Wireless Communications and Mobile Computing

Recent Advances in Cloud-Aware Mobile Fog Computing

Lead Guest Editor: Fuhong Lin
Guest Editors: Lei Yang, Ke Xiong, and Xiaowen Gong

Recent Advances in Cloud-Aware Mobile Fog Computing

Wireless Communications and Mobile Computing

Recent Advances in Cloud-Aware Mobile Fog Computing

Lead Guest Editor: Fuhong Lin
Guest Editors: Lei Yang, Ke Xiong, and Xiaowen Gong

Copyright © 2019 Hindawi. 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.

Editorial Board

Javier Aguiar, Spain
Ghufran Ahmed, Pakistan
Wessam Ajib, Canada
Muhammad Alam, China
Eva Antonino-Daviu, Spain
Shlomi Arnon, Israel
Leyre Azpilicueta, Mexico
Paolo Barsocchi, Italy
Alessandro Bazzi, Italy
Zdenek Becvar, Czech Republic
Francesco Benedetto, Italy
Olivier Berder, France
Ana M. Bernardos, Spain
Mauro Biagi, Italy
Dario Bruneo, Italy
Jun Cai, Canada
Zhipeng Cai, USA
Claudia Campolo, Italy
Gerardo Canfora, Italy
Rolando Carrasco, UK
Vicente Casares-Giner, Spain
Luis Castedo, Spain
Ioannis Chatzigiannakis, Italy
Lin Chen, France
Yu Chen, USA
Hui Cheng, UK
Ernestina Cianca, Italy
Riccardo Colella, Italy
Mario Collotta, Italy
Massimo Condoluci, Sweden
Daniel G. Costa, Brazil
Bernard Cousin, France
Telmo Reis Cunha, Portugal
Igor Curcio, Finland
Laurie Cuthbert, Macau
Donatella Darsena, Italy
Pham Tien Dat, Japan
André de Almeida, Brazil
Antonio De Domenico, France
Antonio de la Oliva, Spain
Gianluca De Marco, Italy
Luca De Nardis, Italy
Liang Dong, USA
Mohammed El-Hajjar, UK
Oscar Esparza, Spain
Maria Fazio, Italy
Mauro Femminella, Italy
Manuel Fernandez-Veiga, Spain
Gianluigi Ferrari, Italy
Ilario Filippini, Italy
Jesus Fontecha, Spain
Luca Foschini, Italy
A. G. Fragkiadakis, Greece
Sabrina Gaito, Italy
Óscar García, Spain
Manuel García Sánchez, Spain
L. J. García Villalba, Spain
José A. García-Naya, Spain
Miguel Garcia-Pineda, Spain
A.-J. García-Sánchez, Spain
Piedad Garrido, Spain
Vincent Gauthier, France
Carlo Giannelli, Italy
Carles Gomez, Spain
Juan A. Gómez-Pulido, Spain
Ke Guan, China
Antonio Guerrieri, Italy
Daojing He, China
Paul Honeine, France
Sergio Ilarri, Spain
Antonio Jara, Switzerland
Xiaohong Jiang, Japan
Minho Jo, Republic of Korea
Shigeru Kashihara, Japan
Dimitrios Katsaros, Greece
Minseok Kim, Japan
Mario Kolberg, UK
Nikos Komninos, UK
Juan A. L. Riquelme, Spain
Pavlos I. Lazaridis, UK
Tuan Anh Le, UK
Xianfu Lei, China
Hoa Le-Minh, UK
Jaime Lloret, Spain
Miguel López-Benítez, UK
Martín López-Nores, Spain
Javier D. S. Lorente, Spain
Tony T. Luo, Singapore
Maode Ma, Singapore
Imadeldin Mahgoub, USA
Pietro Manzoni, Spain
Álvaro Marco, Spain
Gustavo Marfia, Italy
Francisco J. Martinez, Spain
Davide Mattera, Italy
Michael McGuire, Canada
Nathalie Mitton, France
Klaus Moessner, UK
Antonella Molinaro, Italy
Simone Morosi, Italy
Kumudu S. Munasinghe, Australia
Enrico Natalizio, France
Keivan Navaie, UK
Thomas Newe, Ireland
Wing Kwan Ng, Australia
Tuan M. Nguyen, Vietnam
Petros Nicopolitidis, Greece
Giovanni Pau, Italy
Rafael Pérez-Jiménez, Spain
Matteo Petracca, Italy
Nada Y. Philip, UK
Marco Picone, Italy
Daniele Pinchera, Italy
Giuseppe Piro, Italy
Vicent Pla, Spain
Javier Prieto, Spain
Rüdiger C. Pryss, Germany
Sujan Rajbhandari, UK
Rajib Rana, Australia
Luca Reggiani, Italy
Daniel G. Reina, Spain
Jose Santa, Spain
Stefano Savazzi, Italy
Hans Schotten, Germany
Patrick Seeling, USA
Muhammad Z. Shakir, UK
Mohammad Shojafar, Italy
Giovanni Stea, Italy
Enrique Stevens-Navarro, Mexico
Zhou Su, Japan
Luis Suarez, Russia
Ville Syrjälä, Finland

Hwee Pink Tan, Singapore
Pierre-Martin Tardif, Canada
Mauro Tortonesi, Italy
Federico Tramarin, Italy
Reza Monir Vaghefi, USA
Juan F. Valenzuela-Valdés, Spain
Aline C. Viana, France
Enrico M. Vitucci, Italy
Honggang Wang, USA
Jie Yang, USA
Sherali Zeadally, USA
Jie Zhang, UK
Meiling Zhu, UK

Contents

Hindawi
Wireless Communications and Mobile Computing
Volume 2019, Article ID 8204394, 2 pages
https://doi.org/10.1155/2019/8204394

Editorial

Recent Advances in Cloud-Aware Mobile Fog Computing

Fuhong Lin,[1] **Lei Yang**,[2] **Ke Xiong**,[3] **and Xiaowen Gong**[4]

[1]*School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing, 100083, China*
[2]*Department of Computer Science and Engineering, University of Nevada, Reno, NV 89557, USA*
[3]*School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China*
[4]*Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849, USA*

Correspondence should be addressed to Fuhong Lin; fhlin@ustb.edu.cn

Received 4 December 2018; Accepted 5 December 2018; Published 23 January 2019

Copyright © 2019 Fuhong Lin 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.

Mobile fog computing (MFC) is an emerging paradigm that extends cloud computing (CC) by adding a new layer between the cloud and its end users. With the cloud-aware MFC, the cloud can pre-push certain important resources to the fog to reduce the networking latency and release the traffic burden over the links. The end user then is able to perform offline computing on the fog layer so that only the important results need to be delivered to and stored in the cloud. Moreover, the dense geographical deployment of fog servers enables the system to be aware of the end user's location. Therefore, some location-sensitive applications could be well supported by the fog-aided cloud systems. Note that the cloud-aware MFC is different from the mobile edge computing (MEC), another promising technology for overcoming the shortcomings of CC, since MFC is able to jointly work with the cloud, but MEC is usually defined by the exclusion of CC. Specifically, in MEC, computing applications, data, and services are pushed away from the centralized nodes to the network edge, which enables network edge to run in an isolated environment from the rest of the network and provides access to local resources and data. In contrast, MFC provides not only a system-level horizontal architecture but also a new way to distribute, orchestrate, and manage secure resources across the network rather than just performing computing at the network edge.

How to design efficient system architectures, transmission strategies, and protocols for MFC and how to efficiently analyze and evaluate the system performance are very important and essential. These topics have carved out a new area rich in research and innovation potential. This special issue aims to address all these topics and invite contributions from worldwide leading researchers.

This special issue received submissions covering a wide range of topics in MFC. The following is a short summary of the findings of each of these papers.

Y. Zhou et al. took the rural vitalization as an objective and proposed a smart collaborative policy for MFC scenarios; the challenges and drawbacks of extending cloud to fog are reviewed at the beginning. Then, the analysis of policy design is presented from the perspectives of feature comparisons, urgent requirements, and possible solutions.

X. Di et al. designed a resource allocation (RA) algorithm to solve the mobile fog computing-assisted RA problem and demonstrated that the achievable rate is significantly increased by using the proposed RA algorithm.

H. Chen et al. proposed a non-task-specific method for Chinese sentiment classification, which used a new structure convolutional layer to enhance the ability of automatic feature extraction and applied global average pooling layer to prevent overfitting. Through experiments and analysis, they proved the method do achieve competitive accuracy.

X. Tan et al. proposed an Extraction-Inference (E-I) algorithm and built a QoS and user Interest based Engagement (QI-E) regression model. Through experiments on the datasets, they demonstrated that the model reaches an improvement in accuracy by 9.99% over the baseline model. The proposed model has potential for designing QoE-oriented scheduling strategies in various network scenarios, especially in the fog computing context.

W. Chen et al. proposed a novel public auditing protocol based on the adjacency-hash table and demonstrated that dynamic auditing and data updating are more efficient than

those of the state of the arts. Computation and communication costs can be reduced effectively by using the authentication structure.

W. Hui-Juan and J. Yong studied the 2-adic complexity attack ability of the periodic balance sequence in the fog computing environment. They proved that the 2-adic complexity of the periodic balanced sequence is not an attacking threat when used in fog computing by using the exponential function as a new approach.

Y. Wang et al. proposed a method that takes advantage of the immune mechanism to schedule tasks in a decentralized way for fog computing. Distributed schedulers are used to generate the optimal scheduler strategies to deal with overloaded computing nodes and achieve the optimal task finishing time.

H. Zheng et al. proposed a fog-assisted real-time QoE prediction scheme, which can predict the QoE of DASH-supported video streaming using fog nodes. Neither client/server participations nor deep packet parsing at network equipment is needed, which makes this scheme easy to deploy.

H. Zhao et al. took advantage of fog computing and studied transcoding based video caching in cellular networks where cache servers are deployed at the edge of cellular network for providing improved quality of online VoD services to mobile users.

L. Liu et al. proposed a novel classification mining algorithm I-Apriori which is based on the Apriori algorithm to increase productivity and allocate resources appropriately to the tasks. They proposed a novel task scheduling model and a TSFC (Task Scheduling in Fog Computing) algorithm based on the I-Apriori algorithm. Association rules generated by the I-Apriori algorithm are combined with the minimum completion time of every task in the task set.

L. Lu et al. proposed a suboptimal approach for resource allocation of massive MIMO-OFDMA systems for high-speed train (HST) applications. Fast convergence can be achieved for the proposed approach within only several iterations. They showed that the algorithm is superior to existing techniques in terms of system energy efficiency and throughput in different system configurations of HST applications.

Y. Huo et al. proposed a real-time stream data aggregation framework with adaptive-event differential privacy (Re-ADP). In the framework, fog servers will only send aggregated secure data to cloud servers, which can relieve the computing overhead of cloud servers, improve communication efficiency, and protect data privacy.

T. Jing et al. designed an integrated system to prevent illegal privacy leak. They defined a trust degree mechanism to evaluate trustworthiness of a communicator dynamically and set up a new message publishing system to determine who can obtain the message of the publisher. They verified the effectiveness of the proposed message publishing system through analysis of confidentiality performance.

L. Wang and L. Li proposed the fog computing-based differential positioning (FCDP) method which introduces fog computing technology to BeiDou satellite navigation system. Compared with the original data center-based differential positioning method, they demonstrated that the FCDP method decreases the latency of positioning, while ensuring the positioning accuracy.

Y. Fu et al. proposed a novel neighbor-based QoS prediction method for service recommendation and further designed a Nearest Graph algorithm to recognize stable or unstable candidate along with their popularity by a nearest neighbor graph structure which can help make missing QoS values prediction in a certain order to improve final prediction accuracy. Experimental results confirm that the proposed method is effective in predicting unknown QoS values in terms of service recommendation accuracy and efficiency.

S. Li et al. investigated the reliability of concurrent multipath communications in mobile cloud computing (MCC) architectures and proposed two reliability models when paths are in failure. They mainly analyzed that in MCC architectures multipath communications can be achieved with multihomed mobile devices, so as to utilize multiple paths for data transmissions in parallel.

J. Zhang et al. provided a comprehensive overview of key fault tolerance strategies. They mainly summarized three aspects from static fault tolerance strategies, dynamic fault tolerance strategies, and main challenges confronted by fault tolerance for composite service.

Z. Chen et al. proposed a new security scheme based on implicit certificate (IC), which solves the security problem among the access points (APs) in a dynamic APs group (APG) and between the AP and user equipment (UE). They extensively analyzed a lightweight security communication model in terms of security and performance and proved the efficiency of the solution.

Conflicts of Interest

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

Acknowledgments

We would like to thank all reviewers for their great efforts in reviewing the submitted manuscripts, without which this special issue would not have been published as scheduled.

Fuhong Lin
Lei Yang
Ke Xiong
Xiaowen Gong

Hindawi
Wireless Communications and Mobile Computing
Volume 2019, Article ID 1438739, 15 pages
https://doi.org/10.1155/2019/1438739

Research Article

The Construction Method of BeiDou Satellite Navigation Measurement Error System

Jun Xie,[1,2] Jian-jun Zhang,[1,2] and Gang Wang[2,3]

[1]*Beijing Institute of Spacecraft System Engineering, Beijing 100094, China*
[2]*China Academy of Space Technology, Beijing 100094, China*
[3]*China Academy of Space Technology Xi'an Institute of Space Radio Technology, Xi'an 710000, China*

Correspondence should be addressed to Jian-jun Zhang; zhangjianjun@cast.cn

Received 19 June 2018; Revised 25 September 2018; Accepted 2 December 2018; Published 15 January 2019

Guest Editor: Ke Xiong

Copyright © 2019 Jun Xie 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.

Based on the measurement error of pseudorange in BeiDou satellite navigation system, this paper analyzes the measurement principle of the system. Aiming at the difficulties in the system measurement error index system, an overall construction method of measurement error system based on empirical estimation method and error distribution model is proposed. Based on the Analytic Hierarchy Process, the correlation analysis model of the measurement error index is constructed, the relationship between the indicators is analyzed, and the system measurement error index hierarchy is constructed. Based on the empirical estimation method and the error distribution model, the index values are decomposed and assigned based on the final service performance of the system, and a clear representation of the complex relationship of index matching is achieved. Finally, by analyzing the principle of the positioning function in the ground transportation control mode, taking the satellite clock error as an example, the model is decomposed layer by layer and the relevant indicator items are established. The relationship between index terms was studied, and the value of the indicators was quantified. Compared with the actual operating conditions of the current system, the correctness of the method was verified, which provided a basis for the demonstration of the index values of satellite navigation systems. Based on the empirical estimation method and error distribution model as a new type of calculation method, the index system established under a certain set of conditions is reasonable, and it can be applied to the error control adjustment in satellite navigation system engineering construction.

1. Introduction

The Global Navigation Satellite System (GNSS) is satellite-based radio navigation and positioning system. It can provide precise three-dimensional position, three-dimensional speed, navigation data, accurate satellite time reference, and other information for all types of users. It shows more and more important uses in military and civilian applications and its application prospects far beyond people's imagination [1]. Based on the empirical estimation method and error distribution model as a new type of calculation method, the index system established under a certain set of conditions is reasonable, and it can be applied to the error control adjustment in satellite navigation system engineering construction [2–8].

