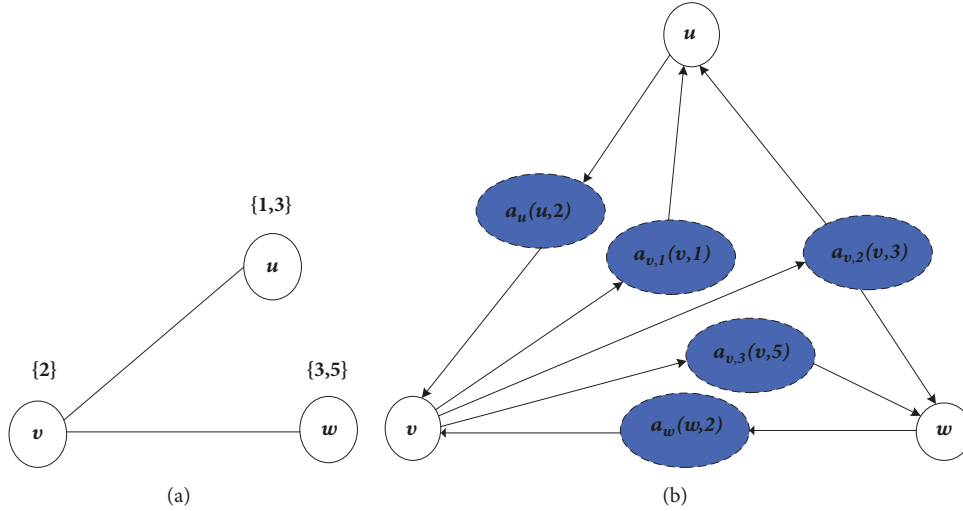


FIGURE 1: An example of the auxiliary graph. Figure 1 is reproduced from Chen et al. (2018) (under the Creative Commons Attribution License/public domain).

constructed to integrate nodes' active time slots into the network. (2) A *Minimum Scheduling Node Covering* problem is designed for multiple active time slot scheduling and an approximate algorithm is proposed for it. (3) A pseudo broadcast tree which contains nodes' schedule information is obtained on the constructed auxiliary graph. (4) A broadcast schedule for MTBDCA problem is computed according to the pseudo broadcast tree.

4.1. Constructing an Auxiliary Graph. In order to integrate nodes' working plan into the network and construct the broadcast tree, an *Auxiliary Graph* is first designed based on the duty-cycled network G . Different from the duty-cycled network, a new kind of node is introduced in the auxiliary graph, *i.e.*, *Schedule Node*. It is used to calculate the transmitting schedules for the nonleaf nodes.

In this paper, let a_u denote a schedule node of u ($u \in V$) (For simplicity, node u is called the primary node of a_u). Each schedule has two properties, *i.e.*, $(a_u.p, a_u.t)$, where $a_u.p$ denotes the schedule node's primary node and $a_u.t$ denotes its transmitting time slot. The auxiliary graph is defined as follows.

Definition 3 (auxiliary graph). Given a duty-cycled network $G = (V, E)$, the auxiliary graph $\mathcal{G} = (V', E')$ is a directed graph which contains nodes' active time slots. The set V' includes two kinds of nodes (primary node and schedule node) and E' is the set of all edges. They are constructed as follows:

- (i) Initially, $V' = V, E' = \emptyset$.
- (ii) For any node $u \in V$ and time $t' \in \bigcup_{v \in NB(u)} \mathcal{W}(v)$, we create a schedule node $a_{u,i}$ ($1 \leq i \leq |\bigcup_{v \in NB(u)} \mathcal{W}(v)|$). Its two properties are set as (u, t') (*i.e.*, $a_{u,i}.p = u$ and $a_{u,i}.t = t'$). Let $\Upsilon(u)$ denote the set of u 's schedule nodes, and then $V' = V' \cup_{u \in V} \Upsilon(u)$.

- (iii) For any node $u \in V$ and schedule node $a_{u,i} \in \Upsilon(u)$, create an edge from u to $a_{u,i}$. Let $E'_u = \{(u, a_{u,i}) \mid a_{u,i} \in \Upsilon(u)\}$, then E' can be updated as $E' = \bigcup_{u \in V} E'_u$.
- (iv) Let v be a one-hop neighbors of u in G . For each schedule node $a_{u,i} \in \Upsilon(u)$, add an directed edge $(a_{u,i}, v)$ in \mathcal{G} if $a_{u,i}.t \in \mathcal{W}(v)$. Let $R(a_{u,i})$ denote the set of such node v . Actually, it denotes the nodes can be reached by $a_{u,i}.p$ at time slot $a_{u,i}.t$. And we can have $E' = E' \cup_{u \in V} \{\bigcup_{a_{u,i} \in \Upsilon(u)} E'_{a_{u,i}}\}$, where $E'_{a_{u,i}} = \{(a_{u,i}, v) \mid v \in R(a_{u,i})\}$.

Note that, the auxiliary graph is directed, where the edges are either started from the primary node u to its schedule nodes in $\Upsilon(u)$, or started from a schedule node $a_{u,i}$ to the primary nodes it can reach at time $a_{u,i}.t$, *i.e.*, $R(a_{u,i})$. For example, there is a simple duty-cycled network in Figure 1(a) and the number in the braces denotes nodes' working plan. According to Definition 3, its auxiliary graph is constructed as in Figure 1(b). As for node v , there are two neighbors and the union of the active time slots of these neighbors are $\{1, 3, 5\}$. Then, three schedule nodes $a_{v,1}$, $a_{v,2}$ and $a_{v,3}$ are created, and their properties are set as $(v, 1)$, $(v, 3)$ and $(v, 5)$. As we can see time slot 1 is only included in the working plan of node u , then there is an edge from node v to $a_{v,1}$ and an edge from $a_{v,1}$ to node u . The pseudo code for constructing the auxiliary graph is shown in Algorithm 1.

The size of the auxiliary graph is analyzed in Theorem 5.

Lemma 4. Any schedule node in the auxiliary graph \mathcal{G} can connect to at most Δ primary nodes.

Proof. According to Definition 3, a schedule node $a_{u,i}$ is connected to u 's neighboring nodes which have the active time slot $a_{u,i}.t$ in its working plan. Obviously, there are at most Δ such nodes. Since the auxiliary graph is directed, thus, the schedule node $a_{u,i}$ can connect to at most Δ primary nodes. \square

```

Input: The duty-cycled network  $G = (V, E)$ , and the source node  $s$ ;
Output: The auxiliary graph  $\mathcal{G} = (V', E')$ ;
(1)  $V' \leftarrow V, E' \leftarrow \emptyset$ ;
(2) for  $\forall u \in V$  do
(3)    $S \leftarrow \bigcup_{v \in NB(u)} \mathcal{W}(v)$ ;
(4)    $i \leftarrow 1$ ;
(5)   for  $\forall t' \in S$  do
(6)      $V' = V' \cup \{a_{u,i}\}$ ;
(7)      $E' = E' \cup \{(u, a_{u,i})\}$ ;
(8)      $a_{u,i}.p \leftarrow u, a_{u,i}.t \leftarrow t'$ ;
(9)      $R(a_{u,i}) \leftarrow \{v \mid v \in NB(u) \& a_{u,i}.t \in \mathcal{W}(v)\}$ ;
(10)    for  $\forall v \in R(a_{u,i})$  do
(11)       $E' = E' \cup \{(a_{u,i}, v)\}$ ;
(12)    end for
(13)     $i \leftarrow i + 1$ ;
(14)  end for
(15) end for
(16) return  $\mathcal{G}$ ;

```

ALGORITHM 1: Algorithm for constructing the auxiliary graph.

Theorem 5. *The number of nodes and edges in the auxiliary graph \mathcal{G} are at most $(1 + |\mathcal{W}|) \times n$ and $(1 + \Delta) \times n \times |\mathcal{W}|$, respectively, where $|\mathcal{W}|$ denotes the length of a working period and $n = |V|$ is the number of nodes of the original graph.*

Proof. Firstly, according to Definition 3, a schedule node is created for each primary node u and each time slot $t \in \bigcup_{v \in NB(u)} \mathcal{W}(v)$. Since there are at most $|\mathcal{W}|$ time slots in $\bigcup_{v \in NB(u)} \mathcal{W}(v)$, then there are at most $|\mathcal{W}|$ schedule nodes for each primary node. Thus, the number of nodes in \mathcal{G} can be calculated as

$$\begin{aligned}
 |V'| &= |V| + |V_s| = n + \sum_{u \in V} |\mathcal{W}| \leq n + n \times |\mathcal{W}| \\
 &\leq (1 + |\mathcal{W}|) \times n
 \end{aligned} \tag{1}$$

where $|V_s|$ is the number of schedule nodes.

Secondly, for any primary node u , there is an edge between u and $a_{u,i}$ ($a_{u,i} \in Y(u)$) according to Definition 3, which is included in E'_u . And for any schedule node $a_{u,i}$, there are at most Δ edges from $a_{u,i}$ to the primary nodes in $R(a_{u,i})$. Then, we can have

$$\begin{aligned}
 |E'| &= \sum_{u \in V} |E'_u| + \left| \bigcup_{u \in V} \left\{ \bigcup_{a_{u,i} \in Y(u)} E'_{a_{u,i}} \right\} \right| \\
 &\leq \sum_{u \in V} |Y(u)| + \sum_{u \in V} (|Y(u)| \times \Delta) \\
 &\leq (1 + \Delta) \times n \times |\mathcal{W}|
 \end{aligned} \tag{2}$$

□

4.2. Minimum Schedule Node Covering Problem. In this subsection, we will introduce a new kind of problem, *i.e.*, *Minimum Scheduling Node Covering* problem. It tries to cover

all the nonsource primary nodes in the auxiliary graph with minimum schedule nodes. The new problem can be defined as follows.

Definition 6 (minimum schedule node covering problem (MSNC)). Given a source node s and an auxiliary graph \mathcal{G} , the MSNC problem is objected to compute a set of schedule nodes, *i.e.*, A_S , which satisfies the following conditions:

- (1) For each node $u \in V \& u \neq s$, there must exist a schedule node in A_S , *i.e.*, a_v , which satisfies that $u \in R(a_v)$ ($R(a_v)$ is the set of primary nodes which can be reached by a_v).
- (2) The size of A_S , *i.e.*, $|A_S|$, is minimized.

Theorem 7. *The MSNC problem is NP-hard and there exists no polynomial-time approximation algorithm with ratio of $(1 - o(1)) \ln \Delta$ for MSNC unless $NP \subseteq DTIME(n^{O(\log \log n)})$.*

Proof. We prove the NP-hardness of the MSNC problem by showing that the MSC problem can be mapped to one of its special cases. For example, Figure 2(a) gives an example of MSC, in which $U = \{d_1, d_2, \dots, d_n\}$ is the universe of all n elements and set $S = \{S_1, S_2, \dots, S_m\}$ denotes m subsets of U . An edge is created between d_i and S_k if d_i ($1 \leq i \leq n$) $\in S_k$ ($1 \leq k \leq m$). The MSC problem is then to calculate a set of minimum nodes from S to cover all the nodes in U .

To prove the NP-hardness of MSNC, a simple graph with a source node s and its neighboring nodes d_i ($1 \leq i \leq n$) is constructed. The working plan of each node is set as follows: (1) Initially, the working schedule of d_i ($1 \leq i \leq n$) is set null; (2) If d_i ($1 \leq i \leq n$) $\in S_k$ ($1 \leq k \leq m$), then $\mathcal{W}(d_i) = \mathcal{W}(d_i) \cup \{k\}$. Then, the corresponding auxiliary graph is shown in Figure 2(b), where S_k ($1 \leq k \leq m$) denote the schedule nodes. Then the MSNC problem is equivalent to the MSC problem, which is proved to NP-hard. The theorem is proved. □

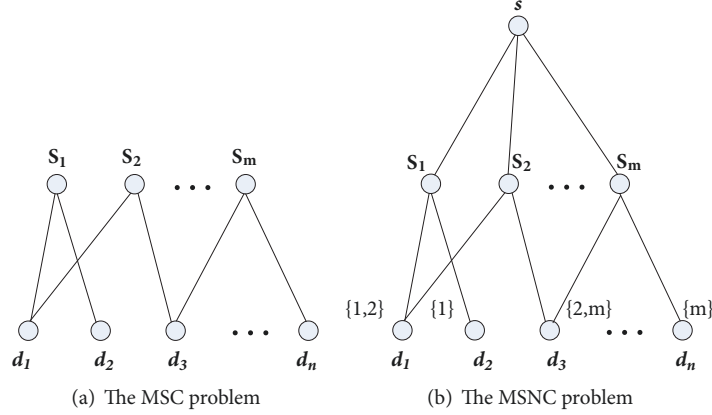


FIGURE 2: The example of MSNC problem. Figure 2 is reproduced from Chen et al. (2018) (under the Creative Commons Attribution License/public domain).