The construction of the BeiDou satellite navigation system is a complex and huge system project. When the BeiDou satellite navigation system is using for position or navigation, there are various errors in the observation and measurement, such as the BeiDou system's own error of the signal which includes satellite ephemeris error and satellite clock error; the propagation error of the BeiDou signal from the satellite to the user receiving antenna includes ionospheric delay correction error and tropospheric delay correction error, etc.; BeiDou user receiver generated signal measurement error, including observation noise error and antenna phase center error. This paper defines the measurement error system of the BeiDou satellite navigation system as an organic combination of error index elements in the measurement process of the BeiDou satellite navigation system [9–15]. It is the correlation and constraint relationship between the performance of a single measurement error indicator and multiple measurement error indicators, and an intrinsic part of the satellite navigation system as well as a crucial foundation for ensuring

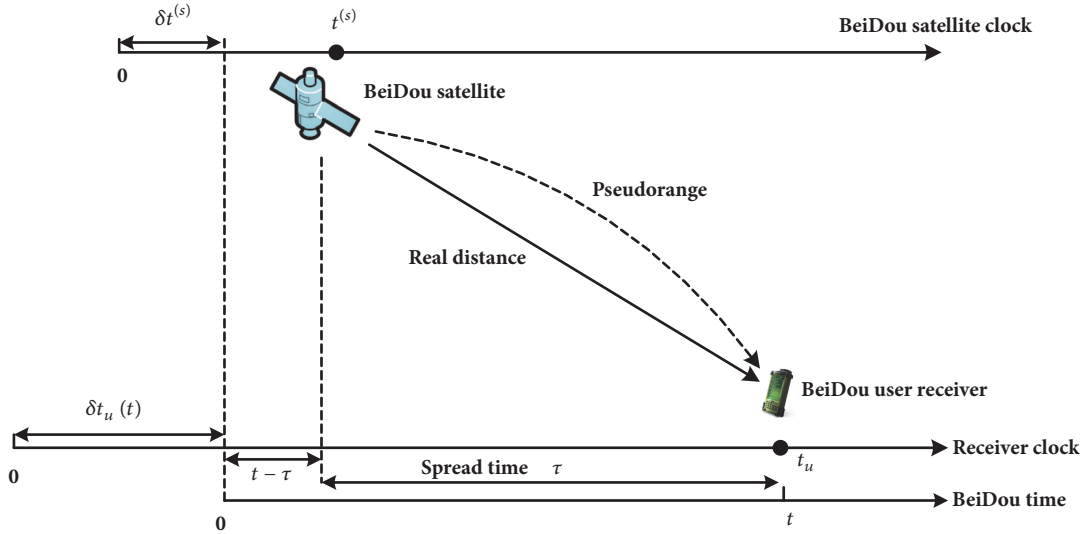


Figure 1: Measurement principle of BeiDou satellite navigation system code pseudorange.

the stable operation of satellite navigation systems. It determines and restricts the functions and performance of satellite navigation systems.

The various measurement errors existing in the BeiDou system will have a certain influence on the measurement accuracy of the system, and the error is unavoidable. However, some techniques can be used to reduce the influence of the error on the measurement accuracy. In order to ensure that the system's final service accuracy meets the design requirements, in the system's demonstration design, various errors need to be repeatedly adjusted and simulated to calculate the system positioning accuracy according to the system's service accuracy requirements and finally be given under the condition of meeting the accuracy index in accordance with the current technical development level of the indicators of the distribution of errors, and the error control of each link in the process of engineering development is guided.

How to start from the system service performance, establish the index system, and clearly describe the complex relationship existing in the index system has become a hot topic in BeiDou satellite navigation system research. Based on the research of BeiDou satellite navigation system measurement error index, this thesis builds a hierarchy of system measurement error indicators guided by Analytic Hierarchy Process, and proposes an overall measurement error system based on empirical prediction method and error distribution model methods to study the relationship between the indicators of the index system and quantify the relevant index values.

2. BeiDou Satellite Navigation System Positioning Principle and Error Analysis

2.1. System Positioning Principle. The principle of the positioning of the BeiDou satellite navigation system is mainly based on the relevant principles of spatial geometry and physical knowledge, using satellites in the spatial distribution and the distance between the satellite and the surface of the earth to calculate the specific location of the ground point. Unlike the ground-based optoelectronic distance measurement method, BeiDou satellite navigation system implements pseudorange measurement [16–19]. In principle, it needs two clocks, one is called a satellite clock and the other is called a user receiver clock (local clock). Because there is a clock error between the satellite clock and the user receiver clock, the distance measurement is called a pseudorange. In order to facilitate the analysis of the influence of various errors on the measurement accuracy, errors are usually attributed to the pseudorange parameter measurement of the satellite and can be regarded as the equivalent error of the pseudorange value. This paper reanalyzed the physical definition of the code pseudorange and the various error source parameters combed out [20–23]. Assume that the time is called the transmission time when the satellite transmits the signal, and the time is called the reception time when the user receives the signal. Figure 1 shows BeiDou satellite navigation system signal code pseudorange measuring principle.

Usually, the user clock is not synchronized with the BeiDou satellite clock. The BeiDou navigation system time (BDT) is actually equal to t. Then the user receiver time at t is $t_u(t)$, and the difference between the user receiver time and the BeiDou navigation system time is recorded $\delta t_u(t)$; that is,

$$t_u(t) = t + \delta t_u(t) \tag{1}$$

Similarly,

$$t^{(s)}(t) = t + \delta t^{(s)}(t) \tag{2}$$

Pseudorange is defined as the distance between signal reception time and transmission time:

$$\rho(t) = c\left(t_u(t) - t^{(s)}(t-\tau)\right) \tag{3}$$

where τ represents the actual propagation time of the signal from the satellite to the receiver.

Substitute (1) into the above equation; that is,

$$\rho(t) = c\tau + c\left(\delta t_u(t) - \delta t^{(s)}(t-\tau)\right) \tag{4}$$

The actual propagation time of the BeiDou satellite navigation signal consists of two parts: one is the time when the signal travels a geometric distance, and the other is the propagation delay caused by the atmosphere; that is,

$$\tau = \frac{r(t-\tau,t)}{c} + I(t) + T(t) \tag{5}$$

The atmospheric propagation delay can be decomposed into two parts: the ionosphere delay $I(t)$ and the troposphere delay $T(t)$. Substituting (5) into (4),

$$\begin{aligned}\rho(t) = {} & r(t-\tau,t) + c\left(\delta t_u(t) - \delta t^{(s)}(t-\tau)\right) + cI(t) \\ & + cT(t) + \varepsilon_\rho(t)\end{aligned} \tag{6}$$

In the formula, $\rho(t)$ represents the pseudorange observation; $r(t-\tau,t)$ is the real distance from the satellite to the receiver; $\varepsilon_\rho(t)$ represents the noise error. Equation (6) is often referred to as the pseudorange observation equation, which is the basic equation for the user receiver to use a pseudorange to achieve a single point absolute positioning.

BeiDou satellite navigation system error positioning accuracy is

$$M_p = PDOP \times U_p \tag{7}$$

In the formula, PDOP is the spatial position geometric precision factor; U_p is the measuring error factor.

2.2. Measurement Error Analysis. There are many errors in the accuracy of the positioning accuracy produced by the BeiDou satellite navigation system, for example, the orbit of the satellites, the atmospheric refraction, and the BeiDou positioning receivers themselves. These errors have a great influence on the positioning accuracy of the BeiDou satellite navigation system. In the navigation and positioning system, the distance error caused by the measurement error can all be equivalent to the error due to the pseudorange measurement. These errors are collectively referred to as the user equivalent distance error (*UERE*).

In order to analyze the effects of various errors on the accuracy, it is assumed that the error sources affecting the positioning accuracy are all independent, and the satellite approximately is represented as a zero-mean Gaussian random variable whose variance is determined by the sum of the variance of each component. The measurement error factor is satellites *UERE*.

$$UERE = \sqrt{\sigma_1^2 + \sigma_2^2 + \cdots \sigma_n^2} \tag{8}$$

According to formula (6), BeiDou satellite navigation system positioning accuracy is closely related to measurement error. Measurement error is affected by many factors, such as satellite orbit error, satellite clock error, ionosphere error, tropospheric error, multipath, and thermal noise. According to the different statistical characteristics of measurement error, these errors can be divided into accidental error, systematic error, and abnormal errors, as shown in Figure 2.

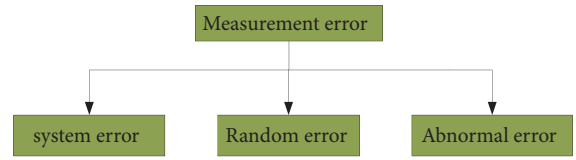


Figure 2: Schematic diagrams of measurement error statistics.

Systematic error refers to the error of a certain law, such as constant system error, periodic system error, etc., caused by the systematic influence of certain factors. The effects of systematic errors are cumulative. For example, dynamic model errors, satellite orbit errors, coordinate system errors, ionosphere delay errors, and tropospheric delay errors are system errors. Random error refers to errors caused by various random factors. Individuals of this type of error are random and irregular, but statistically obey specific statistical rules, such as normal distribution or heavy tail distribution (Huber distribution), for example, satellite ranging error, spatial signal jitter error, etc. Abnormal error refers to the error caused by equipment abnormalities and abnormal changes in observation conditions and is generally expressed in the form of abnormal values. Satellite ranging errors or carrier measurement errors caused by satellite orbit maneuvers, satellite equipment failures, receiver failures, and other extreme errors are abnormal errors.

3. Measurement Error Index Construction Model Based on Analytic Hierarchy Process

3.1. Systematical Measurement Error Level Decomposition Method. The principle of the AHP is to decompose the decision-making first, draw a number of important influencing factors, and classify it to construct a multilevel structural model. After starting from the lower level, analyze the relative importance of the underlying factors to the upper factors and rank according to the degree of importance. The characteristics of AHP are quantification and hierarchization. It decomposes complex problem into a number of relatively simple and small problems, then calculates, and analyzes in turn.

The specific application steps of the AHP are as follows: Determine the specific issues that need to be decided, and decompose the problem into target layer, criterion layer, and indicator layer. The decision-making problem is the target level, the final solution is the indicator level, and the criteria level includes the key criteria that need to be considered when making decisions, as shown in Figure 3.

According to the Analytic Hierarchy Process (AHP), the measurement error elements of the BeiDou satellite navigation system are decomposed into system objectives, criteria, and indicators. Based on this, qualitative and quantitative analysis is performed. The complex system measurement error problem is represented as an ordered hierarchical structure. The hierarchical analysis structure is shown in

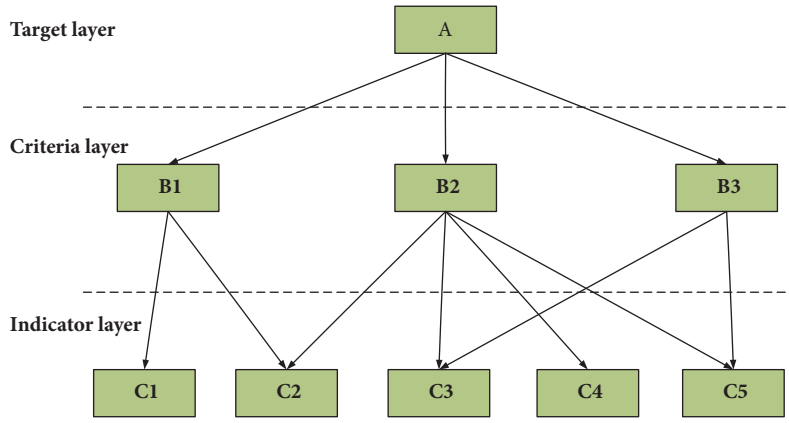


Figure 3: Hierarchical decomposition.

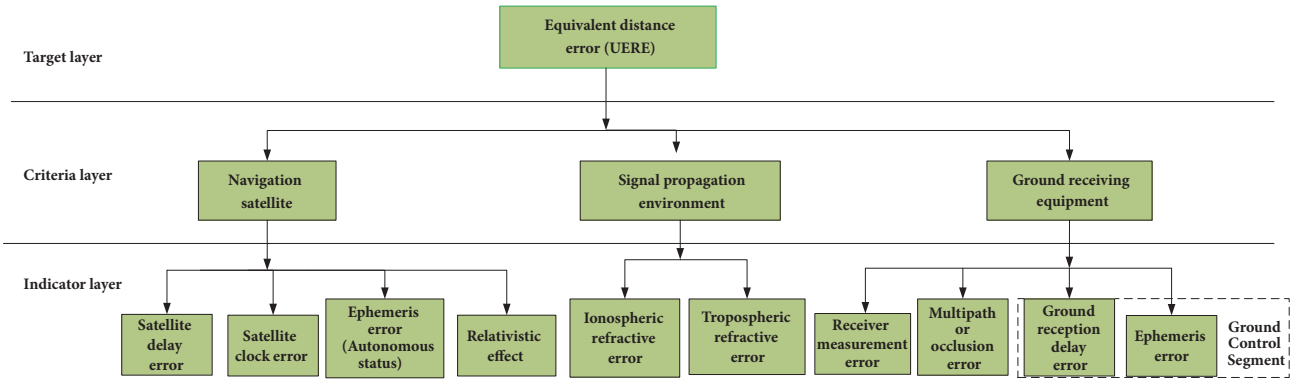


Figure 4: AHP determines the measurement error indicator.

Figure 5. The energy efficiency status of navigation satellites, signal propagation environments, and ground-based receiving equipment directly reflect the source of the measurement error and the level of energy efficiency of the entire system. Well, therefore, navigation satellites, signal propagation environments, ground-based receiving equipment (including BeiDou user receiver and ground transport accused of BeiDou receiver), and the above three (navigation satellite, signal propagation environments, and ground-based receiving equipment) of different period of coupling are difficult to define clearly the error as the criterion layer 2 indicators.

At the same time, satellite delay error, satellite error, its ionosphere, troposphere, and BeiDou user's receiver noise error, multipath error, and many other factors have great impact on measurement error energy efficiency, subdividing them into the 3rd factor indicator level according to the rule, as shown in Figure 4.

3.2. System Measurement Error Indicator Correlation Analysis Model. There is a quantitative relationship between the BeiDou satellite navigation system measurement indicators in the physical sense, but the changes in the visual relationship between any two measurement and the linkage relationship between the measurement error indicator data and through one or a few indicators if it is possible to distinguish the other indicators of changes in laws or is a measure of whether changes depend on one or several indexes such as relationship remain to be further analyzed.

The quantitative relationship of the measurement error indicators of the BeiDou satellite navigation system can be divided into two types: one is a deterministic relationship, which is called a functional relationship; the other is an uncertainty relationship, called the correlation relationship. The functional relationship between the system measurement error indicators is determined by the physical characteristics of the BeiDou satellite navigation system operation and the definition of measurement error characteristic indicators. The correlation relationship reflects the relevant forms and correlation degree of the studied variables or reflects the regular when the variable changes, the other variable will follow the law of the corresponding change, and the value of this change is uncertain. Therefore, the initial search for such uncertainties in the BeiDou satellite navigation system measurement error index can be determined by doing correlation analysis for the measurement error characteristic indices and finding the correlation coefficient between the error characteristic index data.

The correlation analysis of measurement error index data is hoped to mine the inherent law hidden in statistical index data through data mining technology. In most cases, the correlation analysis we perform is performed between the two indicators. This requires the use of a binary variable correlation analysis. Different types of variable data should

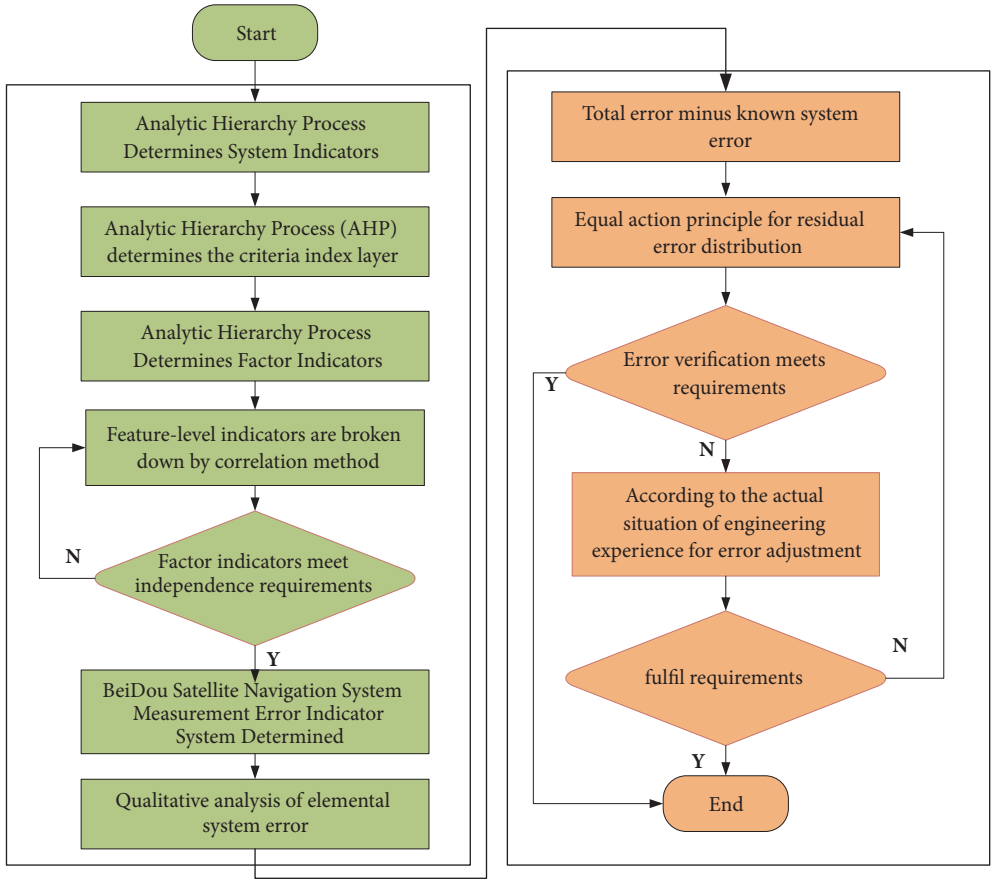


Figure 5: Flow chart of measurement error indicator construction model.

use different correlation analysis methods. When the statistical indicators of measurement error are used for correlation analysis, the error indicators are generally numerical variables. Therefore, the Pearson correlation coefficient analysis method is used to determine the correlation coefficient between the load characteristics law.

Let two random variables be X and y, then the correlation coefficient of the two variables is

$$\rho = \frac{\operatorname{cov}(X, Y)}{\sqrt{\operatorname{var}(X)}\sqrt{\operatorname{var}(Y)}} \tag{9}$$

where $\operatorname{cov}(X, Y)$ is the covariance of two variables; $\operatorname{var}(X)$ $\operatorname{var}(Y)$ are the variance of X and Y; the overall correlation coefficient is a measure of the correlation coefficient between the two variables.

However, in fact, the overall correlation coefficient is generally unknown and needs to be estimated using the sample correlation coefficient. For measurement error characteristics, let $X = (x_1, x_2, \cdots, x_n)$, $Y = (y_1, y_2, \cdots, y_n)$ be the two time series from measurement error characteristic index X and index Y, respectively, then the correlation coefficient between indexes is r:

$$r = \frac{\sum_{i=1}^{n} (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \overline{x})^2 (y_i - \overline{y})^2}} \tag{10}$$

$i = 1, 2, \cdots, n$, $\overline{x}$ and $\overline{y}$ represent the mean values of X and Y sequences, and the measurement error characteristic index data sample correlation coefficient ρ is a uniform estimator of the overall correlation coefficient of index.

3.3. System Measurement Error Distribution Model. The establishment of the measurement error indicator of the BeiDou satellite navigation system is actually the process of error allocation based on the system service design requirements. The allocation of errors should take into account the distribution of all error components. For the BeiDou satellite navigation system measurement error, according to the "BeiDou II" satellite navigation system engineering construction experience, and for a given system error, the impact can be removed first from the total error, and then the remaining random error and the undetermined systematic error distribution problems are analyzed. If the error allowable range of each index item is determined, other errors are assigned according to the determined error. If the error allowable range of the index item cannot be determined, it is distributed according to the equal action principle.

The principle of equal action is to first consider that the error of each part of the BeiDou satellite navigation system has equal influence on the overall error. That is to say, when allocating errors, the error factors are all random errors and are not related to each other, then the error transfer formula of

Table 1: Correlation analysis of factor level indicators.

	r11	r12	r13	r14	r15	r16	r17	r18
r11	1.00	0.15	0.08	0.72	0..57	0.59	0.41	-0.83
r 21	0.15	1.00	0.04	0.09	-0.11	0.06	03	0.19
r31	0.08	-0.04	1.00	0.05	0.12	0.07	0.01	-0.10
r41	0.72	0.09	-0.05	1.00	0.52	0.49	0.32	0.65
r51	0.57	0.11	0.12	0.52	1.00	0.02	0.07	0.81
r61	0.59	0.06	0.07	0.49	-0.02	1.00	0.08	0.85
r71	0.41	0.03	-0.01	0.32	0.07	-0.08	1.00	0.87
r81	0.83	0.19	0.10	0.65	-0.81	0.85	-0.87	1.00

the arbitrary function $y = f(x_1, x_2, \cdots, x_m)$ of the unrelated variable is

$$\sigma_y = \sqrt{\sum_{i=1}^{m} \left(\frac{\partial f}{\partial x_i}\right)^2 \sigma_i^2} = \sqrt{\sum_{i=1}^{m} (D_i)^2} \tag{11}$$

In the formula $D_i = |\partial f / \partial x_i| \sigma_i$, σ_i is the fractional error of the direct measurement. According to the requirement of error distribution, when given σ_y, determine the value of D_i or σ_i, that needs to meet the formula $\sqrt{D_1^2 + D_2^2 + \cdots + D_m^2} \le \sigma_y$, according to the principle of distribution, that requires

$$D_1 = D_2 = \cdots = D_m = \sqrt{\frac{\sigma_y^2}{m}} = \left|\frac{\partial f}{\partial x_i}\right| \sigma_i \tag{12}$$

The distribution of errors by the principle of equal action may appear unreasonable, because the calculated local errors are all equal. This point is easy to achieve for some measurement values to ensure that its measurement error is not beyond the allowable range, and some of these measurement values are difficult to meet the requirements; to meet its measurement accuracy, it is bound to use expensive high-precision instruments or to pay a larger labor. From (9), we can see that when the error of each direct quantity is fixed, the corresponding measurement error is inversely proportional to the error transfer coefficient. Therefore, each measurement error is not equal, and sometimes the phase difference may be large when the local errors are equal. Because of the above two kinds of situations, the errors allocated to the principle of equal action must be adjusted according to the specific circumstances. For error terms that are difficult to guarantee during measurement, the allowable error values should be appropriately expanded. For errors that are easy to guarantee in measurement, the allowable error values should be reduced as much as possible. After the error is adjusted, the total error should also be calculated according to the error distribution formula to see if it exceeds the allowable value of the given function error.

3.4. Measurement Error Index Construction Model. Selecting the appropriate process for accuracy analysis and error index allocation is the key to the design process. The process flow shown in Figure 5 has been used in the analysis of the actual model and achieved the effect of the engineering application.

4. Results and Discussion Analysis on Measurement Errors of BeiDou Satellite Navigation System

4.1. Relevance Decomposition of Systematic Measurement Error Factor Layer Indicators. In order to analyze the correlation between measurement indicators of the BeiDou satellite navigation system error indicators and determine whether it meets the principle established by the index system, we can use a method of calculating correlations to assess the correlation between index items.

Take BeiDou satellite navigation system data released as an example. The data used includes ① satellite clock error (I1), ② satellite delay error (I2), ③ relativistic effect error (I3), ④ satellite ephemeris error (I4), ⑤ ionospheric refractive error (I5), ⑥ tropospheric refractive error (I6), ⑦ multipath or occlusion error (I7), and ⑧ BeiDou satellite navigation receiver measurement error (I8); in the ground control state, ground control section errors are contained within ephemeris errors and satellite clock errors. The correlation calculation results are shown in Table 1.

Calculate the correlation value r according to formula (10). The value of r is between -1 and 1 and describes the degree and direction of the linear correlation between the two measurement error characteristic indices: $r > 0$, there is a positive correlation between the two measurement error characteristic indicators; $r < 0$, there is a negative correlation between the two measurement error characteristic indices; $r = \pm 1$, there is a complete correlation between the two load characteristic indexes; $r = 0$, there is no linear correlation between the two measurement error characteristic indexes. According to experience, the degree of correlation is divided into the following situations: when $|r| > 0.8$, it can be regarded as a high degree of correlation between the load characteristics; when $0.5 \le |r| \le 0.8$, it can be regarded as a measurement error with a moderate degree of correlation; when $0.3 \le |r| \le 0.5$, it is considered that the measurement error characteristic index is low-degree related; when $|r| < 0.3$, the measurement error characteristic index correlation is extremely weak; it may be regarded as irrelevant. The degree of correlation between the measurement error characteristic indicators is determined according to the size of the correlation coefficient, and sorting is performed to remove irrelevant indexes.

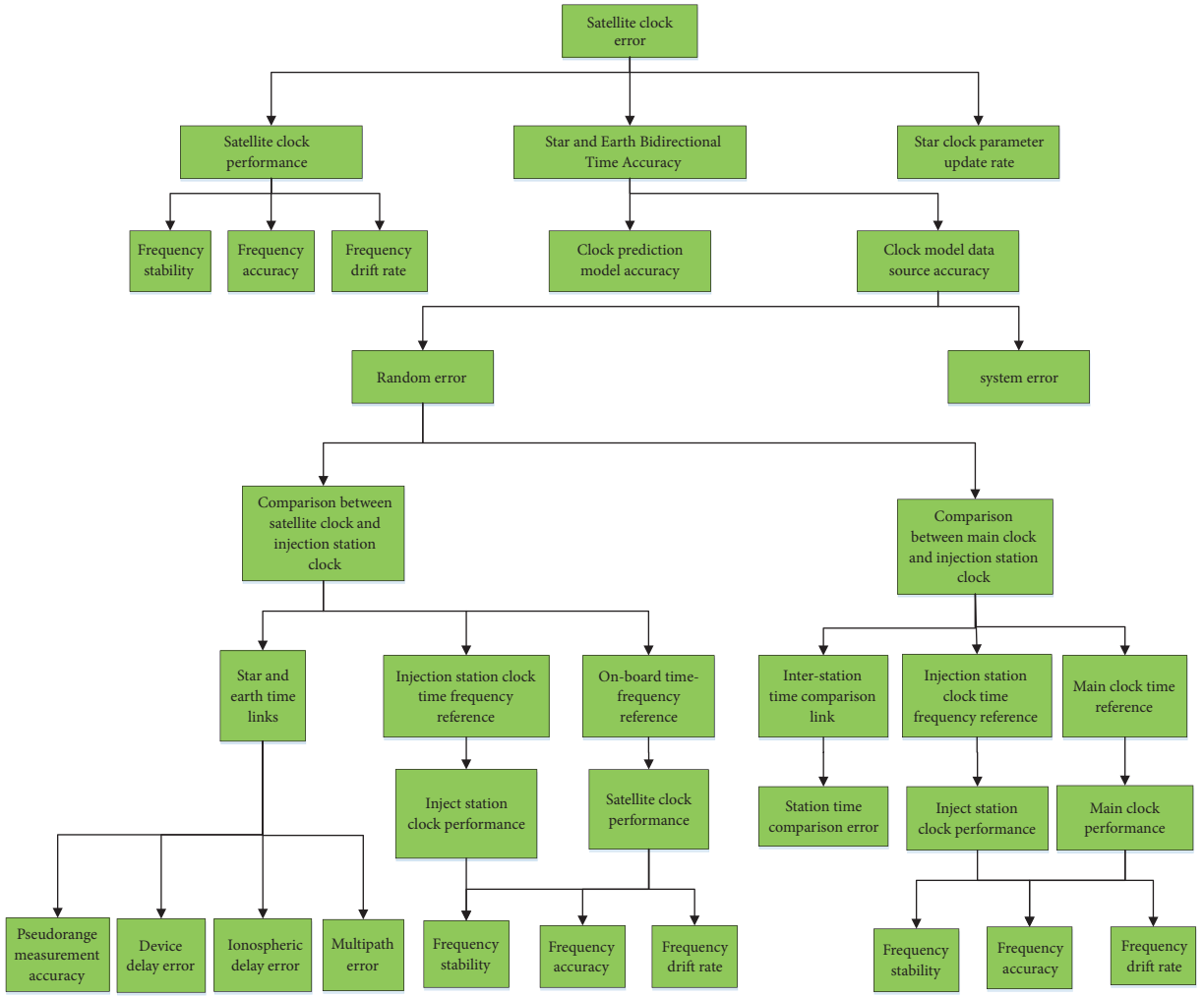


Figure 6: Satellite clock error decomposition indicators.

Table 1 has certain significance, but it does not fully represent the correlation of measurement error indicators. Here, only typical examples are provided for the construction of model methods. According to Table 1, there is a strong correlation between the satellite clock error and other error indicators. That is, the satellite clock error is used as an example for further decomposition until the disaggregated indicator items are independent of each other. The error sources related to satellite clock error errors mainly include the performance of the satellite clock, the accuracy of satellite-to-earth time comparison, and the update rate of the star clock parameters. The satellite clock errors are related to the decomposition index items, as shown in Figure 6.

4.2. Index Item Allocation Based on Empirical Estimation Method

4.2.1. Estimated Distribution of Experience for Each Indicator Item. The availability of the BeiDou satellite navigation system is related to the cut-off angle used by the receiver. Decreasing the cut-off angle can lead to better usability. However, lowering the cut-off angle to observe more satellites will introduce larger atmospheric errors, so a reasonable selection of cut-off angles should be made on the basis of reaching the availability index. According to statistics of system availability at different cut-off angles, it can be seen that when the cut-off angle is 5°, the availability is greater than 98%, which can meet the general demand for availability. Therefore, to ensure generality, when the PDOP value is calculated from the measurement error index value of this paper, the cut-off angle is 5°. At this time, when the PDOP is less than or equal to 2.5, the system is available and the system availability is good. The availability in China is 100%. The system has been able to better meet the positioning and navigation needs in China and its neighboring regions.