Input: The auxiliary graph \mathcal{G} , and the source node s ;
Output: The set of schedule nodes A_S ;
(1) $V_p \leftarrow V - \{s\}, A_S \leftarrow \emptyset$;
(2) $V_s \leftarrow$ all the schedule nodes in \mathcal{G} ;
(3) **while** $V_p \neq \emptyset$ **do**
(4) $a_v = \operatorname{argmax}_{a_u \in V_s} |R(a_u) \cap V_p|$;
(5) $A_S \leftarrow A_S \cup \{a_v\}$;
(6) $V_p \leftarrow V_p - (R(a_v) \cap V_p)$;
(7) $V_s \leftarrow V_s - \{a_v\}$;
(8) **end while**
(9) **return** A_S ;

ALGORITHM 2: Approximate algorithm for MSNC problem.

To solve the MSNC problem, we exploit the greedy set covering method [31] and choose the schedule node iteratively. Initially, A_S is set \emptyset and V_p is set $V - \{s\}$. In each loop, the schedule node which has maximum adjacent uncovered primary nodes, i.e., $a_v = \operatorname{argmax}_{a_u \in V_s} |R(a_u) \cap V_p|$, is found. Then, the schedule node a_v is added to A_S , i.e., $A_S = A_S \cup \{a_v\}$, and the covered primary nodes are removed from V_p . This process repeats until there are no uncovered primary nodes in V_p . The detailed procedure is shown in Algorithm 2. Since a schedule node can connect to at most Δ primary nodes, then Algorithm 2 can have an approximation ratio of $\ln(\Delta + 1)$ [31].

4.3. Generating the Pseudo Broadcast Tree. After obtaining the set of schedule nodes A_S , a *Pseudo Broadcast Tree* in the auxiliary graph is then constructed to connect all schedule nodes in A_S and these schedule nodes' primary nodes. The pseudo broadcast tree is mainly constructed by two steps: (1) The schedule nodes in A_S and their primary nodes are first organized into several subtrees; (2) These subtrees are then merged into one primary tree.

Assume $L(v)$ is the level of each node v (including the primary nodes and schedule nodes in the auxiliary graph). The above step (1) works as follows.

Initially, the level of source node s is set 0, i.e., $L(s) = 0$. Then, a breadth-first search is conducted from s level by level.

Second, let $A_p = \{a_v.p \mid a_v \in A_S\}$ denote the set of primary nodes whose schedule nodes is in A_S . If s is not included in A_p , add s into A_p , i.e., $A_p = A_p \cup \{s\}$.

Third, let u be the primary node with the smallest level in A_p and T_u denote the subtree rooted at u . Add u into T_u at the beginning, i.e., $T_u = T_u \cup \{u\}$. Then we can construct T_u by the following:

- (1) For any primary node m in $T_u \cap A_p$, let $A_S(m) = \{a_v \mid a_v \in A_S \& a_v.p = m\}$ be the set of schedule nodes which are contained in A_S and their primary node is m .
- (2) For any schedule node $a_m \in A_S(m)$, add it into the subtree T_u , i.e., $T_u = T_u \cup \{a_m\}$ and create an edge from m to schedule node a_m . Let n be a primary node in $R(a_m)$ and not included in any subtrees, add it into subtree T_u , i.e., $T_u = T_u \cup \{n\}$ and create an edge from schedule node a_v to n .
- (3) Delete the primary node m from A_p , i.e., $A_p = A_p - \{m\}$.

The above three steps repeat until there are no nodes in $T_u \cap A_p$. Then, we will choose another primary node with the smallest level in A_p to begin the above process. Note that several subtrees may be generated in this step. Finally, when A_p is empty, all the schedule nodes in A_S and the primary nodes in A_p are included in these subtrees.

Next, we will introduce the method to merge these subtrees.

Let TR denote the set of roots of these subtrees and the source node s is also included in TR . The objective of this process is to merge all the subtrees into the tree which is rooted at s , i.e., T_s . Note that T_s may include only one node at the beginning. Delete s from TR . The merging process mainly works as follows.

Case 1. If there is a node m in T_s and one of its schedule nodes, i.e., a_m , that can cover most root nodes in TR , then create an edge from m to a_m , and connect these root nodes in TR to

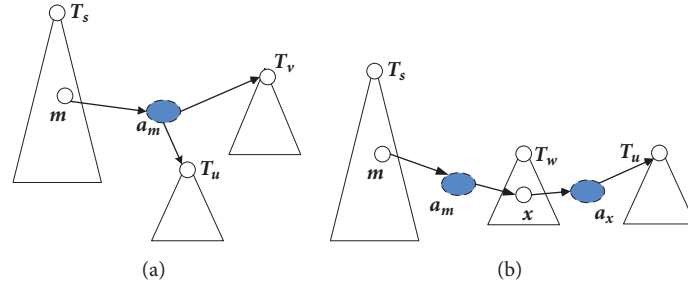


FIGURE 3: The example of the merging process. Figure 3 is reproduced from Chen et al. (2018) (under the Creative Commons Attribution License/public domain).

schedule node a_m , which is shown in Figure 3(a). Delete these root nodes from TR .

Case 2. Assume u is the root node with the smallest level in TR . There must exist a neighbor of u (i.e., $x \in NB(u)$) in the original graph and the primary node x is not in T_s . But it can be reached by a primary node in T_s , i.e., m , just as the example shown in Figure 3(b). Let the schedule node of m which can reach x be a_m , i.e., $x \in R(a_m)$. Create an edge from m to a_m and an edge from a_m to x . After that, let the schedule node of x that can reach u be a_x , then create an edge from x to a_x and an edge from a_x to u . Let the subtree contain x be T_w ; remove node x from T_w . Delete u from TR .

The above process repeats until all the subtrees are added to T_s either by Case 1 or Case 2. And then the pseudo broadcast tree is constructed completely. The correctness of this merging process can be verified by Theorem 10.

Lemma 8. *Let l denote the level of node x in the auxiliary graph. If x is a primary node, then l is even, and if x is a schedule node, then l is odd.*

Proof. First, we prove that if x is a primary node, then l is even. It can be easily verified that there exists a level of schedule nodes between two contiguous level of primary nodes when conducting the breadth-first search initiated from the source node s . Since the level of the source node s is 0, i.e., $L(s) = 0$ at the beginning, then all the schedule nodes of s must be level 1, and then the primary nodes reached by these schedule nodes have the level 2. Actually, let the hop-distance from a primary node u to node s in the original network be k , then we can find the level of the primary node u in the auxiliary graph must be $2k$, which is even.

Second, since the schedule nodes can only be reached by its corresponding primary nodes, whose level is even, then the level of the schedule nodes must be odd.

Therefore, in the generated breadth-first tree, the level of the primary nodes is even and the level of the schedule nodes is odd. \square

Lemma 9. *For any node v in the subtree rooted at node u , i.e., T_u , we have $L(v) \geq L(u) - 2$.*

Proof. First, we prove that, for any primary node v in the subtree rooted at node u , we have $L(v) \geq L(u) - 2$. According

to the construction of subtree, we can find that the root node u must be a primary node in A_p . Since one of its schedule nodes can reach a node whose level is less than u , i.e., w , then we can have $L(w) = L(u) - 2$. Assume there exist a primary node, i.e., w , that $L(w) < L(u) - 2$. Then there must exist a schedule node a_x in A_s that covers primary node w . According to Lemma 4, then we can have $L(w) - 1 \leq L(a_x) \leq L(w) + 1$, that is, $L(a_x) < L(u) - 3$. The level of the primary node of the schedule node a_x , i.e., x , can be calculated as $L(x) = L(a_x) + 1$, which is less than $L(u) - 2$. Since a_x is in A_s , then we can also have x is in A_p . However, when subtree T_u is constructed, primary node u has the smallest level, then we have $L(x) \geq L(u)$, which is a contradiction. Therefore, the level of all the primary nodes is no less than $L(u) - 2$.

Second, since the level of the schedule nodes are larger than level of its primary nodes, thus, the level of all schedule nodes is also no less than $L(u) - 2$.

Therefore, for any node v in the subtree rooted at node u , we have $L(v) \geq L(u) - 2$. The lemma is proved. \square

Theorem 10. *For any subtree rooted at node u , it can be merged into T_s .*

Proof. First, according to the merging process, if there exists a schedule node a_x , whose primary node is included in T_s and can reach node u , then the subtree rooted at node u will be merged into T_s , as in Case 1.

Second, we will prove that if there does not exist such a schedule node, then there must exist a primary node m as in the Case 2. Let the set of the primary nodes whose level is less than the one of u and can reach node u by some of their schedule nodes be $P_{up}(u)$. Then, there must exist a primary node in T_s , i.e., m , whose level is less than the one of $P_{up}(u)$, and can reach some node in $P_{up}(u)$, i.e., x , by some of its schedule nodes. According to Lemma 9, the level of the primary nodes in $P_{up}(u)$ is $L(u) - 2$. Obviously, we can have $L(m) = L(u) - 4$. Let the subtree contain node m be T_w . According to Lemma 9, then we can have $L(w) \leq L(m) + 2 \leq L(u) - 2$. Since the primary node with the smallest level is u , then the subtree T_w and node m must have been included in T_s , which means the primary node m is existed.

Therefore, the subtrees except T_s can be added to the primary subtree T_s either by Case 1 or Case 2. The theorem is proved. \square

Input: The duty-cycled network G and the source node s ;

Output: The broadcast schedule \mathcal{S} ;

- (1) Constructing the auxiliary graph \mathcal{G} as in Section 4.1 by integrating nodes' active time slot into the network;
- (2) Exploiting Algorithm 2 for the MSNC problem to obtain a minimum number of schedule nodes, *i.e.*, $A_{\mathcal{S}}$, to cover all the primary nodes in the auxiliary graph \mathcal{G} ;
- (3) Calculating a set of subtrees as in Section 4.3 with the scheduled nodes in $A_{\mathcal{S}}$ to connect all the primary nodes in \mathcal{G} ;
- (4) Merging all the subtrees into a pseudo broadcast tree which contains the scheduling information of all nodes, *i.e.*, T_s ;
- (5) Transforming the pseudo broadcast tree T_s to a broadcast schedule, *i.e.*, $\mathcal{S} = \{T, \mathcal{S}(T)\}$;
- (6) **return** The broadcast schedule \mathcal{S} ;

ALGORITHM 3: Approximate Minimum-Transmission Broadcasting.

4.4. Calculating the Broadcast Schedule. Actually, the pseudo broadcast tree constructed in the above subsection not only includes the information of the broadcast tree but also the scheduling information of nonleaf nodes. It can be transformed to the broadcast schedule (including a broadcast tree T and the transmitting schedules of nonleaf nodes) easily.

First, the broadcast tree T can be obtained by removing all the schedule nodes and create an edge from its father to all of its child nodes in the generated pseudo broadcast tree T_s .

Second, the transmitting schedules of nonleaf nodes in T , *i.e.*, $\mathcal{S}(T)$ can be obtained according to the schedule nodes in the pseudo broadcast tree. That is, for each schedule node a_u in T_s , a transmitting schedule for node a_u , *i.e.*, $[a_u.p, a_u.t + \chi * |\mathcal{W}|, ch(a_u)]$ is added in $\mathcal{S}(T)$, where $ch(a_u)$ denotes the set of children of schedule node a_u in T_s , and χ is used for collisions avoiding [32, 33] and data freshness guaranteed during broadcasting, which can be just set as the level of node u in the new broadcast tree T .

Now, the complete Approximate Minimum-Transmission Broadcasting (*i.e.*, AMTB) algorithm for MTBDCA problem is introduced and the detailed procedure is shown in Algorithm 3.

4.5. Performance Analysis of AMTB Algorithm. The correctness of AMTB is proved in Theorem 11, where the proof can be found in the conference version [23].

Theorem 11. *The broadcast schedule generated by AMTB is complete and correct.*

Theorem 12 gives the lower bound of MTBDCA problem. The approximation ratio of AMTB is proved in Theorem 13 and the proof of this theorem is available in the conference version [23].

Theorem 12. *The lower bound on the number of transmissions of any optimal broadcast schedule for MTBDCA is at least $|A_{\mathcal{S}}|/\ln(\Delta + 1)$, where $|A_{\mathcal{S}}|$ denote the size of the obtained set of schedule nodes by Algorithm 1.*

Proof. Assume OPT is the size for any optimal schedule for MTBDCA problem. To finish the broadcasting process, all nodes except the source need receive the messages from some intermediate nodes. It means all the primary nodes in the auxiliary graph except the primary node of the source need to be covered at least once, then we can get $OPT \geq |A_{\mathcal{S}}^{opt}|$, where

$|A_{\mathcal{S}}^{opt}|$ denote the size of any optimal schedule for MSNC problem. According to the above analysis, $|A_{\mathcal{S}}| \leq |A_{\mathcal{S}}^{opt}| \cdot \ln(\Delta + 1)$. Thence, we can get $OPT \geq |A_{\mathcal{S}}^{opt}| \geq |A_{\mathcal{S}}|/\ln(\Delta + 1)$. The theorem is proved. \square

Theorem 13. *The approximation ratio of AMTB is at most $\ln(\Delta + 1)$.*

Lemma 14. *The time complexity of Algorithm 2 is at most $O(n^2)$.*

Proof. Since there are at most $(1 + |\mathcal{W}|) * n$ schedule nodes in the auxiliary graph according to Lemma 8, then step (4) in Algorithm 2 takes at most $O((1 + |\mathcal{W}|) * n) = O(n)$ time ($|\mathcal{W}|$ is often a constant). And there are at most $n - 1$ primary nodes need to be covered, then the step (3) to (8) takes at most $O(n^2)$ time. Thus, the time complexity of Algorithm 2 is $O(n^2)$. \square

Theorem 15. *The time complexity of AMTB is $O(n^2)$.*

Proof. First, according to Lemma 8, it takes at most $O(1 + \Delta) \times n \times |\mathcal{W}|$ time to construct the auxiliary graph. Since Δ and $|\mathcal{W}|$ are often constant, then Algorithm 1 takes at most $O(n)$ time.

Second, according to Lemma 14, it needs at most $O(n^2)$ time to return a set of minimum schedule nodes for MSNC problem.

Third, to construct the pseudo broadcast tree, several subtrees are first generated and then these subtrees are merged into a single tree. Since there are at most $O(n)$ root nodes, and each subtree takes at most $O(n)$ time, then it takes at most $O(n^2)$ time to generate the subtrees. To merge these subtrees, one need to choose a subtree to merged into T_s at each step, which takes at most $O(n)$ time, and there are at most $O(n)$ subtrees. Thus, it takes at most $O(n^2)$ time to merge the subtrees.

Finally, the broadcast schedule can be obtained by searching the broadcast tree, which takes at most $O(n)$ time.

Therefore, the time complexity of AMTB is $O(n + n^2 + n^2 + n) = O(n^2)$. \square

4.6. Discussion of Algorithms for All-to-All Broadcasting Problem. In the above subsections, the Minimum-Transmission Broadcasting problem with only one source node is investigated. Despite the one-to-all broadcasting problem, the all-to-all broadcasting problem is also very important in wireless networks [34], which is aimed at distributing the messages

from all the nodes to all the other nodes. Similar as in [34], we assume the size of messages is unbounded. Then the above AMTB algorithm can still give an upper bound for the all-to-all MTBDCA problem. It mainly works as follows. First, all the messages are aggregated to a center node with the data aggregation technique in [17]. Second, the aggregated messages can be delivered to all the other nodes with the AMTB algorithm. The proposed algorithm can achieve an approximation ratio of $\ln(\Delta + 1) + \Delta$, which is shown in Theorem 16. As for the more efficient algorithms for all-to-all MTBDCA problem, it will be investigated in our future work.

Theorem 16. *The approximation ratio of the proposed algorithm for all-to-all MTBDCA problem is $\ln(\Delta + 1) + \Delta$.*

Proof. Let OPT_a and OPT_1 be the number of transmissions of the optimal solution for the all-to-all and one-to-all MTBDCA problem, respectively. In the above algorithm for all-to-all MTBDCA problem, the data aggregation process takes at most $n - 1$ transmissions, and the broadcasting process takes at most $\ln(\Delta + 1) \times OPT_1$ transmissions according to Theorem 13. Then the total number of transmissions is $n - 1 + \ln(\Delta + 1) \times OPT_1$. In addition, each node covers at most δ neighbors in the AMTB algorithm, then we can have $OPT_1 \times \Delta \leq n$. Since OPT_a must be larger than OPT_1 , then we can have $(n - 1 + (\Delta + 1) \times OPT_1) / OPT_a \leq (OPT_1 \times \Delta + \ln(\Delta + 1) \times OPT_1) / OPT_1 \leq \ln(\Delta + 1) + \Delta$. The theorem is proved. \square

5. Simulation Results

In this section, we will evaluate the performance of AMTB through extensive simulations. In the experiments, AMTB is mainly compared with the following algorithms.

First, the existing algorithms designed for MTB problem in duty-cycled networks, including the classical one, *i.e.*, CSCA [18] and the recently proposed one, *i.e.*, BRMS [21], are both evaluated to demonstrate that the AMTB algorithm can benefit from scheduling with nodes' all active time slots.

Second, two baseline algorithms, *i.e.*, CDS-based (Connected Dominate Set) algorithm and SPT-based (Shortest Path Tree) algorithm, are also implemented. In such two algorithms, a CDS-based and SPT-based broadcast tree is constructed firstly. Then, to reduce the number of transmissions, we calculate the transmitting schedules of all forwarding nodes in the broadcast tree optimally by enumerating all the possible results.

In the simulations, we mainly focus on the performance of the number of transmissions of each algorithm under various network topologies. Similar as in [30], the Networkx [35] tool was used to test more complex networks and generate different network topologies, where the number of sensor nodes is set from 100 to 400. The duty cycle of each node is set from 10% to 35% and the nodes' working plan is generated randomly to satisfy a wide range of configurations. In all the simulation results, each plotted point represents the average of 100 executions.

5.1. Impact of $|W|$. First, the performance of each algorithm under different length of working cycle (*i.e.*, $|W|$) and

number of nodes (*i.e.*, N) is evaluated. In this group of experiments, the duty cycle is set 20% and the average number of neighbors of each node is set 5.

Figure 4 shows the number of transmissions of each algorithm when we vary the length of working cycle under different network size. The number of nodes in Figures 4(a), 4(b), 4(c), and 4(d) is set as 100, 200, 300, and 400, respectively. In all scenarios, the number of transmissions of AMTB is much less than the one of other algorithms. Compared to the recently proposed BRMS algorithm, the number of transmissions of AMTB is decreased by 50% on average. This is mainly due to AMTB can exploit the whole active time slots among all neighbors for scheduling to reduce the number of transmissions. Additionally, although the CDS-based and SPT-based methods consider multiple active time slots during scheduling, they perform even worse than BRMS. This demonstrates that the performance can be still be worse even exploiting the optimal scheduling method if the broadcasting tree is not constructed well. Thus one must choose the forwarding nodes carefully when constructing the broadcasting tree.

Another finding is that when the length of working cycle is increased and the duty cycle remains unchanged, the number of transmissions of BRMS and CSCA method is even more, while the one of the algorithms exploiting multiple active slots for scheduling, *i.e.*, AMTB, SPT-based, and CDS-based method, is reduced. This is due to CSCA and BRMS methods only exploit nodes' first active time slot for scheduling. However, the number of neighbors who shares the same active time slot may be reduced when $|W|$ is increased. Actually, the number of active time slots in nodes' working plan is increased, and AMTB, SPT-based and CDS-based method can benefit from the increased active time slots to reduce the number of transmissions.

5.2. Impact of Duty Cycle. In this simulation, the performance of each algorithm under different duty cycle is compared. The length of nodes' working cycle is equal to 20.

Figure 5 shows the number of transmissions of each algorithm under different duty cycle. The number of nodes is set from 100 to 400, and the results are shown in Figures 5(a), 5(b), 5(c), and 5(d). As we can see, AMTB perform much better than all the other methods in all scenarios, which demonstrates its efficiency for MTBDCA problem. The SPT-based method still performs much worse than the others. This demonstrates that the SPT-based method may be not suitable for MTB problem in duty-cycled WSNs.

Another finding is that both the number of transmissions of the existing methods (BRMS and CSCA) and the one of AMTB is improved when we increase the duty cycle. This is because the probability of the first active time slot of neighbors being the same is increased when the duty cycle is increased (which means the size of node's working plan is increased). Note that, if we further increase the duty cycle, the number of transmissions of each method will still be decreased until the duty cycle is increased to 1. In this case, the MTBDCA problem is equal to the MTB problem in the traditional WSNs, and the number of transmissions of each method will be steadily.

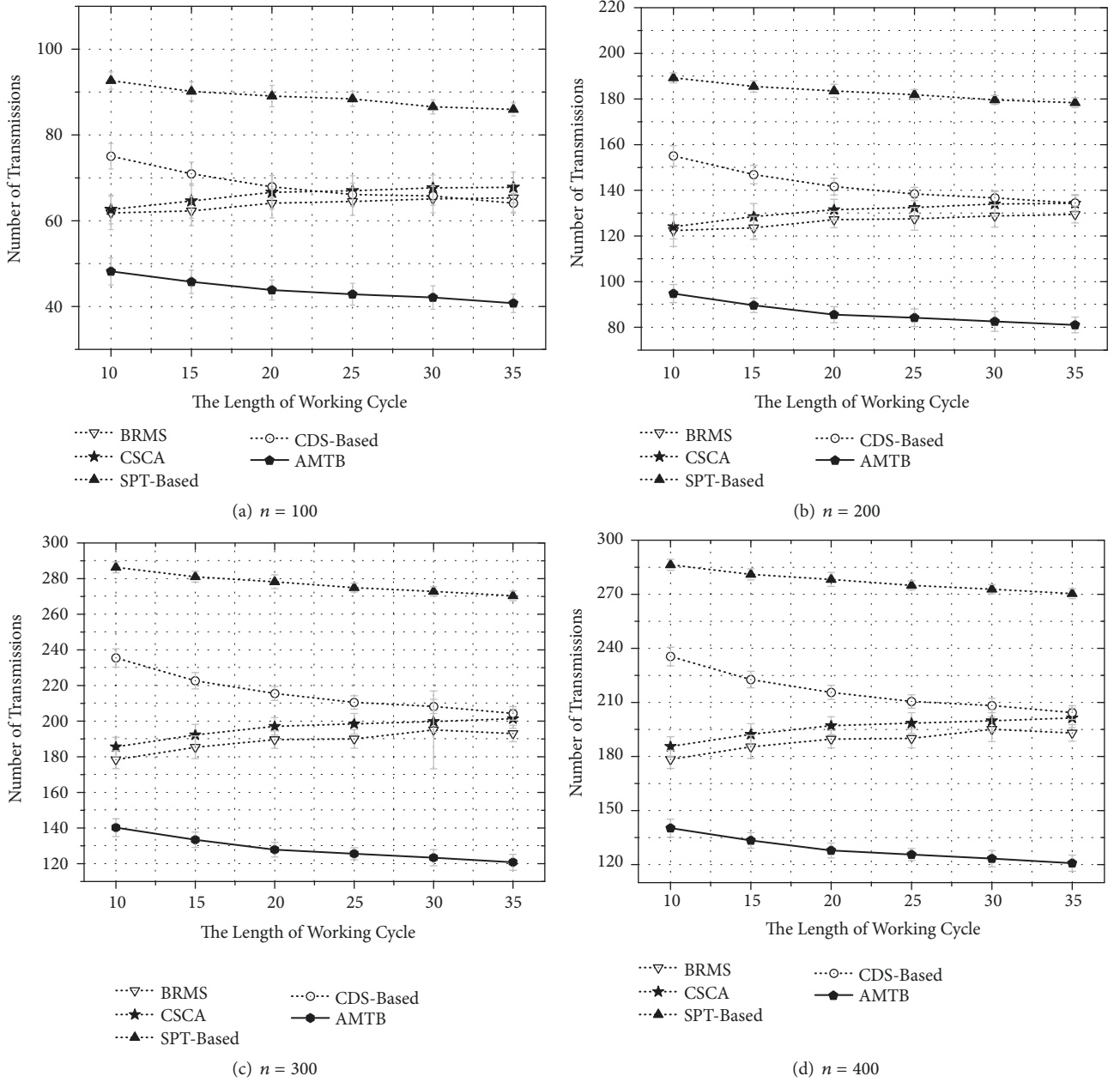


FIGURE 4: The number of transmissions when duty cycle is set 20%. Figure 4 is reproduced from Chen et al. (2018) (under the Creative Commons Attribution License/public domain).