The error positioning accuracy of the BeiDou satellite navigation system is related to the geometric accuracy factor of the spatial position and the error factor for the measurement of U_p. In the ground operation control mode, the positioning accuracy of the system is better than 10m, and

Table 2: Relativity correction error caused by inaccurate.

$t - t_k$	σ_s	σ_d	$\sigma(t)$	**Corresponding ranging error**
4 hours	$1 \times 10^{-15}/s$	1×10^{-15}/day	0.17ps	**5.1E-5m**
1 day	$1 \times 10^{-15}/s$	1×10^{-15}/day	0.45ps	/

the position accuracy factor of the system is PDOP=2.5, then

$$UERE = 10m/2.5 = 4m \tag{13}$$

From Figures 3 and 5, the square root of the sum of the squares of the measurement error index of the indicator layer is 4m; that is,

$$\sigma_{UERE} = \sqrt{\sigma^2_{\text{satellite delay}} + \sigma^2_{\text{satellite clock}} + \sigma^2_{\text{The theory of relativity}} + \sigma^2_{\text{ephemeris}} + \sigma^2_{\text{The ionosphere}} + \sigma^2_{\text{troposphere}} + \sigma^2_{\text{receiver}} + \sigma^2_{\text{multipath}}} \tag{14}$$

① *Satellite Delay Error Indicator Allocation*. Delay from the zero point of the satellite system to the output of the signal conversion circuit, the delay from the output of the signal conversion circuit to the output of the wave modulator, the delay from the output of the microwave modulator to the output of the power amplifier, the time delay from the output of the power amplifier to the phase center of the antenna, and the error between the signals, the phase deviation of the transmitting antenna phase, and the satellite-to-earth time ratio error on-board are the navigation satellite delay errors. At present, the impact of satellite delay error on positioning is expected to be within 0.2ns.

② *Distribution of Relativistic Effect Error Indicators*. According to the principle of relativity, clock oscillators at different speeds of motion will produce frequency offsets, and clock oscillators with different gravitational bits will generate gravitational shifts. During BeiDou satellite navigation and positioning surveys, due to the different statuses of the BeiDou satellite clock and the receiver clock, their movement speed and gravitational force are different.

The frequency stability expression of a satellite-borne atomic clock can be approximated by $\sigma_y(\tau) = \sigma_s\tau^{-1/2} + \sigma_d$ (where σ_s is the second stability, σ_d is the day stability, and τ is the measurement interval). The satellite clock time offset variance $\sigma^2(t)$ caused by it is determined by the time interval and calibration method of satellite clock synchronization calibration (the ground clock is generally based on the hydrogen clock), $\sigma^2(t) = {\sigma_s}^2 \times (t - t_k) + {\sigma_d}^2 \times (t - t_k)^2$ (where t is the current time and t_k is the calibration time). Table 2 shows the relationship between σ_s, σ_d, synchronization calibration intervals $t - t_k$ and $\sigma(t)$, and ranging error.

Therefore, the satellite has been revised, and the influence of relativity on the stability of the bell is below 1E-15. This item can be ignored.

③ *Allocation of Satellite Ephemeris Error Indicators*. Satellite ephemeris error is also called satellite orbit error. Estimating and processing satellite orbital errors is more difficult because satellites are subject to the combined effects of multiple perturbations in orbital operations, and it is difficult for ground monitoring systems to accurately grasp the changing laws of these forces. The BeiDou system satellites are equipped with laser reflectors. The accuracy of the satellite laser reflectors can reach 1 to 2cm, and the existing orbit determination technology and the perturbation model have been improved; therefore, the accuracy of precision orbit determination can theoretically reach the order of decimeters and even centimeters. However, since all three ground monitoring stations of the BeiDou satellite navigation system are located in China, there are few tracking arcs for the satellites, and the distribution is extremely uneven. Therefore, using the observation data from 3 monitoring stations to determine the accuracy of the track is difficult to be improved. At present, the influence of ephemeris errors on positioning is expected to be controlled within 1 m.

④ *Ionospheric Refractive Error Index Allocation*. Ionospheric refractive errors are errors in observations due to ionospheric effects. When the BeiDou satellite navigation signal passes through the ionosphere, the path of the navigation signal will be bent, the propagation speed will also change, the carrier propagation speed will be accelerated, and the code propagation speed will be reduced, so that the measured distance will be deviated. This effect is called for ionospheric refraction. Our country is in the midlatitudes of the northern hemisphere, some regions in the south are located in the anomalous areas of the equator, the difference in the elevation angle between the antenna and the satellite oscillates the ionospheric refraction error, and an average estimate of the ionospheric refraction error index needs to be performed. At this time, taking the ionospheric refraction at an angle of 30o as the average value, the ionospheric refractive error is about 20m, and the equivalent error distance is about 6m, as shown in Figure 7. And the division of ionospheric grids in China is shown in Figure 8.

Using the Klobuchar model to correct ionospheric time refraction, the average effective rate reaches over 70% in the midlatitudes of the northern hemisphere. Combined with an ionospheric error grid correction algorithm and using a grid model with an interval of 5°×5°, the vertical ionospheric delay at the user station's longitude and latitude at the point of penetration was calculated. It is concluded that the residual error of atmospheric refraction after the ionospheric model

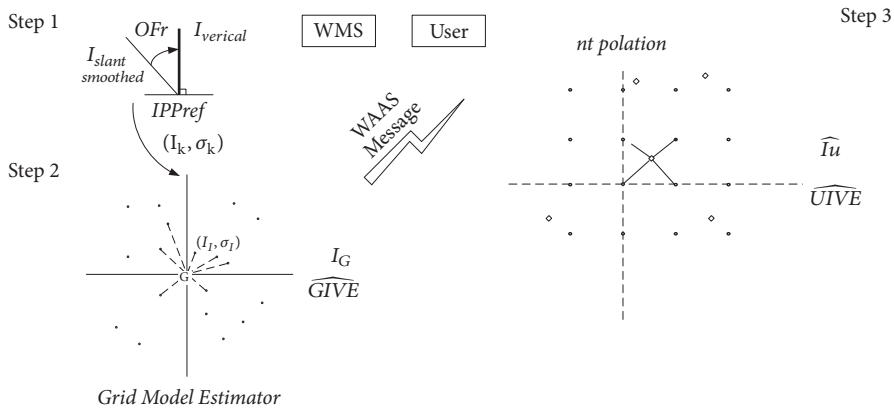


Figure 7: Ionospheric grid model.

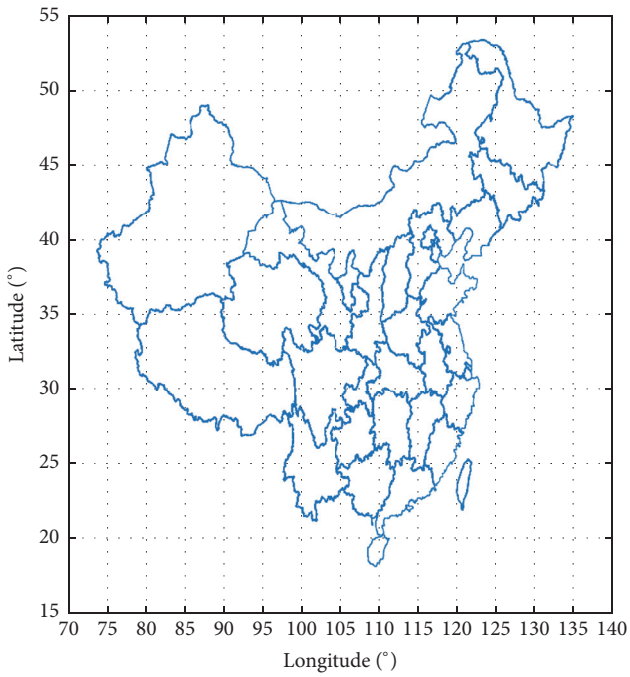


Figure 8: Division of ionospheric grids in China.

parameter correction is expected to be controlled within 3m.

⑤ *Tropospheric Refractive Error Index Allocation*. Because the atmospheric density is greater than the ionosphere and the state of the atmosphere is also more complicated, at the same time, the troposphere is in contact with the ground and receives radiant heat energy from the ground. Its temperature decreases as the altitude increases. Therefore, when the BeiDou satellite navigation signal passes through the troposphere, it will also cause the propagation path to be bent, thus causing a deviation in the measurement distance. This phenomenon is called tropospheric refraction. There is no better way to correct tropospheric delay errors. Usually, corrections are made using models such as Hopfield and Sastomonin. The residual error after this correction is expected to reach lm.

⑥ *Allocation of Multipath or Occlusion Error Indicators*. The error caused by the multipath effect is difficult to eliminate completely and its influence can only be weakened as much as possible. The common practices are to avoid strong reflection surfaces during positioning, use a receiving antenna with anti-multipath effects and use an extended observation time and averaging method. In short, the multipath error can be controlled within 2m using a suitable method.

⑦ *BeiDou Satellite Navigation Receiver Measurement Error Indicator Allocation*. The noise of BeiDou satellite navigation receivers has a wide range of meanings, including the receiver clock skew, code tracking errors caused by thermal noise, interference, etc., also known as pseudorange measurement errors. It also includes the heat of the antenna, amplifier and various electronic devices thermal noise, signal quantization error, cross-correlation between satellite signals, algorithm error in determining code phase and carrier phase, and various calculation errors in receiver software. Based on comprehensive statistics, the impact of the noise error of the former BeiDou satellite navigation receivers on positioning can be controlled within 1 ns.

⑧ *Distribution of Ground Motion Control Segment Error Indicators*. The ground segment of the satellite navigation system is a complex, which is the control center of the entire navigation system. It is a typical mission-critical system and is responsible for the operation and management of the entire navigation system. It is responsible for satellite time synchronization, precision orbit determination, and ionospheric delay processing, system integrity monitoring and wide-area differential processing in key service areas, uplink injection of navigation message parameters, and management and maintenance of satellite constellations and payloads. The impact of systematic errors on positioning is expected to be less than 0.3 ns, within the ground station ranging receiver accuracy and ground transceiver channel calibration error.

4.2.2. Satellite Clock Indicator Item Allocation. According to satellite delay error, relativity error, tropospheric error, tropospheric error, receiver measurement error, and satellite ephemeris error, the satellite clock error indicator is quantitatively calculated:

$$\sigma_{\text{Satellite clock error}} = \sqrt{\sigma_{UERE}^2 - \left(\sigma_{\text{Satellite delay}}^2 + \sigma_{\text{Satellite clock}}^2 + \sigma_{\text{The theory of relativity}}^2 + \sigma_{\text{ephemeris}}^2 + \sigma_{\text{ionosphere}}^2 + \sigma_{\text{troposphere}}^2 + \sigma_{\text{receiver}}^2 + \sigma_{\text{multipath}}^2\right)} \quad (15)$$

By substituting the above-mentioned qualitatively assigned indicators into the above equation, the equivalent distance error of the satellite clock error can be obtained:

$$\begin{aligned}\sigma_{\text{Satellite clock error}} &= \sqrt{4^2 - \left(0.06^2 + 1 + 3^2 + 1 + 2^2 + 0.3^2\right)} \\ &\approx 0.952m\end{aligned} \quad (16)$$

That is, in order to achieve a positioning service accuracy of 10 meters, the satellite clock bias forecast accuracy is better than 3.2 ns when the system is performing a satellite clock bias forecast (between the two forecasts, the precision error caused by the clock difference should be less than 0.952m).

4.3. Error Allocation Modeling of Satellite Clock Error Index Value. There are frequency and phase deviation and phase noise between the carrier, pseudocode signal, and nominal carrier and pseudorange signal by navigation satellites. The jitter error of the navigation signal at the receiver is mainly reflected in the ability to capture and distinguish the signal. The front-end of the receiver removes the interference signals of adjacent frequency bands through signal filtering. At the same time, in order to better capture weak signals, the signal power should be put in about 1010-1011 times. That is, if the carrier frequency error is less than 1E-11Hz, then the receiver cannot discriminate the carrier frequency error during signal acquisition and recovery and has no effect on system service accuracy. The second-order stability of the satellite-borne time-frequency reference 10.23M signal is generally 0.5 to 1 order of magnitude higher. According to the development of the satellite-borne atomic clock of the BeiDou satellite navigation system, the second-order stability of the satellite-borne time-frequency reference 10.23M signal is 5E-12, and the corresponding second-order atomic clock satellite stability is at least 5E-12.

According to Figure 7, the indicators related to satellite clock error indicators include satellite clock performance, satellite-to-ground bidirectional time comparison, and satellite clock parameter update rates. The update rate of the satellite clock parameter is a constraint indicator, but it plays a key role in the control of the satellite clock error. In general, the higher the satellite clock parameter update rate is, the higher the satellite clock accuracy is, and at the same time, it can compensate for the error caused by insufficient satellite clock performance. In this paper, the forecasting strategy of one-hour forecasting for one hour is used to decompose the indicator value to improve the update rate of the star clock parameter.