5.3. *Impact of Neighbors.* Finally, we exploit Networkx to investigate the relationship between the number of transmissions of each method and the average number of neighbors, *i.e.*, ψ . To compare, the number of nodes is set 200 and the average number of neighbors is set 5 and 10 in two group of experiments.

Figure 6 presents the number of transmissions of the above algorithms under different length of working cycle, where the duty cycle is set 20%. Firstly, as we can see in Figures 6(a) and 6(b), the number of transmissions of each method is decreased notably if we increase the average number of neighbors in the network from 5 to 10. Although

the total number of nodes in the network is unchanged, the nodes sharing the same active time slot is increased when the number of neighbors is increased. As a result, the required number of transmissions is reduced. Compared to other methods, the proposed AMTB algorithm can reduce the number of transmissions by 40%, which demonstrates the efficiency of the proposed method. Similar to Figure 4, the number of transmissions of BRMS and CSCA are both increased when the length of working cycle is increased since they only exploit nodes' first active time slot for scheduling.

Figure 7 compares the number of transmissions of each method when we set $n = 200$ and $\psi = 10$ and change

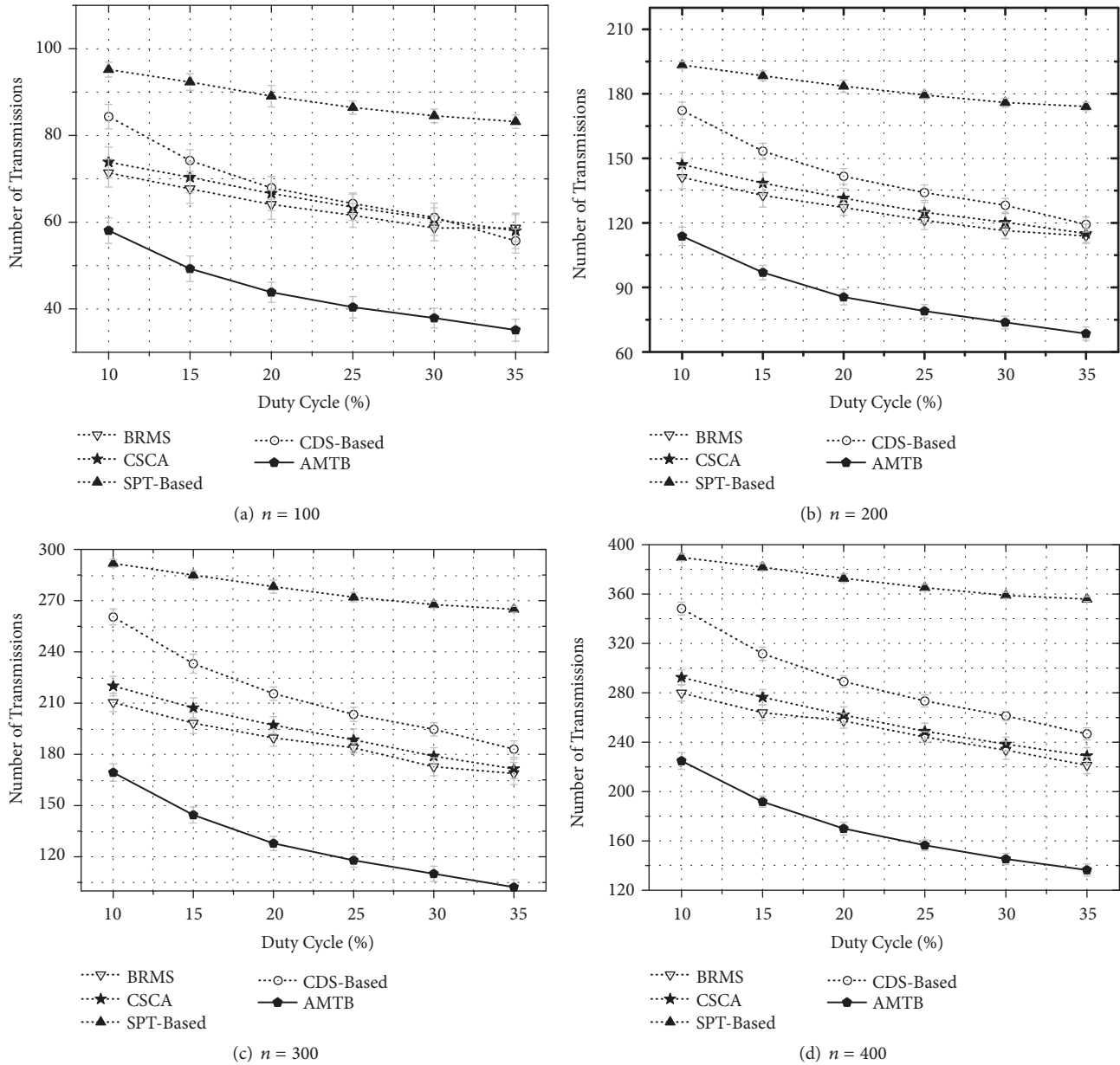


FIGURE 5: The number of transmissions when $|W|$ is set 20.

the duty cycle. In this case, the trend of each method is similar as in Figure 5, *i.e.*, the number of transmissions of each method is decreased when the duty cycle is increased. And the number of transmissions of AMTB is much less than the one of other methods. Comparing Figure 7(a) with Figure 7(b), the number of transmissions of each method is decreased notably (almost 50%) when the average number of neighbors is increased. This demonstrates that the number of transmissions is also highly related to the average degree in the network.

6. Conclusion and Future Work

In this paper, the MTBDCA problem in duty-cycled wireless networks is investigated. It is proved to be NP-hard

and $o(\ln \Delta)$ -inapproximable, where Δ denotes the maximum degree in the network. An auxiliary graph and the minimum schedule node covering problem is proposed to exploit nodes' all active time slots for scheduling. Based on this, a $\ln(\Delta + 1)$ -approximation algorithm is proposed for MTBDCA. The efficiency of the proposed algorithm is demonstrated by extensive simulations.

As for the future work, we will further investigate the all-to-all MTBDCA problem, including providing a more efficient algorithm for the all-to-all MTBDCA problem and studying the tight approximation ratio of the proposed methods. The simulations will also be conducted to demonstrate the efficiency of the proposed algorithms.

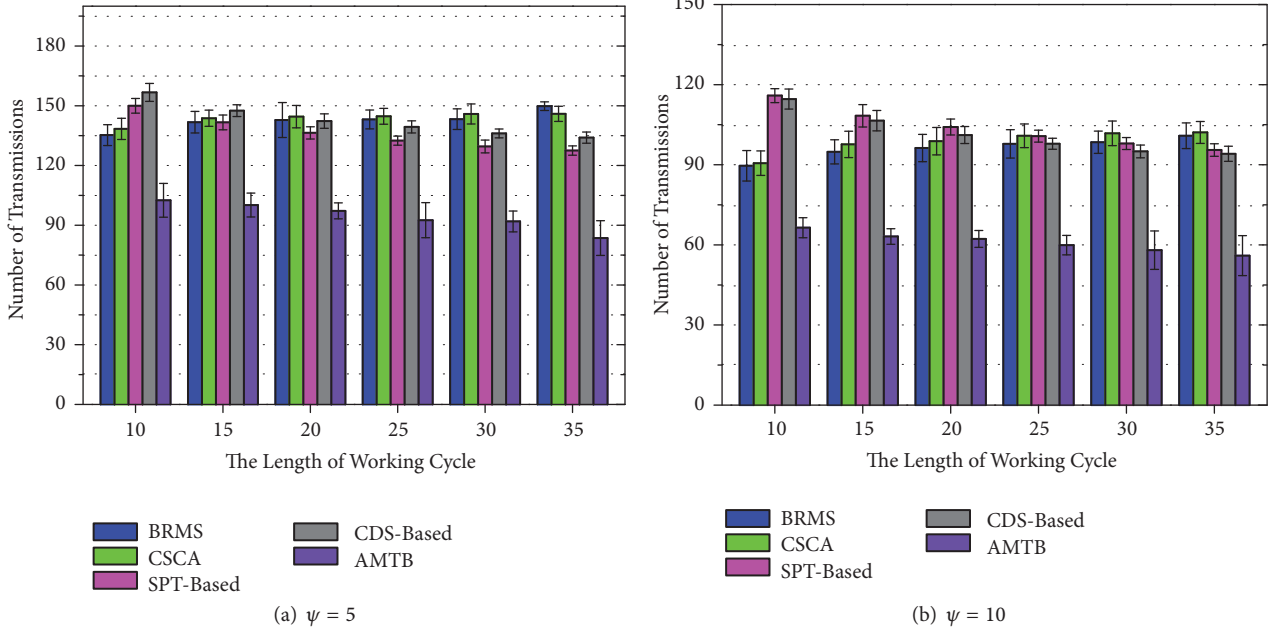


FIGURE 6: Performance analysis under different [W].

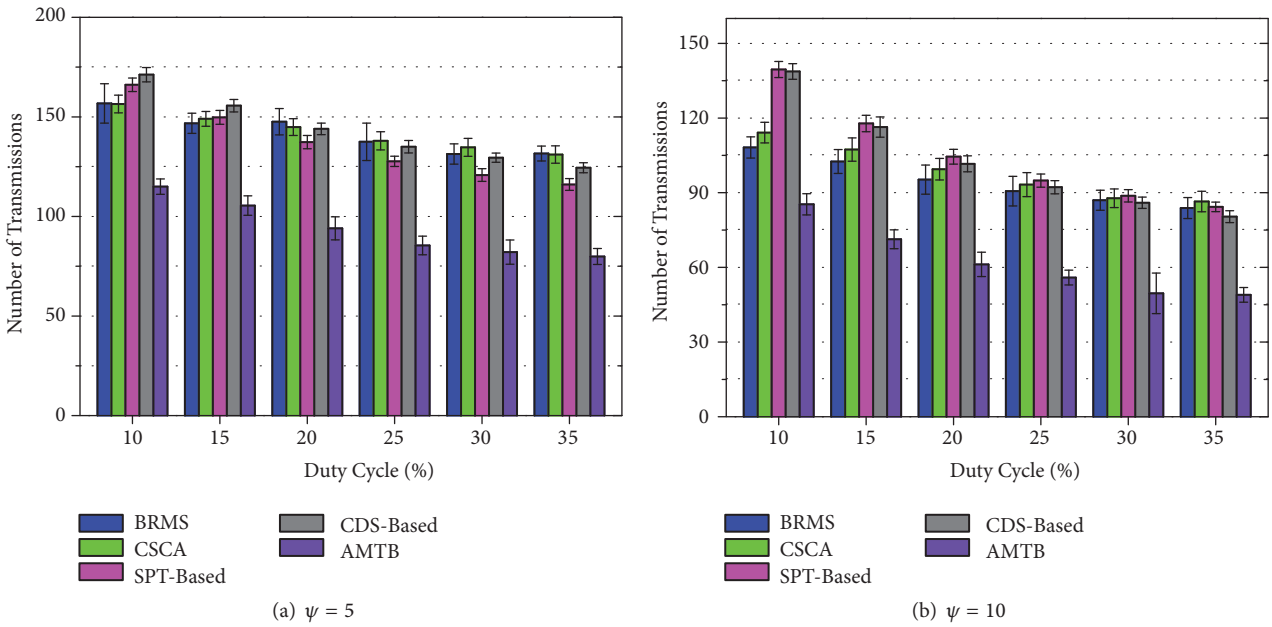


FIGURE 7: Performance analysis under different Duty Cycle.

Data Availability

The simulation data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

We thank T. Le and Z. Cai for useful contributions in the conference version [23]. This work is partly supported by the Opening Project of Guangdong Province Key Laboratory of Cyber-Physical System under Grant no. 2016B030301008, the National Natural Science Foundation of China (NSFC) under Grants nos. 61802071, 61502116, and 61502110, the China

Scholarship Council under Grant no. 201808440012, and the Natural Science Foundation of Guangdong under Grants nos. 2018A030310541, 2015B010104005, and 2017A010101017.

References

- [1] S. Cheng, Z. Cai, J. Li, and H. Gao, "Extracting kernel dataset from big sensory data in wireless sensor networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 4, pp. 813–827, 2017.
- [2] J. Yu, S. Wan, X. Cheng, and D. Yu, "Coverage contribution area based k-coverage for wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 9, pp. 8510–8523, 2017.
- [3] J. Li, S. Cheng, Z. Cai et al., "Approximate holistic aggregation in wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 13, no. 2, pp. 1–24, 2017.
- [4] Q. Chen, S. Cheng, H. Gao, J. Li, and Z. Cai, "Energy-efficient algorithm for multicasting in duty-cycled sensor networks," *Sensors*, vol. 15, no. 12, pp. 31224–31243, 2015.
- [5] S. Cheng, Z. Cai, J. Li et al., "Curve query processing in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 11, pp. 5198–5209, 2015.
- [6] Z. He, Z. Cai, S. Cheng, and X. Wang, "Approximate aggregation for tracking quantiles and range countings in wireless sensor networks," *Theoretical Computer Science*, vol. 607, no. 3, pp. 381–390, 2015.
- [7] I. Chlamtac and S. Kutten, "Tree-based broadcasting in multi-hop radio networks," *IEEE Transactions on Computers*, vol. C-36, no. 10, pp. 1209–1223, 1987.
- [8] W. Lou and J. Wu, "A cluster-based backbone infrastructure for broadcasting in MANETs," in *Proceedings of the International Parallel and Distributed Processing Symposium, IPDPS '03*, 2003.
- [9] B. Das and V. Bharghavan, "Routing in ad-hoc networks using minimum connected dominating sets," in *Proceedings of the IEEE International Conference on Communications (ICC '97)*, pp. 376–380, June 1997 (Basque).
- [10] J. Wu and H. Li, "On calculating connected dominating set for efficient routing in ad hoc wireless networks," in *Proceedings of the 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIALM '99)*, pp. 7–14, 1999.
- [11] W. Lou and J. Wu, "On reducing broadcast redundancy in ad hoc wireless networks," *IEEE Transactions on Mobile Computing*, vol. 1, no. 2, pp. 111–123, 2002.
- [12] J. Wu and W. Lou, "Forward-node-set-based broadcast in clustered mobile ad hoc networks," *Wireless Communications and Mobile Computing*, vol. 3, no. 2, pp. 155–173, 2003.
- [13] J. Wu, "An enhanced approach to determine a small forward node set based on multipoint relays," in *Proceedings of the Vehicular Technology Conference*, pp. 2774–2777, 2003.
- [14] C. Adjih, P. Jacquet, and L. Viennot, "Computing connected dominated sets with multipoint relays," *Ad Hoc and Sensor Wireless Networks*, vol. 1, no. 1, pp. 27–39, 2005.
- [15] J. Wu, W. Lou, and F. Dai, "Extended multipoint relays to determine connected dominating sets in MANETs," *IEEE Transactions on Computers*, vol. 55, no. 3, pp. 334–347, 2006.
- [16] Q. Chen, H. Gao, S. Cheng et al., "Centralized and distributed delay-bounded scheduling algorithms for multicast in duty-cycled wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 6, pp. 3573–3586, 2017.
- [17] Q. Chen, H. Gao, S. Cheng et al., "Distributed non-structure based data aggregation for duty-cycle wireless sensor networks," in *Proceedings of the IEEE INFOCOM*, pp. 145–153, 2017.
- [18] J. Hong, J. Cao, W. Li, S. Lu, and D. Chen, "Minimum-transmission broadcast in uncoordinated duty-cycled wireless Ad Hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 307–318, 2010.
- [19] T. L. Duc, D. T. Le, H. Choo et al., "On minimizing the broadcast redundancy in duty-cycled wireless sensor networks," in *Proceedings of the ACM International Conference on Ubiquitous Information Management and Communication*, 2013.
- [20] T. Le-Duc, D. Le, H. Choo et al., "Level-based approach for minimum-transmission broadcast in duty-cycled wireless sensor networks," *Pervasive and Mobile Computing*, vol. 27, pp. 116–132, 2016.
- [21] T. Le-Duc and V. Zalyubovskiy, "Towards broadcast redundancy minimization in duty-cycled wireless sensor networks," *International Journal of Communication Systems*, vol. 30, no. 6, pp. 1–12, 2017.
- [22] L. Cheng, J. Niu, Y. Gu, C. Luo, and T. He, "Achieving efficient reliable flooding in low-duty-cycle wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 24, no. 6, pp. 3676–3689, 2016.
- [23] Q. Chen, T. Le, L. Cheng et al., "Approximate minimum-transmission broadcasting in duty-cycled WSNs," in *Proceedings of the WASA*, pp. 40–52, 2018.
- [24] P.-J. Wan, K. M. Alzoubi, and O. Frieder, "Distributed construction of connected dominating set in wireless Ad Hoc networks," *Mobile Networks and Applications*, vol. 9, no. 2, pp. 141–149, 2004.
- [25] P.-J. Wan, L. Wang, and F. Yao, "Two-phased approximation algorithms for minimum CDS in wireless ad hoc networks," in *Proceedings of the 28th International Conference on Distributed Computing Systems, ICDCS '08*, pp. 337–344, July 2008.
- [26] T. Shi, S. Cheng, J. Li et al., "Constructing connected dominating sets in battery-free networks," in *Proceedings of the IEEE INFOCOM*, pp. 1–9, May 2017.
- [27] J. Yu, N. Wang, and G. Wang, "Constructing minimum extended weakly-connected dominating sets for clustering in ad hoc networks," *Journal of Parallel and Distributed Computing*, vol. 72, no. 1, pp. 35–47, 2012.
- [28] J. Yu, X. Ning, Y. Sun et al., "Constructing a self-stabilizing CDS with bounded diameter in wireless networks under SINR," in *Proceedings of the IEEE INFOCOM*, May 2017.
- [29] K. Han, Y. Liu, and J. Luo, "Duty-cycle-aware minimum-energy multicasting in wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 21, no. 3, pp. 910–923, 2013.
- [30] Q. Chen, H. Gao, Y. Li et al., "Edge-based beaconing schedule in duty-cycled multihop wireless networks," in *Proceedings of the INFOCOM*, 2017.
- [31] U. Feige, "A threshold of $\ln n$ for approximating set cover," *Journal of the ACM*, vol. 45, no. 4, pp. 634–652, 1998.
- [32] J. Yu, B. Huang, X. Cheng et al., "Shortest link scheduling algorithms in wireless networks under SINR," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 2643–2657, 2017.
- [33] X. Zhang, J. Yu, W. Li, X. Cheng, D. Yu, and F. Zhao, "Localized algorithms for yao graph-based spanner construction in wireless networks under SINR," *IEEE/ACM Transactions on Networking*, vol. 25, no. 4, pp. 2459–2472, 2017.

- [34] X. Jiao, W. Lou, J. Ma, J. Cao, X. Wang, and X. Zhou, "Minimum latency broadcast scheduling in duty-cycled multihop wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 1, pp. 110–117, 2012.
- [35] Networkx, 2012, <http://networkx.lanl.gov>.

Research Article

A Center-Based Secure and Stable Clustering Algorithm for VANETs on Highways

Xiaolu Cheng¹ and Baohua Huang² 

¹Virginia Commonwealth University, Richmond, Virginia 23220, USA

²Guangxi University, Nanning, Guangxi 530004, China

Correspondence should be addressed to Baohua Huang; bhhuang66@gxu.edu.cn

Received 15 October 2018; Revised 5 December 2018; Accepted 20 December 2018; Published 2 January 2019

Guest Editor: Zaobo He

Copyright © 2019 Xiaolu Cheng and Baohua Huang. 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.

Currently, communications in the vehicular ad hoc network (VANET) can be established via both Dedicated Short Range Communication (DSRC) and mobile cellular networks. To make use of existing Long Term Evolution (LTE) network in data transmissions, many methods are proposed to manage VANETs. Grouping the vehicles into clusters and organizing the network by clusters are one of the most universal and most efficacious ways. Since the high mobility of vehicles makes VANETs different from other mobile ad hoc networks (MANETs), the previous cluster-based methods for MANETs may have trouble for VANETs. In this paper, we introduce a center-based clustering algorithm to help self-organized VANETs forming stable clusters and decrease the status change frequency of vehicles on highways and two metrics. A novel Cluster Head (CH) selection algorithm is also proposed to reduce the impact of vehicle motion differences. We also introduce two metrics to improve the security of VANETs. A simulation is conducted to compare our mechanism to some other mechanisms. The results show that our mechanism obtains high stability and lower packet loss rate.

1. Introduction

As a key component of Intelligent Transportation Systems (ITS), vehicular ad hoc network (VANET) has attracted plenty of researchers from different fields, and massive research efforts have been made.

In VANETs, there are two types of communications [1]. VANETs enable both vehicle-to-vehicle (V2V) communications and vehicle-to-infrastructure (V2I) communications. In VANETs, vehicles and the infrastructures, such as Roadside Units (RSU) and application servers, exchange information for navigation, safe driving, entertainment, and so on.

Generally, communications in VANETs are roughly categorized into two classes according to the adopted radio interfaces. One class of approaches is based on Dedicated Short Range Communication (DSRC). The other class is based on existing cellular technology [2].

DSRC began to be used for V2V communication from the 90s. It has a shortage in medium range, which is about 300 meters. It is inadequate for large-scale deployment

[3] because its coverage radius is not large enough. With the rapid improvement of mobile cellular networks, some researchers supposed to utilize the existing mobile cellular infrastructures and technologies for communications in VANETs. Mobile cellular networks provide wider and larger coverage, while their delay is longer than DSRC for real-time information exchanges in local areas [4]. Therefore, both DSRC and mobile cellular networks cannot fully meet the needs of ITS. As a result, VANETs support communication not only via LTE but also via DSRC.

To make use of existing mobile cellular networks for data transmissions, many methods are proposed to manage VANETs. However, if VANETs are fully managed by infrastructures, low efficiency will be a big issue, while fully decentralized VANETs must create a lot of overhead. Therefore, VANETs usually combine some centralized parts and decentralized parts. To decrease the overhead via DSRC channels and the probability of LTE channel congestion, VANETs are centralized by cellular-based connections and

scheduling. Meanwhile, vehicles may also exchange messages with their neighbors via DSRC. Dividing vehicles into clusters is a common and reasonable approach for VANETs management. In a cluster-based framework, vehicles are signed into clusters. The range of a cluster is smaller or equal to the range of 802.11p, so that vehicles in the same cluster can exchange messages via DSRC. A single eNodeB manages many clusters around it. Within a cluster, at least one vehicle performs as a Cluster Head (CH) to collect information of all Cluster Members (CM) via DSRC and exchanges data with the eNodeB via TLE. This architecture decreases the management overhead while utilizing both DSRC and LTE.

Compared to other MANETs, nodes in VANETs have higher mobility and higher speed. Cluster reforming and CH changing must be much more frequent than other typical MANETs. To decrease the management overhead and increase communication quality, the clustering algorithm for VANETs should be able to form stable clusters. To achieve this goal, in this paper we propose a stable clustering algorithm for VANETs. We propose a novel approach to form and maintain stable clusters for VANETs on highways to avoid continual cluster reforming. A center-based clustering algorithm is used to locate the initial clusters' centers. In every cluster, a suitable CH is chosen by vehicles' position, speed, and maximal acceleration. A cluster maintenance algorithm is proposed to keep CMs in its CH's transmission range.

The rest of the paper is organized as follows. The Related Work briefly reviews the current literature on clustering algorithms in VANETs. The proposed scheme is detailed in the Proposed Scheme. The simulation parameters, simulation results, and analysis are shown in the Performance Evaluation. In the Conclusion, we state the conclusion.

2. Related Work

In the literature, clustering is the process to group vehicles in VANETS.

Ref. [5] proposes a method, named LTE4V2X, to organize vehicular networks. In the centralized vehicular networks, eNodeB manages vehicles in its coverage and divides them into clusters. LTE4V2X protocol defines how the self-organized network works. In LTE4V2X, eNodeB creates clusters which contain the largest number of nodes circulating in the same direction.

Ref. [6] extends LTE4V2X to increase information dissemination efficiency. It selects CH by the distance from vehicles to eNodeB. Although, comparing to the original approach, the complexity is lower and the LTE channel quality is higher, the power consumption of message exchanging is not optimized. Nevertheless, [6] states that the system can calculate the transmit power of DSRC channels by the distance between vehicles so that the transmit power could be dynamically adjusted.

Road condition affects the speed and direction of vehicles. For example, vehicle's speed is lower on the bumpy road than a smooth road. Vehicle mobility is determined by human behavior. Take a street connected megapolis and a village as an example. In the morning, most vehicles move from the village (home) to the megapolis (office). In the evening, most

vehicles run following the reverse path. Ref. [7] quantifies temporal locality similarity to measure the relation of two vehicles' mobility. Then, they utilize the relation of vehicles' movements to form stable clusters. The locality can also be used for reducing energy consumption [8].