4.3.1. Distribution of Satellite Clock Performance Error Indicators. According to atomic clock noise characteristics, the satellite clock is stable at 5E-12 seconds. The frequency stability within the range of tens of seconds is related to the time interval τ-1/2. Since the observation strategy is to observe 1 hour forecast for 1 hour, the 1-hour stability of the satellite atomic clock is

$$\sigma(3600s) = \sqrt{\left(\frac{1}{3600}\right)} \times \sigma(1s) = 8.3E-14 \quad (17)$$

The calculation results based on the error estimation formula $\Delta t = \sigma(\tau) \times \tau$ represent the ultimate accuracy of the model prediction. The actual model prediction results will have a certain degree of precision attenuation. In order to ensure the validity of the actual accuracy, a margin of 30% must be reserved, and the performance of the satellite atomic clock can be obtained (1-hour stability) as

$$\sigma(3600s) = 8.3E-14 \times (1-30\%) \approx 5.8E-14 \quad (18)$$

Similarly, the other stability of the satellite clock can be estimated as follows:

Second stability:

$$\sigma(1s) = 5E-12 \quad (19)$$

Ten seconds stability:

$$\sigma(10s) = \left(\frac{1}{10}\right)^{1/2} \times \sigma(1s) \times (1-30\%) \approx 1.1E-12 \quad (20)$$

100 second stability:

$$\begin{aligned}\sigma(100s) &= \left(\frac{1}{100}\right)^{1/2} \times \sigma(1s) \times (1-30\%) \\ &= 3.5E-13\end{aligned} \quad (21)$$

Thousand seconds stability:

$$\begin{aligned}\sigma(1000s) &= \left(\frac{1}{1000}\right)^{1/2} \times \sigma(1s) \times (1-30\%) \\ &\approx 1.1E-13\end{aligned} \quad (22)$$

Million second's stability:

$$\begin{aligned}\sigma(10000s) &= \left(\frac{1}{10000}\right)^{1/2} \times \sigma(1s) \times (1-30\%) \\ &= 3.5E-14\end{aligned} \quad (23)$$

Day stability:

$$\sigma(86400s) = \left(\frac{1}{86400}\right)^{1/2} \times \sigma(1s) \times (1 - 30\%) \approx 2E - 14 \tag{24}$$

The effect of satellite atomic clock performance (1 hour stability) on satellite clock error is

$$5.8 \times 10^{-14} \times 3.0 \times 10^{8} \times 3600 = 0.06m \tag{25}$$

4.3.2. Decomposition of the Two-Dimensional Time-to-Time Ratio Accuracy Index. According to the principle of error distribution, the two-way time accuracy error of satellites and satellites is

$$\begin{aligned}\sigma_{\substack{\text{The precision of the two-way}\\ \text{time comparison}}} &= \sqrt{\sigma^2_{\text{Satellite clock error}} - \sigma^2_{\text{Satellite clock performance}}} \\ &= \sqrt{0.952^2 - 0.06^2} \approx 0.95m\end{aligned} \tag{26}$$

The precision of the two-way time comparison between the star and the earth is divided into the accuracy of the clock error forecast model and the source accuracy of the clock error forecast model. Among them, the application of the clock error prediction model is using some linear models as commonly used models, including polynomial models, gray models and time series models. The error introduced by the characteristics of the clock error prediction model is 0.1ns to 0.01ns. If it exceeds 0.1ns, the forecast model has no use value. Taking the maximum error of 0.1ns and converting the equivalent distance error to 0.03 m, the data source accuracy of the clock error prediction model is

$$\begin{aligned}\sigma_{\substack{\text{the precision of the data source of the}\\ \text{bell difference prediction model}}} &= \sqrt{\sigma^2_{\substack{\text{precision of bell difference}\\ \text{prediction}}} - \sigma^2_{\substack{\text{precision of bell difference}\\ \text{prediction model}}}} \\ &= \sqrt{0.95^2 - 0.03^2} \approx 0.94m\end{aligned} \tag{27}$$

4.3.3. Clock Error Forecast Model Data Source Precision Index Item Quantization Decomposition. The error introduced by the data source accuracy of the clock bias forecast model is divided into systematic error and random error. The systematic error mainly refers to the systematic error introduced by the time comparison link in the clock difference data observation process. According to the principle of error allocation

$$\begin{aligned}\sigma_{\substack{\text{the precision random error of the data source}\\ \text{of the bell difference prediction model}}} &= \sigma_{\substack{\text{error of accuracy system of data source}\\ \text{of bell difference prediction model}}} \\ &= \sqrt{\frac{1}{2}\sigma^2_{\substack{\text{the precision of the data source of the}\\ \text{bell difference prediction model}}}} \approx 0.66m\end{aligned} \tag{28}$$

① *Clock Error Forecast Model Data Source Accuracy Random Error*. According to Figure 7, the random error of the data source of the clock error prediction model is mainly caused by the error caused by the time comparison between the satellite clock and the injection station and the error of the injection station clock compared with the main clock.

$$\begin{aligned}\sigma_{\substack{\text{satellite clock and injection station}\\ \text{clock time contrast random error}}} &= \sigma_{\substack{\text{the injection station clock and the main}\\ \text{clock time contrast random error}}} \\ &= \sqrt{\frac{1}{2}\sigma^2_{\substack{\text{the precision of the data source of the}\\ \text{bell difference prediction model}}}} \approx 0.46m\end{aligned} \tag{29}$$

② *Comparison of the Quantification of Random Error Indicator Terms between Satellite Clocks and Monitoring Stations*. According to the correlation decomposition, the random error between the satellite clock and the monitoring station clock can be divided into satellite-to-ground link random error, satellite clock time-frequency reference performance, and injection station clock time-frequency reference performance.

Since the hypothetical observation strategy is observation for 1 hour and forecast for 1 hour, the performance of the satellite-borne time-frequency reference is directly related to the 1-hour stability of the satellite clock. According to the above chapter, the performance of the satellite clock is decomposed, and the 1-hour stability of the satellite clock is $\sigma(3600s) = 8.3E - 14 \times (1 - 30\%) \approx 5.8E - 14$. The equivalent distance error is $(5.8E - 14) \times 3E8m/s \times 3600s \approx 0.06m$.

The performance of the injected station clock is slightly better than that of the satellite clock, but it is reflected in the user equivalent distance error, which is almost equal.

According to the principle of error allocation,

$$\begin{aligned}\sigma_{\substack{\text{the time of the star is compared}\\ \text{with the random error of link}}} &= \sqrt{\sigma^2_{\substack{\text{injection station clock and satellite}\\ \text{clock time contrast random error}}} - \sigma^2_{\substack{\text{satellite clock time-frequency}\\ \text{standard}}} - \sigma^2_{\substack{\text{injection station}\\ \text{clock time base}}}} \\ &= \sqrt{0.46^2 - 0.06^2 - 0.06^2} \approx 0.45m\end{aligned} \tag{30}$$

The equivalent distance error of the time-to-space error of the star-to-ground random error is 0.45m; that is, the time-to-station random error of the satellite-to-ground time is 1.5ns. From the point of view of model analysis, the

time-to-station comparisons between stations and satellites are equivalent to random errors.

④ *Star-Time Comparison of Random Error Indicator Terms*. According to Figure 7, using correlation decomposition, the link-to-link random error of satellites and satellites includes pseudorange measurement accuracy, equipment delay error, ionospheric delay error, and multipath error, etc., according to the principle of error allocation:

$$\begin{aligned}\sigma_{\substack{\text{accuracy of pseudo distance}\\ \text{measurement}}} &= \sigma_{\substack{\text{equipment delay}\\ \text{error}}} \\ &= \sigma_{\substack{\text{ionospheric delay}\\ \text{error}}} = \sigma_{\substack{\text{multipath}\\ \text{error}}} \\ &= \sqrt{\frac{1}{4}\sigma^2_{\substack{\text{star time comparison}\\ \text{random error}}}} \approx 0.22m\end{aligned} \tag{31}$$

According to formula (31), the user equivalent distance error of pseudorange measurement accuracy, device delay error, ionospheric delay error, and multipath error is 0.22m, which is 0.7ns.

⑤ *The Main Clock Performance Index Quantitative Decomposition*. Since the observation strategy is observation for 1 hour and forecast for 1 hour, it can be first concluded that the 1-hour stability of the main bell is equal to the stability of the satellite clock for 1 hour; that is, the second stability is 5E-12. However, in practice, when the main control station selects the main clock, the selected main clock is at least half a second higher than the satellite clock, so the second-degree stability assigned to the main clock is 1E-12. According to the foregoing chapter's index decomposition theory of bell performance can estimate the stability indicator of the main bell:

Seconds stability:

$$\sigma(1s) \approx 1E-12 \tag{32}$$

Ten seconds stability:

$$\sigma(10s) = \left(\frac{1}{10}\right)^{1/2} \times \sigma(1s) \times (1-30\%) \approx 2.2E-13 \tag{33}$$

100 second stability:

$$\sigma(100s) = \left(\frac{1}{100}\right)^{1/2} \times \sigma(1s) \times (1-30\%) \approx 1E-13 \tag{34}$$

Thousand seconds stability:

$$\begin{aligned}\sigma(1000s) &= \left(\frac{1}{1000}\right)^{1/2} \times \sigma(1s) \times (1-30\%) \\ &\approx 3E-14\end{aligned} \tag{35}$$

Million seconds stability:

$$\begin{aligned}\sigma(10000s) &= \left(\frac{1}{10000}\right)^{1/2} \times \sigma(1s) \times (1-30\%) \\ &\approx 1E-14\end{aligned} \tag{36}$$

Day stability:

$$\begin{aligned}\sigma(86400s) &= \left(\frac{1}{86400}\right)^{1/2} \times \sigma(1s) \times (1-30\%) \\ &\approx 3E-15\end{aligned} \tag{37}$$

There is a maximum requirement for the constant term of the clock difference prediction model in the navigation message broadcast by the BeiDou satellite navigation system, and the physical deviation between the satellite clock time and the system time should be less than 1 millisecond. If it is higher than 1 millisecond, the atomic clock needs to be physically adjusted, and this adjustment will affect the stability output indicator of the on-board atomic clock. It requires less adjustment, the general adjustment interval should be greater than 100 days, and it can be estimated that the frequency accuracy of the satellite clock is

$$1ms/100d = 1E-10 \tag{38}$$

In order to ensure the reliability of practical applications, it is necessary to reserve a certain margin and take an index value of one order of magnitude higher; that is, the frequency accuracy of the satellite clock is about 1E-11. According to experience, the frequency accuracy of the main clock is 2 orders of magnitude higher than that of the satellite clock. Therefore, the frequency accuracy of the main clock can be set to 1E-13.

The frequency drift rate of a satellite clock can be approximated by

$$1ms/2\,(100d)^2 = 2.7E-17/s = 2.3E-12/d \tag{39}$$

In order to ensure the reliability of practical applications, it is necessary to reserve a certain margin and take an index value of one order of magnitude higher; that is, the frequency drift rate of the satellite clock is about 2E-13. After each comparison, the traceability of its accuracy needs to be increased by one order of magnitude. The frequency drift rate of the main clock is one order of magnitude higher than the frequency drift rate of the satellite clock. Therefore, the frequency drift rate of the main clock can be taken as 2E-14.

5. BeiDou Satellite Navigation System Measurement Error Index Quantitative Verification

Through the analysis of the magnitude and value relationship of the measurement error indicators of the BeiDou satellite navigation system, the magnitude of measurement error indicators of the BeiDou satellite navigation system in the ground control mode can be summed up. Comparing with the BeiDou satellite navigation system measurement error indicators currently completed can verify the correctness of the method, as shown in Table 3.

The accuracy of clock bias forecasting is an important indicator that affects the measurement accuracy of BeiDou satellite navigation system. It uses the indicator system under ground control mode as input, quantifies the satellite clock

Table 3: ERE index allocation table.

BeiDou Satellite Navigation System Measurement Error Index			Equivalent distance error (m) quantified according to the model	Completion of BeiDou Satellite Navigation System Measurement Error Index (m)
positioning accuracy			10m	10m
System position accuracy factor PDOP			2.5	2.5
UERE			4	4
Satellite delay error			0.06	0.06 (Qualitative)
Relativity effect error			—	—
Satellite ephemeris error			1	1
Ionospheric refractive error			3	3
Tropospheric refractive error			1	1
Multipath or occlusion error			2	2
BeiDou satellite navigation receiver measurement error			0.3	0.3
Satellite clock difference			0.952	0.9
	Spaceborne atomic clock performance indicators	Frequency stability	Second stability 5E-12, Ten-second stability 1.1E-12, 100-second stability 3.5E-14, Thousand-second stability 1.1E-14, Million-second stability 3.5E-14, Day stability 2E-14	Second stability 3E-12, Ten-second stability 1E-12,100-second stability 3E-13, Thousand-second stability 1E-13, Million-second stability 3E-14, Day stability 2E-14
		Frequency accuracy	1E-11	1E-11
		Frequency drift rate	2E-13/d	1E-13
	Inject station clock performance index	Frequency stability	Second stability 5E-12, Ten-second stability 1.1E-12, 100-second stability 3.5E-14, Thousand-second stability 1.1E-14, Million-second stability 3.5E-14, Day stability 2E-14	Second stability 3E-12, Ten-second stability 1E-12, 100-second stability 3E-13, Thousand-second stability 1E-13, Million-second stability 3E-14, Day stability 2E-14
		Frequency accuracy	1E-11	1E-11
		Frequency drift rate	2E-13	1E-13
	Main clock performance index	Frequency stability	Second stability 1E-12, Ten-second stability 2.2E-13,100-second stability 1E-13, Thousand-second stability 3E-14, Million-second stability 1E-14, Day stability 3E-15	Million-second stability: 1E-14, Day stability: 3E-15
		Frequency accuracy	1E-13	3E-14
		Frequency drift rate	2E-13/d	1E-14/d
	Star time comparison	Pseudorange measurement accuracy	0.7ns	0.5~1ns
		Device delay error	0.7ns	0.5~1ns
		Ionospheric delay error	0.7ns	0.5~1ns
		Multipath error	0.7ns	0.3~1ns

error indicator, and compares with the indicators of measurement error of the Compass satellite navigation system completed in the construction. The arguments deduced that the main clock, the satellite clock, and the injection clock are in the seconds, ten seconds, and the hundred-second stability index which is within the accuracy of the system's true value. More than 1000 seconds is away from the accuracy range of the system, which is consistent with the hopping characteristics of the chime clock after thousands of seconds. The accuracy of the ground pseudorange measurement, the on-board pseudorange measurement accuracy, the equipment delay error, and the ionospheric delay error are 0.7ns in accordance with the error allocation principle. Among them, the measurement accuracy of ground pseudorange and the precision of on-board pseudorange measurement are related to the ranging code used by the navigation signal. Therefore, 0.7 ns represents the orientation of the scope. Within the scope of the real value, prove the validity of the method. Equipment time delay error of the emitting and receiving equipment include time delay and instability of satellite repeater delay etc. 0.7ns is within the true value range, the verification method is correct, and the ionospheric delay error can be further reduced to 0.5ns by using multistation and dual-frequency monitoring observations, indicating that 0.7ns is within the verification range and the method is effective. The multipath error is related to the position and environment of the receiver, and the quantified value is within the true value of the system, which has a typical significance.

6. Conclusion

During the demonstration process of the BeiDou satellite navigation system, it is necessary to repeatedly adjust the errors and simulate the calculation system positioning accuracy according to the system's service accuracy requirement; finally, the indicator distributions that meet the current technological development level under the condition of satisfying the accuracy index are given, and the error control of each link in the engineering development process is guided. In this paper, based on the pseudorange measurement error in BeiDou satellite navigation system, an overall construction method of measurement error system based on empirical prediction method and quantitative decomposition modeling is proposed. A clear representation of the complex relationship of index matching was achieved and qualitatively combined with the numerical matching relationship and constraint relationship between the index values of the indicators related to system service performance and the measurement error index system of the BeiDou satellite navigation system which was established. From the verification results, this method is feasible and can be used to guide the error control in the system engineering construction.

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 work was supported by National Natural Science Foundation of China Major Research Project under Award 91538109 and the National Nature Science Foundation of China under Award 61203226.

References

[1] P. Misra and P. Enge, *Global Positioning Systems:Signals, Measurements, and Performance*, Ganga-Jamuna Press, 2001.

[2] K. Xiong, P. Fan, C. Zhang, and K. B. Letaief, "Wireless information and energy transfer for two-hop non-regenerative MIMO-OFDM relay networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 8, pp. 1595–1611, 2015.

[3] X. Di, K. Xiong, P. Y. Fan, and H. C. Yang, "Simultaneous wireless information and power transfer in cooperative relay networks with rateless codes," *Wireless Communications and Mobile Computing*, vol. 66, no. 4, pp. 2981–2996, 2017.