Ref. [9] proposes a clustering approach to minimize the total power consumed by DSRC communications. They use a weighted distance matrix to indicate power consumed between each pair of vehicles. In this way, the CH selection problem is formulated as a variant of the p -median problem in graph theory [10]. In this approach, the number of clusters p is determined first based on LTE coverage radius and DSRC coverage radius. The p cluster zones are determined by vehicle number and 802.11p coverage radius. p Cluster Heads that are closet to the eNodeB are selected. Then, the system dynamically selects new CH to minimize the transmission power between CMs and CH based on weighted distance and the p -median issue in graph theory. Although this approach minimizes the power consumption within a single cluster, the power consumption of V2I communications has not been considered. The method to decide the zones is vague and complicated. Moreover, this approach is not suitable for the scenario that CMs not only send their information to CH but also communicate among themselves.

Ref. [11] proposes a high-integrity file transfer scheme for VANETs on highways named Cluster-based File Transfer (CFT) scheme. In this scheme, CMs help their CH to download file fragments and then transmit fragments to the CH which requests the file. Since the very high speed of vehicles on highways, CFT is a good approach to help the vehicles download files which they have not enough connection time to download. However, CFT just considers the bidirection environment. In addition, with CFT CH broadcasts its request to its neighbors; then, neighbors that receive the invitation join the cluster and broadcast the request to invite more vehicles to join the cluster until there are enough vehicles. Therefore, CFT may not able to apply in complicated environment, and it may cause network congestions.

Ref. [12] proposes an evolutionary game theoretic (EGT) framework for clustering and CH selecting. Their protocol is based on game theory. They defined the net utility of a CH to select the CH which may achieve high throughput. A cluster size is added in the utility function for CH to optimize the size of a cluster. Ref. [13] proposes an intelligent naive Bayesian probabilistic estimation practice (ANTSC) method. This method is based on the traffic flow. To increase the stability of cluster, a CH must be in the lane having the heaviest traffic flow. Naive Bayes algorithm is used to select the CH which may make the cluster most stable. However, [12] just compared the EGT clustering with one clustering algorithm proposed in 2010. Both [12, 13] did not consider the security of the network.

3. Proposed Scheme

3.1. Overview and Assumption. Clustering algorithm groups a set of unlabeled nodes into clusters. In cluster-based VANETs, all vehicles send their information to eNodeB.

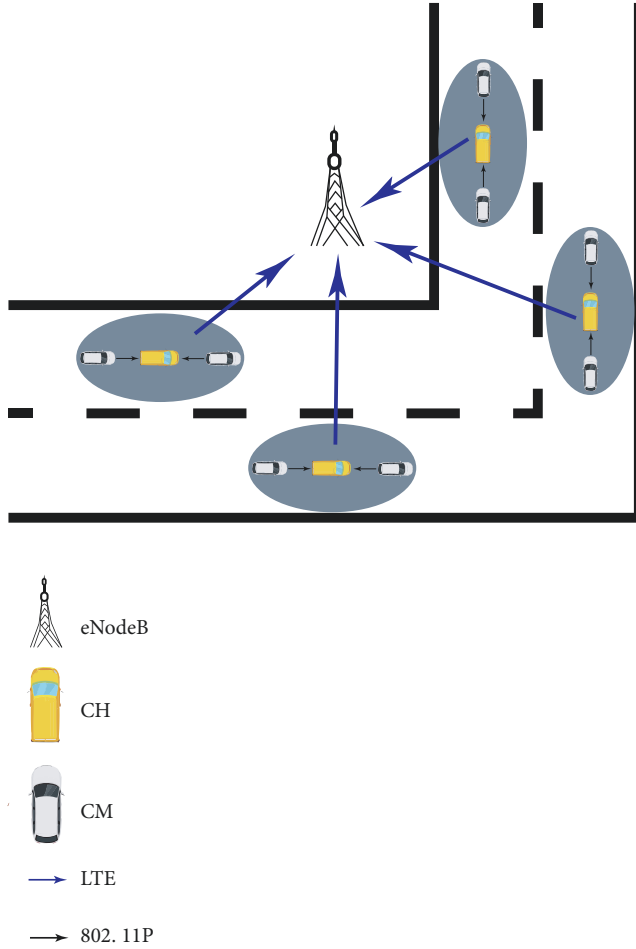


FIGURE 1: Communications within one cluster.

Then, eNodeB manages the vehicles by clusters. A CH acts as a messenger to help eNodeB and CMs exchange information.

We assume all vehicles are able to communicate via both LTE and DSRC. The size of cluster is smaller or equal to the range of 802.11p, so that vehicles in the same cluster can exchange messages via DSRC. DSRC coverage radius is about 300 meters. LTE coverage radius is about 1 kilometer. Therefore, a single eNodeB manages many clusters around it. Within a cluster, a vehicle acts as a CH to collect information of all CMs via 802.11p and exchanges data with the eNodeB via LTE. Figure 1 is a simplified view of a cluster-based vehicular network.

For cluster in this paper, we have some assumptions:

- (1) All vehicles have both LTE and 802.11p interfaces
- (2) All vehicles are equipped with Global Positioning System (GPS) devices. So, they have accurate geolocations
- (3) All vehicles know their destination, speed, and maximal acceleration

Based on the assumptions, we propose a center detection based clustering algorithm. We group the vehicles in the region where the density of vehicles is higher than other areas

into clusters with the help of blob detection method or an improved high-degree algorithm. Some parameters, such as speed and acceleration, are added to the CH selection metric to make the cluster stabler and decrease the CH reselection frequency.

3.2. Cluster Formation. In our proposed algorithm, in the initialization stage of cluster formation, vehicles send beacon messages to the eNodeB. The beacon message of one vehicle contains the vehicle's ID k , current position (x_k, y_k) , current speed v_k , maximal acceleration a_k , and direction type t_k .

Direction type is decided by the angle from the current position to the destination. For vehicle k , whose destination position is (x'_k, y'_k) , the direction angle θ_k is

$$\theta_k = \tan^{-1} \frac{y'_k - y_k}{x'_k - x_k} \quad (1)$$

When $\theta_k \in [0^\circ, 90^\circ)$, $t_k = 1$. When $\theta_k \in [90^\circ, 180^\circ)$, $t_k = 2$. When $\theta_k \in [180^\circ, 270^\circ)$, $t_k = 3$. When $\theta_k \in [270^\circ, 360^\circ)$, $t_k = 4$. Vehicles that have different t are managed, respectively.

The clustering algorithm is described in Algorithm 1.

After receiving the beacon messages, the system analyzes vehicles' position information and detects the centers of the ranges where the vehicle density is higher than in other areas. If the vehicle quantity or the vehicle density is not very large, an improved Highest-Degree Algorithm is applied. Several vehicles which have more neighbors in their transmit range are detected. We improve the original Highest-Degree Algorithm to make sure the distance between any two vehicles we detected is larger than the DRSC range. The positions of detected vehicles will be the centers we use in the clustering algorithm. Otherwise, when the vehicle quantity and the vehicle density are very large, to decrease the computing complexity and analyze time, the system draws dots on the map to indicate vehicles. Then, we can carry out the blob detection. The blob detection is able to detect the regions where the gray pixel value is greater. Thus, we can use the blob detection algorithm, e.g., [14], to detect the centers of regions on the map where dot density is higher.

All vehicles whose distances to the center are not larger than the range of DSRC are labeled as one cluster. Then, the system selects one nearest intersection for every center among all intersections that meet the following conditions:

- (1) The distance from it to the points in P is not smaller than the range of DSRC
- (2) The intersection is not in any cluster's region

Vehicles near those selected intersections are grouped into clusters. Then, eNodeB uses the same way to select intersections near the selected intersections and groups vehicles. After iterations, ungrouped vehicles are grouped into clusters. The distance between two vehicles in the same cluster is not larger than the range of DSRC. To further decrease computing complexity, in line 8 of clustering algorithm, a vehicle or infrastructure located in the center or intersection can broadcast a request to invite neighbors to join the cluster. In line 37, the chosen vehicle e can broadcast an invitation instead of calculating distance by the system.


```

Input: Vehicle set V
Output: Initial clusters
1 Initialize center set  $C = \phi$ ;
2 Locate the centers and add them into C;
3 Initialize point set  $P = C$ ;
4 while  $P \neq \phi$  do
5   foreach point  $p$  in  $P$  do
6     Initialize node set  $cluster_p = \phi$ ;
7     foreach vehicle  $e$  in  $V$  do
8       if  $d_{ep} \leq R$  then
9         Add  $e$  into  $cluster_p$ ;
10        Remove  $e$  from  $V$ ;
11      end
12    end
13    if  $cluster_p \neq \phi$  then
14      Call Algorithm 2;
15      Return set  $cluster_p$ ;
16      Estimate  $S_{ic}$  of the CH  $c$  of  $cluster_p$ ;
17      if  $S_{ic}$  is remarkable high then
18        Check all nodes in  $cluster_p$  to detect attacker;
19      end
20    else
21       $c$  check  $S_a$  value of each CM;
22      if  $S_{am}$  is remarkable high then
23        Report to the server;
24      end
25    end
26    Add the intersection nearest to  $p$  which meets the conditions into  $P$ ;
27  end
28  Remove  $p$  from  $P$ ;
29 end
30 end
31 while  $V \neq \phi$  do
32   foreach point  $c$  in  $C$  do
33     Select an element  $e$  in  $V$  nearest to  $c$ ;
34     Initialize set  $cluster_e = \{e\}$ ;
35     Remove  $e$  from  $V$ ;
36     Set  $e$  as CH;
37     foreach vehicle  $v$  in  $V$  do
38       if  $d_{ev} \leq R$  then
39         Add  $v$  into  $cluster_e$ ;
40         Remove  $v$  from  $V$ ;
41       end
42     end
43     Return  $cluster_e$ ;
44     Estimate  $S_{ic}$  of the CH  $e$  of  $cluster_e$ ;
45     if  $S_{ic}$  is remarkable high then
46       Check all nodes in  $cluster_e$  to detect attacker;
47     end
48   else
49      $c$  check  $S_a$  value of each CM;
50     if  $S_{am}$  is remarkable high then
51       Report to the server;
52     end
53   end
54 end
55 end

```

ALGORITHM 1: Clustering algorithm.

3.3. Cluster Head Selection. Compared to other MANETs, VANETs have lower stability, because of the high mobility of vehicles. Although we divide the vehicles with the help of direction vector \vec{v}_k , the stability of clusters cannot be guaranteed. To select an appropriate CH which can increase the cluster lifetime and decrease the CH reselecting frequency, a relative mobility metric M is introduced for CH election.

For a vehicle k , which is in the cluster $cluster_i$, the position difference between it and all other N vehicles in the same cluster $cluster_i$ is

$$D_k = \sum_{n=1}^N \sqrt{(x_k - x_n)^2 + (y_k - y_n)^2} \quad (2)$$

The speed difference between k and all other N vehicles in the same cluster is

$$V_k = \sum_{n=1}^N |v_k - v_n| \quad (3)$$

The maximal acceleration difference between k and all other N vehicles in the same cluster is

$$A_k = \sum_{n=1}^N |a_k - a_n| \quad (4)$$

The relative mobility metric M is

$$M_k = \alpha \frac{D_k}{\max \{D_n \mid \forall n \in C_i\}} + \beta \frac{V_k}{\max \{V_n \mid \forall n \in C_i\}} + \gamma \frac{A_k}{\max \{A_n \mid \forall n \in C_i\}}, \quad (5)$$

where α , β , and γ are weighted coefficients. $\alpha + \beta + \gamma = 1$. They can be adjusted to fit the different traffic conditions. When the traffic condition is good and all vehicles are driving at a similar speed, the distance between vehicles has a greater effect. Thus, the value of α should be higher than the other two. When vehicles are driving at high speed, the value of β is higher than the other two. When the vehicles enter an area which speed limit changes continually, a higher γ should be considered.

The relative mobility metric M evaluates the relative position, speed, and maximal acceleration differences between one vehicle and all other vehicles in the same cluster. A smaller M indicates the vehicle has lower relative mobility than other vehicles in this cluster. Algorithm 2 explains the process of Cluster Head selection. All clusters formed with the help of centers and intersections use Algorithm 2 to select CH. As a CH, the vehicle's relative mobility metric is smaller than any CMs. That means the motion mode of CH is similar to the whole cluster.