[4] K. Xiong, C. Chen, G. Qu, P. Fan, and K. B. Letaief, "Group cooperation with optimal resource allocation in wireless powered communication networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3840–3853, 2017.

[5] K. Xiong, B. Wang, and K. J. R. Liu, "Rate-energy region of SWIPT for MIMO broadcasting under nonlinear energy harvesting model," *IEEE Communications Letters*, vol. 16, no. 8, pp. 5147–5161, 2017.

[6] F. Lin, Y. Zhou, X. An, I. You, and K. R. Choo, "Fair resource allocation in an intrusion-detection system for edge computing: ensuring the security of internet of things devices," *IEEE Consumer Electronics Magazine*, vol. 7, no. 6, pp. 45–50, 2018.

[7] X. S. An, X. W. Zhou, X. Lü, F. H. Lin, and L. Yang, "Sample selected extreme learning machine based intrusion detection in fog computing and MEC," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 7472095, 10 pages, 2018.

[8] J. Su, F. Lin, X. Zhou, and X. Lu, "Steiner tree based optimal resource caching scheme in fog computing," *China Communications*, vol. 12, no. 8, Article ID 7224698, pp. 161–168, 2015.

[9] P. Héroux and J. Kouba, "GPS precise point positioning using IGS orbit products," *Physics and Chemistry of the Earth, Part A: Solid Earth and Geodesy*, vol. 26, no. 6-8, pp. 573–578, 2001.

[10] W. Chen and S. Gao, "Effects of inonspheric scintillations on GPS observations in low latitude area," in *Proceedings of the Int Symposium on GPS/GNSS*, pp. 339–346, Tokyo, Japan, 2003.

[11] J. Kouba and P. Héroux, "Precise point positioning using IGS orbit and clock products," *GPS Solutions*, vol. 5, no. 2, pp. 12–28, 2001.

[12] T. H. Yi, H. N. Li, and M. Gu, "Recent research and applications of GPS-based monitoring technology for high-rise structures," *Structural Control and Health Monitoring*, vol. 20, no. 5, pp. 649–670, 2013.

[13] P. Xu, C. Shi, R. Fang et al., "High-rate precise point positioning (PPP) to measure seismic wave motions: an experimental comparison of GPS PPP with inertial measurement units," *Journal of Geodesy*, vol. 87, no. 4, pp. 361–372, 2013.

[14] Y. Wu and J. Guo, "Single point positioning with sequential least-squares filter and estimated real-time stochastic model," *Geo-Spatial Information Science*, vol. 11, no. 1, pp. 13–16, 2008.

[15] A. P. Psimoulis and S. C. A. Stiros, "Supervised learning computer-based algorithm to derive the amplitude of oscillations of structures using noisy gps and robotic theodolites (RTS)records," *Computers Structures*, vol. 92/93, pp. 337–348, 2012.

[16] F. Moschas and S. Stiros, "PLL bandwidth and noise in 100 Hz GPS measurements," *GPS Solutions*, vol. 19, no. 2, pp. 173–185, 2015.

[17] D. Loverro, "Global Positional System Modernization," ION National Technical Meeting, Long Beach, Calif, USA, January 2001.

[18] Z. Jianjun, X. Jun, and X. Ming, "Research on space non cooperative target of relative navigation system based on GNSS reflected signal bistatic radar," in *Proceedings of the 14th International Space Conference of Pacific-basin Societies (ISCOPS '14)*, 2014.

[19] T. Melgard, D. E. Vigene, K. Jong et al., "Pulling in all signals PPP with GPS and GLONASS: the new G2," *GPS World*, vol. 21, no. 3, p. 28, 2010.

[20] P. Li and X. Zhang, "Integrating GPS and GLONASS to accelerate convergence and initialization times of precise point positioning," *GPS Solutions*, vol. 18, no. 3, pp. 461–471, 2014.

[21] J. Ray, M. Cannon, and P. Fenton, "GPS code and carrier multipath mitigation using a multiantenna system," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 1, pp. 183–186, 2001.

[22] D. T. Cox, K. W. Shallberg, and A. Manz, "Definition and analysis of waas receiver multipath error envelopes," *Navigation*, vol. 46, no. 4, pp. 271–282, 1999.

[23] C. Cai and Y. Gao, "Modeling and assessment of combined GPS/GLONASS precise point positioning," *GPS Solutions*, vol. 17, no. 2, pp. 223–236, 2013.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 2643653, 10 pages
https://doi.org/10.1155/2018/2643653

Research Article

A Smart Collaborative Policy for Mobile Fog Computing in Rural Vitalization

Yutong Zhou,[1] Wei Shi,[1] and Fei Song [2,3]

[1]*Institute of Education and Economy Research, University of International Business and Economics, China*
[2]*School of Electronic and Information Engineering, Beijing Jiaotong University, China*
[3]*Faculty of Arts and Sciences, China University of Petroleum-Beijing, Karamay, China*

Correspondence should be addressed to Fei Song; fsong@bjtu.edu.cn

Received 5 May 2018; Revised 17 September 2018; Accepted 24 September 2018; Published 5 November 2018

Guest Editor: Fuhong Lin

Copyright © 2018 Yutong Zhou 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.

Mobile Fog Computing (MFC), as a crucial supplement to cloud computing, has its own special traits in many aspects. As smart mobile devices grow and vary in shapes and formats over the years, the need for real-time interactions and an easy-to-use network is imminent. In this paper, we propose a smart collaborative policy for MFC scenarios by considering the target of rural vitalization. The challenges and drawbacks of extending cloud to fog are reviewed at the beginning. Then, the analysis of policy design is presented from the perspectives of feature comparisons, urgent requirements, and possible solutions. The details of policy establishment are introduced with necessary examples. Finally, performance evaluations are provided based on simulation platforms. Validation results related to round trip time and transmission time illustrate the significant improvements of our proposal in certain ways compared to the original candidate, which enables larger deployment in impoverished areas.

1. Introduction

Mobile Fog Computing (MFC) [1], with outstanding performance and specialties in certain aspects of network deployment [2–4], is gradually attracting attentions to solve the problems of practical usage benefitting impoverished regions. For countries such as China, with large gaps of development between urban and rural areas, there is an urgent need for a solution that could handle issues such as limited infrastructural facilities [5–7], tight distribution of current adjacent resources [8–10], and connecting remote districts to the world outside [11–13]. As rural vitalization process is on the way in many places, a comprehensive architecture utilizing modern technology could bring new development opportunities. In this paper, we propose a smart collaborative policy involving mobile fog computing to fully leverage local advantages and avoid possible troubles in operation.

Rural vitalization strategy, which is believed to be one of the Chinese government's next major steps, is a key factor to determine the future direction of the nation. Due to the differences between people's living standards, educational levels, etc., efficient solutions need to be adopted to bring more evolutional opportunities to these less developed regions, including medical, educational, technological, and employment improvement. In China, rural areas are often associated with poverty, distance, and insufficient infrastructural facilities, along with conservative mindset. As part of the plan, the government hopes to adopt an approach which could create a user friendly network that covers remote impoverished districts and coordinates with the progress to achieve functions. Hopefully, it can enable local rural citizens to receive online education [14] and online medical care [15], together with creating an authoritative platform to introduce more Internet-based enterprises to benefit people by providing employment opportunities [16] and advanced technology [17], as well as supplying fresh and high quality products.

Motivated by the facts stated above, we aim to find a suitable solution utilizing mobile fog computing to assist the rural vitalization process, together with implementation specifically planed for impoverished regions considering their special needs. The contributions of this paper are as follows: A smart collaborative policy for mobile fog computing is

proposed and analyzed based on the rural vitalization background. A simulation is executed and the validation results illustrate the practicability of our proposal.

The structure of this paper is as follows: Section 2 presents related work from four different perspectives. Both the advantages and disadvantages are carefully demonstrated. Section 3 introduces generic ideas during the policy design processes. Section 4 discusses the details of policy establishments. Necessary examples and evidences are also provided. Section 5 focuses on performance comparisons. Relevant experiments, scenarios, and cases are properly imported and analyzed. Section 6 concludes the whole paper and points out the future work.

2. Related Work

There are already several surveys and review works done on Mobile Edge Computing (MEC) [18–20] and IoT [21–23] fields. Here, we mainly focus on the latest and typical progress in four specific areas.

For security improvement, Shirazi et al. [24] identified the need for establishing a secure and resilient extension for current cloud network. The authors made a comprehensive comparison between fog and MEC network and analyzed relevant requirements. Both the classic features and implementation methods were discussed to investigate the capacity of the system. The deployment issues were also presented to highlight the availability of schemes. Rathore et al. [25] focused on the contradictions between dynamical changes of security services and efficient support of hosts' needs. The selection process can be modeled by multicriteria decision making. Hesitant fuzzy soft set was utilized to ensure lower or upper approximation operators. A practical case was introduced during the validation procedures and the results were provided via tabular form. Bierzynski et al. [26] emphasized the importance of combining cloud, fog, and edge computing. They proposed four different schemes to allocate the workload to proper components. Moreover, this paper also analyzed potential issues in the utilization process, including transparent gateways, end-to-end encryption, and hardware security. Dang et al. [27] advocated the merits of using cloud and fog together and discussed the difficulty of data protection in mobile circumstances. A novel model named Region Based Trust Aware was proposed to guarantee dependable translations. An access control scheme was presented for fog nodes as well. Both the applicability and efficiency were verified based on the implemented results.

For mobility enhancement, Puliafito et al. [28] gave a brief description of fog computing evolution and explained the relationship between cloud and edge. Since it is hard to ensure the original transmission connection in mobile fog scenarios, the authors introduced three representative situations: citizen's healthcare, drones for smart urban surveillance, and tourists as time travelers. This paper also analyzed the relevant problems to illustrate mobility support between fog and IoT. Allam et al. [29] focused on the combination issues of powerful cloud and mobile terminations and insisted that available resources should be provided to end users in a mobile cloud computing environment. Many crucial questions, such as limited computational capacity and battery life, connectivity, data security and privacy, latency, and heterogeneity were presented and corresponding solutions were provided. Tang et al. [30] worried that the existing fog computing could not handle the mobility when a large number of users and plenty of applications were involved. Therefore, the authors proposed an intriguing container migration algorithm to reduce the cost of computational power and communication latency. Markov Decision Process spaces were adopted to establish a container migration model. The benefits of implementation were quantitatively demonstrated based on a prototype system. Zhang et al. [31] extended the mobility investigations from IoT to Internet of Vehicles and declared the shortcomings of the current solutions. By referencing the requirements of smart city, the authors took an initiative to alleviate data burdens of traffic and proposed a regional cooperative fog computing architecture. Service types, such as data obtained in multiple sources, distributed computation, and transmission in multiple paths, are discussed in depth. Both the intra-fog and inter-fog resource administration were analyzed.

For application supporting, Bilal et al. [32] highlighted the significance of video services by introducing the situations of its bandwidth utilization and routine usage. Since the latest demands of interactive gaming are strict, this paper mainly studied the efficient schemes for reducing transmission delay and other resource consumption. Liu et al. [33] reviewed the requests offloading from fog and cloud aspects. The authors insisted that the performance could be improved if energy harvesting and social network are taken into account. To decrease the execution cost of social groups, game theory was leveraged to schedule the computation tasks. Multiple queuing models were adopted to describe the latency and energy usage. Hakiri et al. [34] not only indicated the necessity of exploring wireless mesh networks, but also declared the obstacles of hop-based routing protocols. The authors selected Software Defined Networking (SDN) to extend the visibility of management in wireless fog environment. The evaluations illustrated the results of load balancing, delay decreasing, and other capacities. Tinini [35] emphasized that optical networks could jointly operate with fog computing and network function virtualization to support data exchanging from energy perspectives. An integer linear programming model was built to design an optimal scheme for processing. The comparisons with other candidates showed that the power consumption could be cut down.

For platform establishment, Alonso-Monsalve et al. [36] made an attempt to utilize storage and computing resources to avoid bandwidth saturation by reassigning workload. The proposal was described in video-downloading and video-filtering scenarios with volunteers donating part of the buffer space in their personal smart devices. Multiple experiments were executed and simulation results displayed the performance of the new scheme, in terms of network load, servers load, and throughput. Roca et al. [37] introduced a triple-layer architecture of fog computing and proposed a new orchestration scheme to enhance the interoperability. Both the nodes constellations and Fog Function Virtualization (FFV) were implemented to establish a scalable and pervasive platform. Specific approaches of new services deployment

were also presented based on detailed examples. Ali et al. [38] pointed out that, although cloud computing is a promising paradigm, drawbacks are still obvious, such as latency in real-time video streaming, mobility support, and location identification. To minimize the delays, the authors put forward an optimization solution by selecting joint cloudlets within fog scenarios. The relevant limitations in workload were properly designed and validations illustrated the feasibility of the idea. Verma et al. [39] performed a research for patient health remote monitoring through the smart gateway. Several practical services, including embedded data mining and distributed storage, were explained and supported. Totally 67 patients whose homes were equipped with IoT facilities were carefully observed for 30 days. The response latency and accuracy of Bayesian belief network classifier-based model were validated based on comparisons with other baseline algorithms.

3. The Analysis of Policy Design

In terms of educational needs in rural areas, lacking of good faculty resource is a major concern. Putting together the best possible teaching staffs by utilizing the mobile fog computing approach would enable students to receive real-time visual lessons. Moreover, such method could also provide them with the opportunity to learn from default lessons online, especially for schools with minimal amount of students (due to geographical and historical reasons). In this way, they could exchange ideas, raise questions through a shared system, and interact with peers without latency.

When it comes to medical conditions in certain rural areas, well trained medical staffs are in high demand. However, due to budget concerns of local government as well as living standards in such regions, it is difficult for medical units to recruit enough staffs with sufficient skills. There are still critical vacancies need to be filled. Mobile fog computing, on the other hand, is able to integrate local medical resources and provide patients with decent medical treatments. It is also possible to provide online treatment through patients' personal devices under supervision. By adopting this approach, the gaps of medical level between various areas will not be so obvious. At the moment, it is difficult to send well-trained medical staffs to impoverished districts to give local medical staffs guidance and to treat local patients. However, with the completion of a well-designed mobile fog computing platform, it is likely that medical staffs in the entire area could learn from more skillful ones without latency, reducing physical distances and travel obstacles.

Currently, the government is attempting to introduce Internet related enterprises to enter the rural market, aiming to bring about advanced technology and employment opportunities, along with a variety of goods that were previously distant from the residents' daily lives in impoverished regions. Nevertheless, some of these residencies are located in rocky mountain areas isolated from the world outside, with poor signal coverage, extreme weather, and traffic conditions. Considering such natural disadvantages, it is crucial that the logistics approaches adopted in such areas can be supervised through real-time equipment to ensure the safety of couriers and to know precise locations during delivery procedures.

3.1. Feature Comparisons. Based on the structure of mobile fog computing platform, growing combinations to bring fundamental changes to rural vitalization are expected. When it comes to designing a network that meets the need of local people, comprehensive factors should be considered. Cloud computing, already a successful and widely utilized solution, had been existing for more than 10 years to date. It does have its own merits, such as high coverage and large storage space, but with the ever-growing demand to achieve specific approaches, there are still some challenges:

(i) Setup costs: Compared to mobile fog computing, it is expensive (including longer deployment periods and higher prices) for the cloud computing systems to establish available connections. These costs are significant in most cases.

(ii) Real-time response: The time it takes to upload and download within cloud network largely depends on the distance and intermediate facilities, as well as the devices in use. Such a situation makes video streaming and interactive gaming unstable.

(iii) Localized features: Normally, the current mobile fog network is based on locations, and many of the rural regions are isolated or even located in mountain areas. A network with geographic traits would encourage local people to connect with each other achieving regional collaborations.

Mobile fog computing, as an extension and successor to the existing cloud network, makes the connection of various portable devices, such as laptops, tablet PCs, and smart phones a lot easier, and adding new devices into existing networks is also relatively convenient. Compared with cloud computing, fog computing focuses on the edges of calculation resources, which reduced severe risks that large data processing centers face such as malicious attacks and distributed denial of services.

(i) Data transmission speed: As fog network is established within a specific area, the transferring speed among participant devices is faster than that of cloud network due to short communication distance.

(ii) Sharing of storage capacity: MFC allows users to store their data on nearby devices safely, which means the buffer capacity of each candidate can be greatly extended if availability is well guaranteed.

(iii) Cost-effective and resource friendly: The cost to build a fog network is considerably less compared to a cloud network. Besides, as not all the information is being processed through the same router nor at the same time, bandwidth utilization will be more flexible and can be reserved for specific needs.

These specialties determine that fog computing is a more suitable approach for smaller scale coverage, budget concerned, and real-time stream required circumstances. In many cases, the effective usage of fog computing would enable rural areas to develop at a higher speed with lower cost. Meanwhile, it can also be a protection for companies or the

government to carry out experimental actions. For example, if an Internet company aims to provide its services to a remote impoverished region, considering its geographic distance and natural environment, it is unsure whether the investment will be effective. By adopting a cloud computing network, it will first need to prepare for the entire connection procedure, which takes more time and more resources. If the project cease to continue in the future, the primary input will then be significantly huge compared to adopting a fog computing method which is more on the easy-to-use and easy-to-quit side.

3.2. Urgent Requirements. Big cities have to establish highly reliable wireless systems. From the infrastructural perspective, massive redundant base stations are built to guarantee basic telecom signal coverage. Both sporadic and intensive access points of WiFi are strong supplements for Internet signal coverage. Huge populations and diverse applications had continuously added enormous burdens and challenges (such as low spectrum utilization, regular transmission congestion) for all kinds of access modes. Big data generated from each end host stimulate redundant constructions of base stations and access points. Such circulations had been witnessed for many years. The designer, implementer, and administrators are attempting to enhance the efficiency and reduce the cost since the beginning. From the service perspective, encryption is a kind of frequently used approach to achieve dependable transmission. Many complicated algorithms had been proposed in different layers to improve security, which enable the possible utilization in multipath scenarios. A simple understanding is that distributing the data packets on multiple available paths would definitely increase the difficulty of sampling. More importantly, the transmission would not be interrupted when one path fails since data packets could be sent on other paths and the service could be maintained. This evidence shows that more paths may bring strong dependability.

Based on the population mobility features, information point distribution patterns, online business content, user sensitive data, etc., it makes sense to provide full coverage, high bandwidth, excessive links, and smart mechanisms to meet various users' demands in metropolises. However, for impoverished regions, the urgent requirements are quite different.

Firstly, the mobility of population is relatively low and the distribution of impoverished people is unbalanced. Without modern transport vehicles and convenient highways, it is difficult for the natives to extend the range of routine activities. In such circumstances, the capacity of one base station might be more than enough for a small village. It is inefficient and uneconomical to establish individual coverage just for sparsely populated places. A better way is to cover larger areas with more base stations. However, the reasonable deployment patterns need to be investigated carefully.

Secondly, more and more data collection points (for plant monitoring, animal shepherding, etc.) and dissemination points (for agricultural knowledge training, culture courses teaching, etc.) are urgent for natives to connect to the Internet. In order to help people to shake off the burden of poverty, bidirectional communication with high quality and high dependability is necessary. As the strong supporters, various candidates are qualified in this direction.

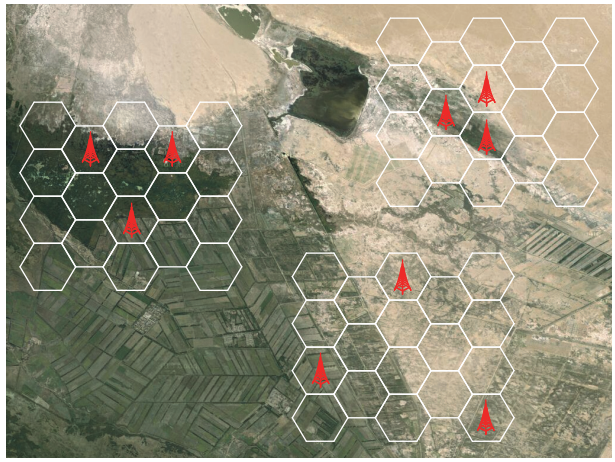


Figure 1: The demonstration for coverage mode.

Thirdly, the environment and condition of deploying networks in impoverished regions are harsh and complicated. Building wired connections, sometimes, is extremely hard and may cost triple or quadruple capital expenditures comparing with establishing them in cities. Therefore, more wireless connections should be considered and adopted. Some hybrid solutions are also preferable if the construction condition is allowable.

3.3. Possible Solutions. The reasonable responses for previous requirements can be generated from different points of view based on novel mechanisms, emerging technologies, and up to date equipment.

Firstly, optimizing strategy and sufficient planning should be made by considering overall situation comprehensively. For instance, traditional impact factors, such as execution difficulties, people distributions, and signal coverage, should be used to determine the construction modes of the base stations and the access points. Taking a large-scale case as an example, three kinds of patterns, i.e., uniform, intensive, and dispersive, are demonstrated in Figure 1. One satellite picture is selected to schematically show the location of native people. The base stations marked with red color can also be replaced by access points in reality. Covered areas are enclosed with multiple hexagons. These fundamental infrastructures also provide support for the following new policy implementation.

Secondly, multiple paths created by various access modes (2.5G, 3G, 4G, etc.) can be synergistically utilized to improve reliability, enhance security, reduce latency, and increase bandwidth. More specifically, as the aggregation node, it may contain several Subscriber Identity Module (SIM) cards and one standard WiFi interface simultaneously to achieve flexible networking. Native users could easily connect to it via electronic devices to communicate with remote experts. The intermediate routers are also available to provide encryption and decryption. Such reliable transmission in a mobile multipath environment is illustrated in Figure 2.

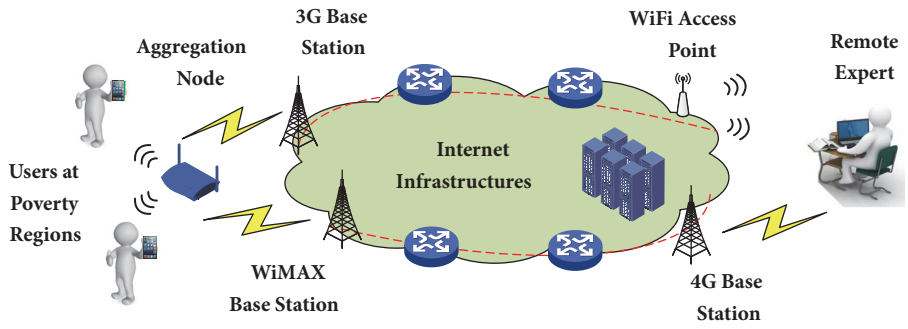


Figure 2: The transmission example of new policy.

Thirdly, wireless LTE-enable microcell and picocell can be implemented to cooperate with base stations. Due to the advantages in size, weight, and cost, such equipment can be easily adopted in complicated and severe environments. In order to leverage the high performance of wired connections, the combination scheme between LTE and Ethernet is also convenient to achieve. Industrial class products (CPU, memory, hard disks, etc.) are preferred to guarantee the reliability.

4. The Details of Policy Establishment

There are plenty of reasons to adopt one way delay instead of round trip time (RTT) in designing a smart collaborative policy. A high priority packet arriving at the destination with a faster speed is recommended. The selected path with minimum forward or backward delay will be the most suitable candidate.

Due to the absence of mandatory clock synchronization in current Internet or other computer networks, the absolute value of one way delay is extremely hard to obtain, especially in some large-scale scenarios. The existing random errors in the clock calibration procedure may severely confuse the measurement results. By witnessing these facts, we proposed Relative Delay Estimator (RDE) in the previous work. The idea is to calculate relative differences rather than the absolute values. The implementation scheme simply added some new chunks to the protocol stack. However, we argue that the efficiency can be further enhanced via novel approach.

The basic clue is to change the format of current Stream Control Transmission Protocol (SCTP) chunks (i.e., data chunk and acknowledgment chunk) and encapsulate relevant information used by RDE into packets. Three time variables are applied to calculate relative delay for each path. By updating these variables continually during packet interaction procedures, the sender will be clearly aware of the changes in one way delay.

4.1. Timestamp Usage. If the sending (receiving) timestamp at the sender-side is stored when the data packet (acknowledgment sent by receiver) is being sent (received), only adding the timestamp of receiver-side into the acknowledgment is sufficient for RDE calculation. The basic assumption is that the receiver would respond as soon as it obtains the data packet (i.e., there is no latency between the data packet receiving and acknowledgment sending at the receiver-side). In order to prove the rationality and feasibility, we review the functions of conventional timestamp in Transmission Control Protocol (TCP).

Firstly, the length of sequence number field in TCP packet header is 32 bits, which seriously restricts the maximum value of corresponding volume, i.e., 2^{32}. In high bandwidth circumstances, the sequence number might wrap within a short period if there are huge data packets need to be transferred. That means it is possible to see two data packets with the same sequence number in the same network. The TCP timestamp can be used to avoid this awkward situation. As an extension to the sequence number space, it will assist the sender and receiver to differentiate these data packets and make the right decision. Such scheme is called "Protection Against Wrapped Sequence (PAWS) Numbers".

Secondly, in some implementation processes, TCP only measures the RTT once in each data sending window. The retransmission timer will be set when the data packet is sent, and the RTT would be calculated when the acknowledgment is received by the sender. However, the value of such calculation cannot accurately reflect the variety of RTT very well. Therefore, the timestamp can be added to each packet header to obtain a more precise result without increasing more storage burdens at the sender-side. Whenever an acknowledgment arrives, the sender can compute directly by using sending timestamp (inside the packet header) and receiving timestamp (recorded by the local timer). If the delay acknowledgment is enabled, the corresponding adjustments will be needed. For example, the specific delay period could be pointed out by the receiver and piggybacked with acknowledgment. When the sender obtains such information, the value of RTT could be easily achieved.

Based on above discussions, there are two main reasons why TCP contains sending timestamp in the packet header. The first one is for protecting the PAWS. The second one is to accurately calculate RTT and reduce the burden of the sender.

However, we argue that these two reasons may not hold true in SCTP anymore. Although there are still 32 bits in sequence number field for both SCTP and Concurrent Multipath Transfer (CMT) SCTP cases, it is not easy to see the

packet with wrapped sequence numbers even in high bandwidth networks, because each Transmission Sequence Number (TSN) is assigned to a data chunk (in TCP case, TSN is assigned to each byte). The second reason is true only when the resource of user's equipment is quite limited. With the evolution of hardware design, the performance of terminals has been greatly improved after TCP timestamp was proposed. Since the data packets sent to network can be uniquely identified with TSN, one can store the sending timestamp for RTT calculation at the sender-side without consuming too much storage resource. Compared with the burden introduced by recording sending timestamp, filling more messages into each data packet may be more important in many scenarios.

As a brief summary, for the high performance end host which aims to implement new scheme and send more messages in each packet, it makes sense to record the sending timestamp at the sender-side. For the end host which really cares about the storage resource, they could implement new schemes by adding sending timestamp into packet header.

4.2. Protocol Simplification. Apart from whether it is suitable to keep the sending timestamp at the sender-side, there are still some difficulties during the protocol simplification.

The first one is whether to let the receiver know about the result of RDE (i.e., the backward path with minimum delay). If the answer is "Yes", some changes in both the data packet format and process procedures are needed. Since there is no RDE state recorder inside the receiver, a possible solution for the sender is to contain the path identifier in retransmitted packets and guide the receiver to transmit the Selective ACKnowledgment (SACK) via a corresponding path. However, it may consume precious space of the data packet header. The extra process for checking the notification provided by the sender has to be designed at the receiver-side. If the answer is "No", the original data packet format could be maintained. In most current multipath transport protocols, the data packets sent on one path could also be acknowledged by the SACKs sent on other paths. Such mechanism enables the sender to update relevant calculation variables quickly. Nevertheless, the disadvantage is that the sender may wait for a bit longer time to realize the receiving situation of previous retransmitted packets. Here, we choose the second scheme to minimize modifications at both sides.

The second one is on how to find the data chunk that triggered the SACK chunk inside the send buffer. In order to better prepare the retransmission brought by packet loss, all the data chunks sent on different paths will be stored inside the send buffer. Our solution is to use a chain table to link each data chuck and the sending timestamp. When the sender receives a new SACK chunk, it should look up the data chunk inside the send buffer to search for the relevant sending timestamp. We suggest recording the sending timestamp in the original structure of the data chunk. Due to the rules that one SACK chunk may acknowledge more than one data chunk, such as delay SACK or SACK missing, the sender should treat the data chunk with largest sending timestamp as the "trigger data chunk" and use the sending timestamp of this data chunk to update corresponding calculation variables.

The third one is on how to modify the format of SACK chunk. There is no doubt that the receiver's timestamp is needed when the sender wants to update relevant information. According to the requirement of RDE, we add 4 bytes receiver's timestamp into each acknowledgment. The difference between the original and the new format of SACK chunk is shown in Figure 3. Necessary actions on both sides will be given in the following. Based on the Karn algorithm, the SACK of retransmitted packets should not be used in calculating RTT. That is because both original and retransmitted packet may trigger such SACK. The situation is the same during the RDE calculation process. For distinguishing these SACKs from normal ones at the sender-side, the bits inside "Chunk Flags" could be fully utilized. For instances, something like "FLAG RTX" can be used to indicate the SACKs which should be ignored. At the receiver-side, it is not necessary to contain the receiver's timestamp anymore for these SACK chucks.

The fourth one is on how to deal with "out-of-order" SACK chunks. In the mechanism of SCTP or CMT-SCTP, a cumulative TSN value will be maintained at the sender-side. When a new SACK chunk with a larger cumulative TSN is received, the sender will update this value and delete the data chunks with smaller TSNs inside the send buffer. If the new SACK chunk has a smaller cumulative TSN, it will be ignored. It is true that out-of-order SACK chunk should not be processed when the sender calculates the congestion window increase value. However, in the new scheme, such mechanism may introduce serious problems. In a common case, assuming "SACK chunk A" was firstly sent on path 1, then "SACK chunk B" was launched on path 2. Their arriving sequence to the destination might be reversed due to the difference of transmission delay. Normally, the cumulative TSN contained in "SACK chunk B" should be larger than that of "SACK chunk A". As a result, the sender will drop "SACK chunk A" thus cannot update calculation variables of the path which has large backward delay. Even though this "SACK chunk A" is processed by the sender when it arrives, the corresponding sending timestamp may not be found because the data chunks with smaller TSNs had been deleted. If most SACK chunks sent on path 1 are handled in the same way, the information related to path 1 cannot be gathered to reflect the current situation. In order to avoid it, the sender should keep the sending timestamp of the data chunk until it is acknowledged by the SACK chunk sent on the same path. More importantly, each SACK chunk should be carefully processed even if it is out-of-order.