3.4. Cluster Maintenance and Reforming. The unpredictability and mobility of traffic make the cluster lifetime temporary. It is infeasible to reform clusters in real time or very frequently. To minimize the frequency and overhead of cluster

Input: Vehicles in one cluster
Output: Cluster head o of the corresponding cluster

```

1 Set  $M_{min} = +\infty$ ;
2 foreach vehicle  $k$  do
3   Calculate the relative mobility metric  $M_k$ ;
4   if  $M_k < M_{min}$  then
5      $M_{min} = M_k$ ;
6      $o = k$ ;
7   end
8 end
9 return  $o$ ;
```

ALGORITHM 2: Cluster head selection algorithm.

reforming, we propose a cluster maintenance algorithm. Algorithm 3 explains the cluster maintenance process.

(1) *No Connections between CH and CM.* When a CH cannot connect to a CM, the CH will delete the CM from its record and notice eNodeB. When a CM cannot reach its CH, the CM will check the signal it received via DSRC and join the cluster whose signal of CH is strongest. If the CM cannot receive a message strong enough, it will notice eNodeB via LTE and become a CH.

(2) *No Connections between eNodeB and CH.* When eNodeB notices it has lost connection to a CH, it recalls Cluster Head Selection Algorithm and a new vehicle will be CH of that cluster instead of the leaving vehicle.

(3) *A Vehicle Joins the Network.* When a vehicle comes into the network, it first tries to join the nearest cluster by broadcasting a CH request via DSRC. If it fails, it will send a message to eNodeB. eNodeB will help the vehicle to join a cluster or to be a CH and form a new cluster by itself.

(4) *Two Clusters Are Too Close.* With the movement of the vehicles, two clusters may be very close. When the distance between two CHs is shorter than R for a period Δt , the two clusters are merged into one cluster. The Cluster Head Selected Algorithm is recalled. A new CH for the new cluster is selected. Then, all vehicles, which are out of the transmission range of the new CH, leave this cluster and check the invitation signal they have received via DSRC and join the cluster whose signal of CH is the strongest. If a vehicle does not find a cluster to join in, it notices eNodeB via LTE and becomes a CH.

3.5. Security Mechanism. To further improve the VANETs security and availability, a novel security mechanism is proposed to detect malicious nodes.

In clustered networks, the availability and security of CHs are incredibly crucial. CHs help the servers to collect and transmit messages to CMs. If an attacker wants the access to other vehicles' private information, it should act as a CH. The most common and most executable method for an attacker to be selected as a CH is launching a

```

Input: Initial clusters and vehicle set V
Output: New clusters
1 if the eNodeB can not reach a CH then
2   Call Cluster Head Selection Algorithm;
3 end
4 if the CH can not reach a CM then
5   Remove the CM;
6   Notice eNodeB;
7 end
8 if the distance between two CHs  $\leq R$  for a period  $\Delta t$  then
9   Merge the two clusters into one cluster;
10  Call the Cluster Head Selected Algorithm;
11  Check  $S_t$  values of new CHs;
12  New CHs check  $S_a$  values of their CMs;
13 end
14 if a CM can not reach the CH then
15   if it can receive a signal from CHs then
16     Join the cluster whose signal of CH is strongest;
17     CH check its  $S_a$  value;
18   end
19   else
20     Notice eNodeB;
21     The node performs as a CH;
22   end
23 end

```

ALGORITHM 3: Cluster maintenance algorithm.

Sybil attack. In a Sybil attack, the vehicle controlled by a malicious attacker presents multiple identities (vehicles), and all of the vehicles have similar directions, positions, speeds, and maximal acceleration. Hence, these vehicles must have higher relative mobility metrics and higher probabilities to be selected as CH.

To protect the CMs' privacy, we introduce a trajectory similarity metric S_t to defend Sybil attacks. For a cluster contains N nodes, the trajectory similarity metric of its CH c is

$$S_{tc} = \frac{1}{N-1} \sum_{i=1}^{N-1} \left(\frac{\Delta t_i}{lifetime_i} \right), \quad (6)$$

where Δt_i is the duration of both c and node i that belong to the same cluster, and $lifetime_i$ is the lifetime of i in this network. Every time a CH is selected, the server estimates its trajectory similarity metric. If one CH has a remarkable higher trajectory similarity metric, the server will check all nodes in the cluster to detect the malicious attacker.

A denial-of-service attack (DoS attack) is another common attack in VANETs. In DoS attack, the attacker floods the CH or server to make the network services unavailable. The connections of authenticated vehicles to the network are temporarily broken. Therefore, the legitimate requests of server and authenticated vehicles cannot be actioned.

To protect the network availability, we introduce an activity similarity metric S_a to detect DoS attacks. For a

vehicle m that belongs to a cluster containing N nodes, the activity similarity metric is

$$S_{am} = \frac{\sum_{i=1}^{N-1} p_i - p_m}{\sum_{i=1}^{N-1} p_i} \cdot \frac{N}{N-1}, \quad (7)$$

where p_i is the number of requests between vehicle i and CH c during a period of time Δt ; p_m is the number of requests between vehicle m and CH c in the same time period. The higher the activity similarity metric, the greater the proportion of requests of vehicle m in all the requests of this cluster. If one CM has a remarkable higher similarity metric, the CH will regard it as a DoS attacker and report to the server. Meanwhile, the sever can also use activity similarity metrics to detect DoS attack from CHs.

4. Performance Evaluation

4.1. Simulation Parameter. We perform the simulation with the help of Veins LTE. Veins LTE is a simulator developed on Veins [15], which is an open source framework for simulation of vehicular networks based on both IEEE 802.11p and LTE. It integrates a network simulator named OMNeT++ and a traffic simulator named Simulation of Urban MObility (SUMO) [16].

In our experiment, vehicles run on a real map of Washington, DC, USA, obtained from OpenStreetMap [17]. We extract the data of highways in the center of Washington, DC. The total length of road is 30.38 km. The total lane length is

90.09 km. Every vehicle has random source and destination edge. The route from the starting point to the destination is the shortest path found by Dijkstra's algorithm [18]. The maximal acceleration ability of vehicles we have used is $2.6 m/s^2$. The maximal deceleration ability of vehicles is $4.5 m/s^2$. The vehicle's maximum velocity is $55.55 m/s$.

We compare our proposed clustering algorithm, Center-Based Stable Clustering Algorithm (CBSC), with a K-Means-Based method (KMB) and SCAE algorithm [19]. K-means algorithm [20] is commonly used in VANETs for clustering, e.g., [21–23]. In KMB method, we divide the vehicles into two parts by the angle of the vehicles and perform KMB on them, respectively. The cluster maintenance algorithm KMB is proposed in [24]. The predefined threshold Δv_{th} is $5 m/s$. In the simulations, all vehicles' movement information is resent to eNodeB for cluster status update in every 10 seconds. eNodeB needs exchange data with vesicles every 3 seconds. The simulation time is 503 seconds.

4.2. Results and Analysis. The goal of this paper is to propose a stable clustering algorithm for VANETs. To check whether a clustering algorithm can solve the high mobility of vehicles on the highways, the cluster stability should be evaluated. The metrics we use to show the performance of clustering algorithm are as follows:

- (1) *Average CH Lifetime.* The CH lifetime is the period from the state when the vehicle is a CH to the state when it is not a CH (i.e., being a CM or leaving the system). When a CH ends its lifetime, a new CH is elected, or the cluster is dissolved.
- (2) *Average CM Lifetime.* CM lifetime represents the duration at which a CM stays in the same cluster. The average CM lifetime is the average length of all vehicles' CM lifetime. It is another important metric to evaluate the stability of clusters.
- (3) *Average Number of Reaffiliation Times per Vehicle.* The average number of reaffiliation times per vehicle represents the average number of times a vehicle changes the cluster it belongs to during the simulation time.
- (4) *Packet Loss Rate.* Packet loss rate is the percentage of packets lost with respect to packets sent.

In the experimentation, we compare the four metrics of the three methods with different vehicle numbers, transmission ranges, or highway speed limits. Figures 2, 4, 6, and 8 show the results obtained with the variety of total vehicle number (N) and the variety of transmission range (R), when the highway speed limit (v) is $100 km/h$. Figures 3, 5, 7, and 9 show the results obtained with the variety of transmission range (R) and the variety of highway speed limit (v), when the total vehicle number (N) is 300.

Figures 2 and 3 represent the average CH lifetime for the three methods. Those figures show that the CHs under KMB have a marked shorter lifetime. Although our CBSC has a higher value than SCAE a few times, in general, SCAE performs slightly better than CBSC on the average CH lifetime.

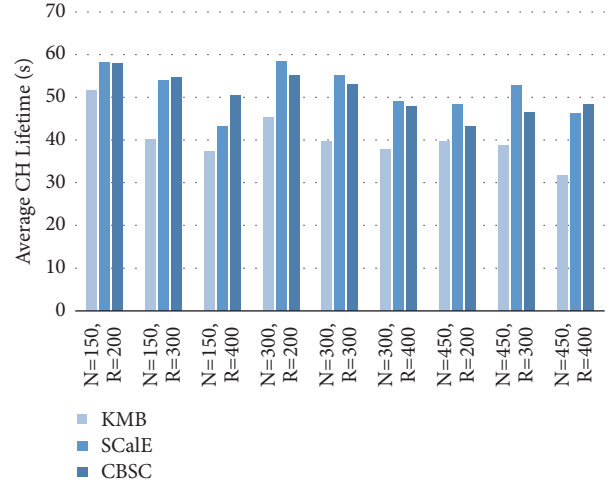


FIGURE 2: Average CH lifetime versus N and R.

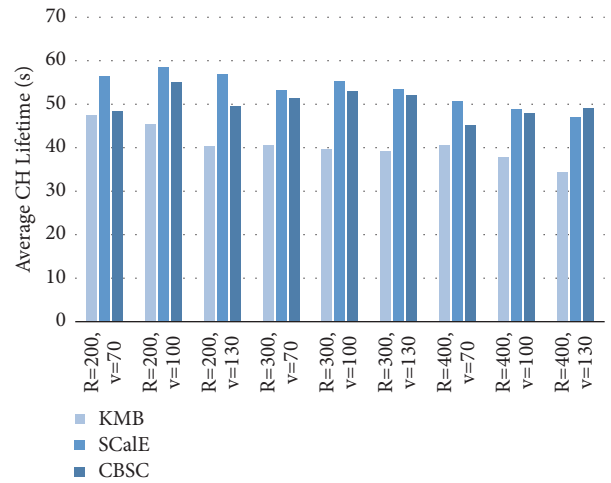


FIGURE 3: Average CH lifetime versus R and v.

The average CM lifetime values produced by KMB, SCAE, and the CBSC methods are shown in Figures 4 and 5. From those two figures, we can see that the average CM lifetime produced by CBSC is much longer than the other two methods. SCAE has the worst performance on the average CM lifetime.

Figures 6 and 7 show the average number of reaffiliation times per vehicle obtained in 503 seconds. Obviously, comparing to other two algorithms, vehicles with SCAE change status much more frequently. The data on the two figures shows CBSC not only produces a lower cluster status change frequency than KMB produces, its superiority, but also is bigger with the increase in highway speed limit.

The results of simulation illustrate that clusters under CBSC are the stablest in the three algorithms. They have the longest average CM lifetime and lowest average number of reaffiliation times per vehicle. Although SCAE performs slightly better than CBSC on the CH lifetime experiment, it produces a much shorter average CM lifetime. Besides, the number of CMs is much larger than the CHs in one system.

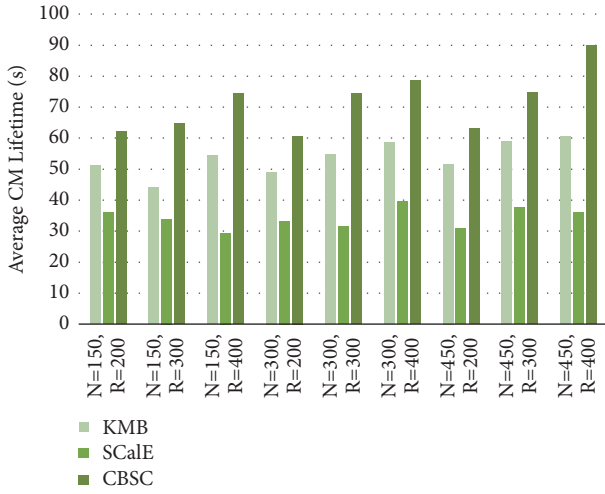


FIGURE 4: Average CM lifetime versus N and R.

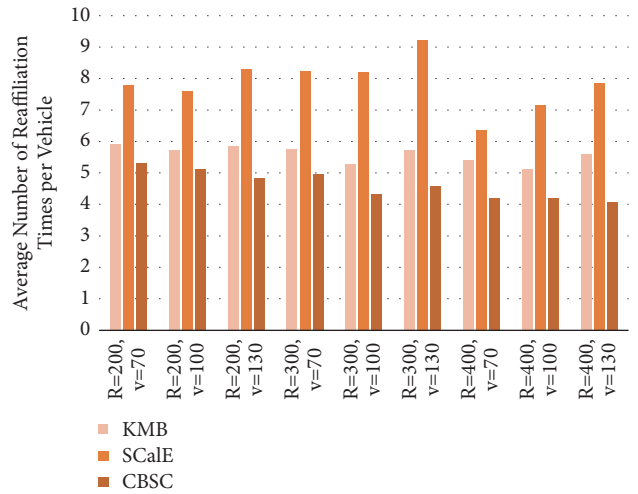


FIGURE 7: Average number of reaffiliation times per vehicle versus R and v.

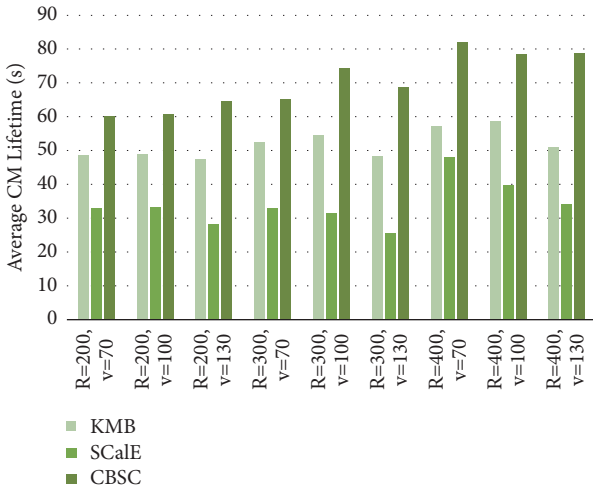


FIGURE 5: Average CH lifetime versus R and v.

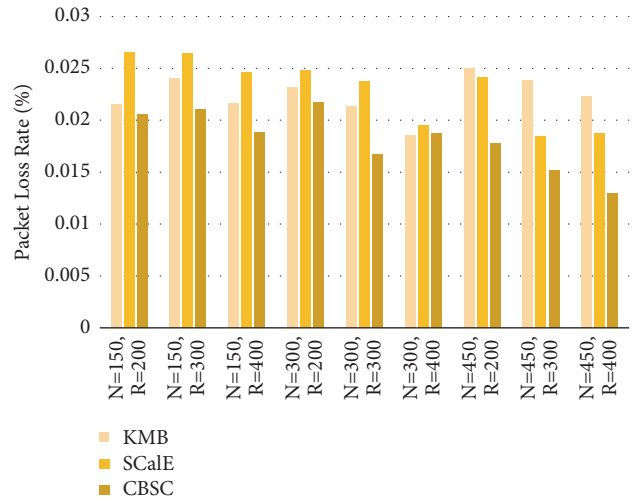


FIGURE 8: Packet loss rate versus N and R.

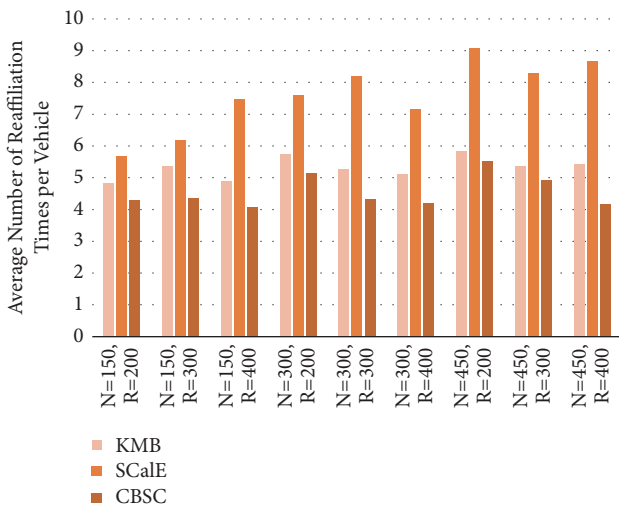


FIGURE 6: Average number of reaffiliation times per vehicle versus N and R.

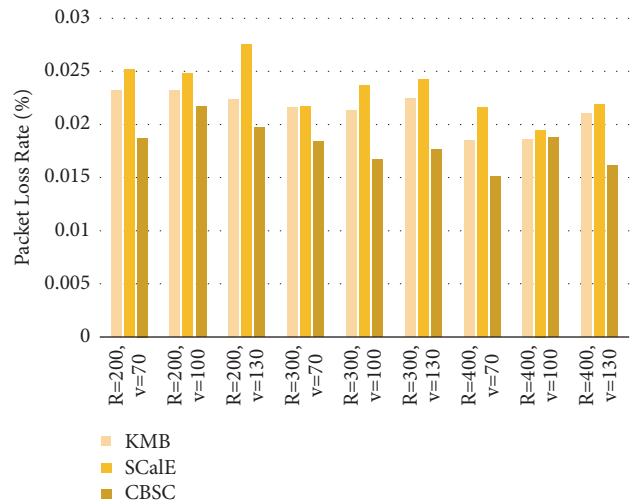


FIGURE 9: Packet loss rate versus R and v.

Therefore, we consider that CBSC has higher stability than ScaE.

The basic function of VANETs is supporting communication between separated vehicles and infrastructures. To test the performance of data dissemination in VANETs, we do experiment on packet loss rate with different methods. Packet loss means a packet fails to arrive at its destination. A high packet loss rate decreases the data dissemination efficiency and may cause network congestion. Therefore, an efficient data dissemination mechanism should have a low packet loss rate. In our experiment, all vehicles exchange data with eNodeB every three seconds. That means, in every three seconds, eNodeB sends data to all vehicles once, and each vehicle sends data to eNodeB once. Like the scene we described in the previous section, eNodeB communicates with the nodes in its record via CHs, and vehicles which are CMs send data to their CHs first. Figures 8 and 9 show the results of packet loss rate. With the increase in vehicle velocity or the transmission range, the packet loss rates obtained by all the three mechanisms decrease. But CBSC gets lower packet loss rate, while KMB performs the worst, when the amount of vehicle is larger. That insinuates CBSC has a good ability to handle a considerable amount of data. In the experiment, CBSC always obtains lowest packet loss rate. Since the interval between cluster status updates is 10 seconds, we can know that the probability of CM leaving its CH between two cluster status updates in CBSC is lower than the other two algorithms. We can consider that the proposed relative mobility metric M and CH selection algorithm of CBSC do reduce the impact of vehicle mobility on cluster stability.

5. Conclusion

To decrease the management overhead and increase the quality of communications, we try to make the clusters in VANETs as stable as possible while keeping the network performance acceptable. In this paper, we propose a stable clustering algorithm for VANETs on highways, which utilizes direction vector, the centers of vehicle denser areas, and intersections to group less quantity of more stable clusters. To reduce the impact of vehicle type and drivers' driving habits, we propose a novel CH selection algorithm and cluster maintenance algorithm, which use the relative mobility metric to reduce the influence of vehicle's distance, velocity, and maximal acceleration. To protect the vehicles' privacy and the network availability, we introduce two mechanisms to detect malicious attacker. In the simulation experiment, our algorithm's performance ranks up against the other two algorithms (KMB and ScaE) on both stability and package delivery rate. In the future, we would like to further improve the algorithm for the complex urban environment.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular Ad Hoc network," *Journal of Network and Computer Applications*, vol. 37, no. 1, pp. 380–392, 2014.
- [2] J. Luo and J. P. Hubaux, "A survey of inter-vehicle communication," Tech. Rep., 2004.
- [3] Y. J. Li, "An overview of the dsrc/wave technology," in *Proceedings of the International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, pp. 544–558, Springer, 2010.
- [4] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang, and Y. Zhou, "Heterogeneous vehicular networking: a survey on architecture, challenges, and solutions," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2377–2396, 2015.
- [5] G. Remy, S.-M. Senouci, F. Jan, and Y. Gourhant, "LTE4V2X: LTE for a centralized VANET organization," in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference: "Energizing Global Communications", GLOBECOM 2011*, pp. 1–6, Houston, TX, USA, December 2011.
- [6] A. Memedi, F. Hagenauer, F. Dressler, and C. Sommer, "Cluster-based transmit power control in heterogeneous vehicular networks," in *Proceedings of the IEEE Vehicular Networking Conference, VNC 2015*, pp. 60–63, December 2015.
- [7] Y. Li, M. Zhao, and W. Wang, "Intermittently connected vehicle-to-vehicle networks: detection and analysis," in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference (GLOBECOM '11)*, pp. 1–6, December 2011.
- [8] L. Zhang, Y. Deng, W. Zhu, J. Zhou, and F. Wang, "Skewly replicating hot data to construct a power-efficient storage cluster," *Journal of Network and Computer Applications*, vol. 50, pp. 168–179, 2015.
- [9] P. Dong, X. Du, J. Sun, and H. Zhang, "Energy-efficient cluster management in heterogeneous vehicular networks," in *Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHOPS)*, pp. 644–649, San Francisco, Calif, USA, April 2016.
- [10] S. L. Hakimi, "Optimum distribution of switching centers in a communication network and some related graph theoretic problems," *Operations Research*, vol. 13, pp. 462–475, 1965.
- [11] Q. Luo, C. Li, Q. Ye, T. H. Luan, L. Zhu, and X. Han, "CFT: A Cluster-based File Transfer Scheme for highway VANETs," in *Proceedings of the ICC 2017 - IEEE International Conference on Communications*, pp. 1–6, Paris, France, May 2017.
- [12] A. A. Khan, M. Abolhasan, and W. Ni, "An Evolutionary Game Theoretic Approach for Stable and Optimized Clustering in VANETs," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 4501–4513, 2018.
- [13] A. Mehmood, A. Khanan, A. H. H. M. Mohamed, S. Mahfooz, H. Song, and S. Abdullah, "ANTSC: An Intelligent Naïve Bayesian Probabilistic Estimation Practice for Traffic Flow to Form Stable Clustering in VANET," *IEEE Access*, vol. 6, pp. 4452–4461, 2017.
- [14] T. Lindeberg, "Feature detection with automatic scale selection," *International Journal of Computer Vision*, vol. 30, no. 2, pp. 79–116, 1998.
- [15] C. Sommer, D. Eckhoff, R. German, and F. Dressler, "A computationally inexpensive empirical model of IEEE 802.11p radio shadowing in urban environments," in *Proceedings of the 8th International Conference on Wireless On-Demand Network Systems and Services (WONS '11)*, pp. 84–90, January 2011.

- [16] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo-simulation of urban mobility: an overview," in *Proceedings of the SIMUL 2011, The Third International Conference on Advances in System Simulation*, ThinkMind, 2011.
- [17] (OSMF), O. F., "Openstreetmap." <http://www.openstreetmap.org/#map=14/38.9004/-77.0282>.
- [18] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
- [19] G. V. Rossi, Z. Fan, W. H. Chin, and K. K. Leung, "Stable clustering for Ad-Hoc vehicle networking," in *Proceedings of the 2017 IEEE Wireless Communications and Networking Conference, WCNC 2017*, pp. 1–6, March 2017.
- [20] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no 14, pp. 281–297, Oakland, CA, USA, 1967.
- [21] E. Ben Hamida and M. A. Javed, "Channel-aware ECDSA signature verification of basic safety messages with K-means clustering in VANETs," in *Proceedings of the 30th IEEE International Conference on Advanced Information Networking and Applications, AINA 2016*, pp. 603–610, March 2016.
- [22] Q. Zhang, M. Almulla, Y. Ren, and A. Boukerche, "An efficient certificate revocation validation scheme with k-means clustering for vehicular ad hoc networks," in *Proceedings of the 2012 IEEE Symposium on Computers and Communications (ISCC)*, pp. 000862–000867, Cappadocia, Turkey, July 2012.
- [23] R. Chai, X. Ge, and Q. Chen, "Adaptive K-Harmonic Means clustering algorithm for VANETs," in *Proceedings of the 14th International Symposium on Communications and Information Technologies, ISCIT 2014*, pp. 233–237, IEEE, September 2014.
- [24] Z. Y. Rawashdeh and S. M. Mahmud, "A novel algorithm to form stable clusters in vehicular ad hoc networks on highways," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, p. 15, 2012.