4.3. Implementation Procedure. After four-way handshakes, a CMT-SCTP association will be established. Then the calculation variables of RDE for each path should be set to 0 at the sender-side. The normal data chunk (not retransmitted ones) and normal SACK chunk (not for confirming retransmitted ones) should be transferred as ordinary rules, i.e., on the preassigned path based on a scheduling mechanism. RDE calculation will be operating in backstage and the result should be prepared for further usage at any time. When packet drop is pointed out via duplicated SACKs or timeout, the retransmitted data chunk(s) will be sent on the path with minimum

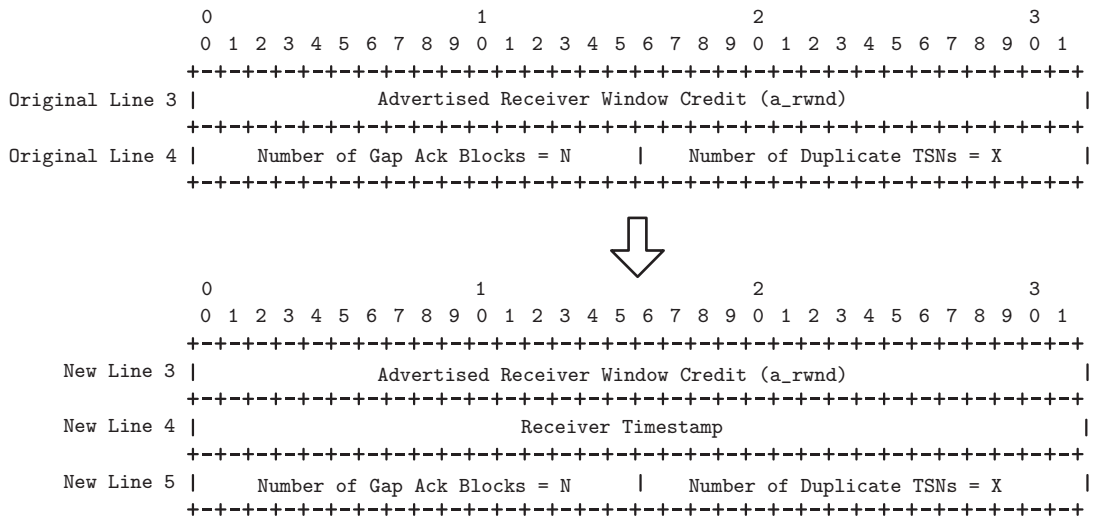


Figure 3: The updating for the format of SACK chunk.

forward delay. Although the corresponding SACK chunk may not be sent on the path with minimum backward delay, the overall performance should not be seriously influenced because retransmitted data chunks could be quickly confirmed by the SACK chunks sent on other paths.

During the discussion process, we only selected a SCTP-based transport protocol as the modified object. However, the described method is workable as well for TCP-based protocols in multiple scenarios.

5. Simulation and Comparisons

To better evaluate the performance of our new policy, comprehensive validations are executed based on NS2 and Matlab. A generic mobile fog computing environment is implemented within a specific area with source, intermediate, and destination nodes. Necessary modifications have been carefully accomplished in source code. The key parameters, such as covered areas, packet volume, and transmission latency, are adjusted according to different requirements.

Two experiments together with two scenarios are established based on moving ability and interval delay, respectively. Case one is to illustrate that a few nodes are randomly deployed and resulted in weak connectivity. Case two is to demonstrate that more nodes are involved and abundant connections have been created.

5.1. Experiment One. The first experiment is focusing on the static situation. During the transmission processes, the participants are able to store and forward the content without changing locations. Two cases are implemented to display the improvements.

More than 1 hop might be triggered from the source to the destination when a single packet was sent. The relationships between RTT and number of hops are illustrated in Figure 4(a). For case one (shown in blue lines), the initial values of two policies are quite close to each other. From 2 to 10 hops, the differences can be clearly observed. The gap value is getting larger until the number of hops reaches 6. The reason might be that most packet sending is finished within 8 hops in such situations. When 9 or 10 hops appear, the gap value becomes larger again. For case two (shown in green lines), similar phenomenon can be found if more nodes join in. Due to expanded distance, the cost of packet forwarding is also increased compared with that of case one. However, for both of them, the curve fluctuations of our policy are always small since it can choose a better route among various candidates.

The transmission time is recorded and analyzed when multiple packets are sent, which is demonstrated in Figure 4(b). From the beginning to the end, all the packets are generated uniformly. For case one (shown in red lines), the gap value is quite small when 500 packets appear. Although advantages of the policy can be easily displayed in the previous figure, it is not obvious when the unit was changed from ms to s. With the increasing of packet number, the gap value arises again and reaches the maximum when 2500 packets are set. For case two (shown in black lines), it is quite interesting to discover that gap value is enlarged when X axis is equal to 2000. Similar variation cannot be found in case one. The reason might be that more routing options are provided when strong connectivity is established. If there is only one path between the source and destination node, the overall latency will be the same no matter which policy is adopted. Such accumulation will not benefit the transmission efficiency if too many "no option" packets are forwarded within the network. Anyhow, both cases have illustrated the superiority of the new policy.

5.2. Experiment Two. The second experiment is focusing on a dynamic situation, which means the involved nodes can change their positions during data exchange. Two cases, similar to previous experiment, are launched to describe the complex processes.

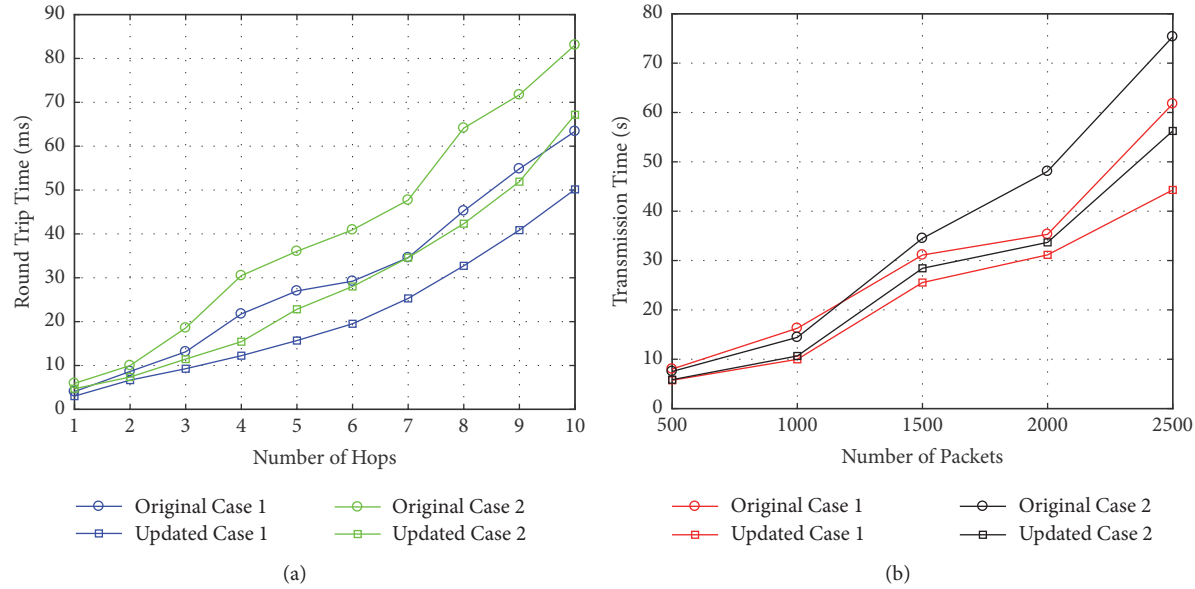


Figure 4: Static nodes with different hops and packets.

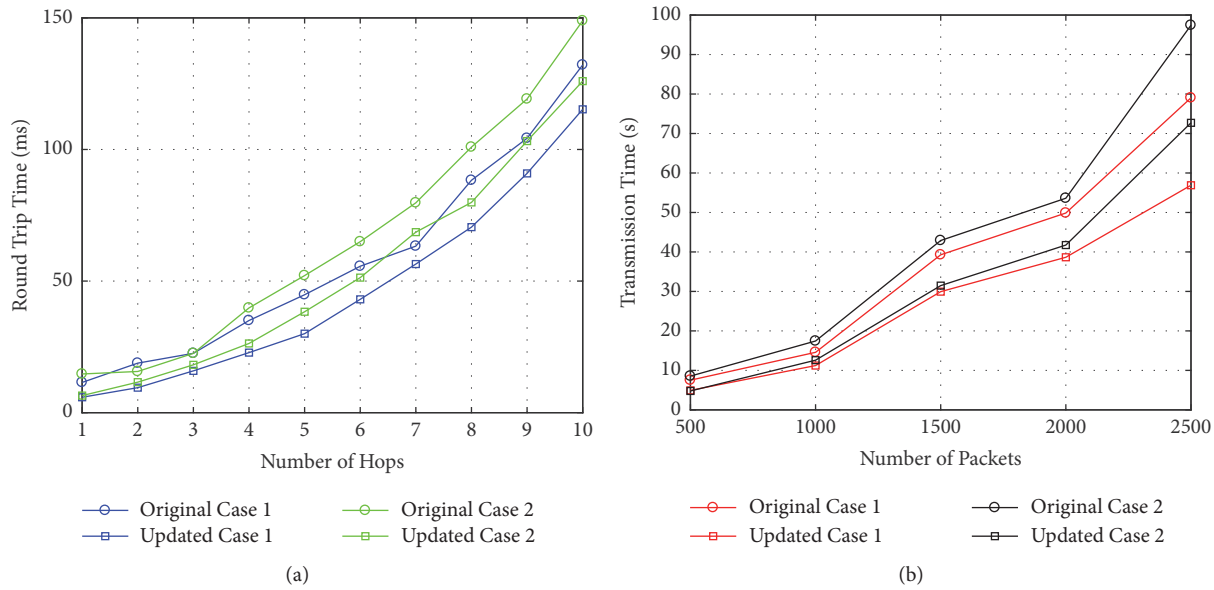


Figure 5: Dynamic nodes with different hops and packets.

In Figure 5(a), the variations of RTT are described based on the hop number increasing. For case one, the curve of original policy (marked with circle) is getting higher with strong fluctuations. The curve of updated policy (marked with square) maintains its stable characteristic. At the beginning, the improvement ratios for 1 and 2 hops are 49.1% and 49.5%. Then, such value is reduced to 29.6% when hop number is equal to 3. The unstable pattern of gap values can be observed as well. The minimum and maximum enhancements are 10.8% and 34.9% when the hop number is from 4 to 10. For case two, the locations of two curves are a bit higher than that of case one. A noteworthy observation is that the fluctuation of original policy is weakened. One possible reason is that strong connectivity in dynamic situation converges the distribution of nodes. Due to position changing, transmissions might be interrupted or stopped, which seriously affects the curve tendency. Therefore, all the values are greater than 90 ms if 9 or 10 hops are selected.

In Figure 5(b), the growth of transmission time is presented based on different packet numbers. For case one, curves of the original policy and updated policy are increased proportionately. The improvement ratios are 34.4%, 23.0%, 23.8%, 22.5%, and 28.0%, respectively. Although more packets have inhibited the fluctuations of the original policy, the final performance is still not acceptable. For case two, such pattern is maintained and even more connections are

involved. About 44.1% (or 22.1%) overall latency can be reduced in the best (or worst) circumstance. When the packet number is equal to 2500, the maximum gap value is 24.7s. From the above results, the influences brought by node mobility are almost negligible if the new policy is employed accurately.

6. Conclusions

Mobile Fog Computing (MFC) has been recognized as a powerful tool for cloud computing extensions. In order to solve practical difficulties during the rural vitalization process, we proposed a smart collaborative policy based on specific demands. Firstly, the significant preliminaries and discussions were given to deal with multiple application categories. Secondly, the timestamp usage, protocol simplification, and implementation procedure were respectively, presented to establish the policy in detail. Thirdly, two experiments were designed according to the node mobility situations. For each of them, two scenarios and two cases were involved to provide comprehensive validations. The advantages of our policy, in terms of round trip time and transmission time, were fully illustrated when different hop and packet numbers were utilized. For instance, in the second experiment, the improvement ratio is 29.6% when hop number is equal to 3.

For future work, the large-scale deployment will be considered and implemented. The enhancements of the smart collaborative policy should be achieved based on the feedback of applications.

Data Availability

The 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

This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant 2017JBM012, in part by Joint Foundation of China University of Petroleum-Beijing at Karamay, and in part by the Project of State Grid Corporation of China under Grant SGRIXTJSFW[2016]377.

References

[1] X. An, X. Zhou, X. Lü, F. Lin, and L. Yang, "Sample Selected Extreme Learning Machine Based Intrusion Detection in Fog Computing and MEC," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 7472095, pp. 1–10, 2018.

[2] F. Song, Z.-Y. Ai, J.-J. Li et al., "Smart collaborative caching for information-centric IoT in fog computing," *Sensors*, vol. 17, no. 11, 2017.

[3] J. Wu, B. Cheng, M. Wang, and J. Chen, "Energy-Efficient Bandwidth Aggregation for Delay-Constrained Video over Heterogeneous Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 1, pp. 30–49, 2017.

[4] R. Gu, S. Zhang, Y. Ji, and Z. Yan, "Network slicing and efficient ONU migration for reliable communications in converged vehicular and fixed access network," *Vehicular Communications*, vol. 11, pp. 57–67, 2018.

[5] Y. Zhang, P. Shi, C.-C. Lim, H. Zhu, J. Hu, and Y. Zeng, "Chaotification of a class of linear switching systems based on a Shilnikov criterion," *Journal of The Franklin Institute*, vol. 354, no. 13, pp. 5519–5536, 2017.

[6] F. Lin, Y. Zhou, X. An, I. You, and K. R. Choo, "Fair Resource Allocation in an Intrusion-Detection System for Edge Computing: Ensuring the Security of Internet of Things Devices," *IEEE Consumer Electronics Magazine*, vol. 7, no. 6, pp. 45–50, 2018.

[7] F. Song, Y. Zhou, Y. Wang, T. Zhao, I. You, and H. Zhang, "Smart collaborative distribution for privacy enhancement in moving target defense," *Information Sciences*, 2018.

[8] Z. Ai, Y. Liu, F. Song, and H. Zhang, "A Smart Collaborative Charging Algorithm for Mobile Power Distribution in 5G Networks," *IEEE Access*, vol. 6, pp. 28668–28679, 2018.

[9] Y. Zhang, X. Liu, H. Zhang, and C. Jia, "Constructing chaotic systems from a class of switching systems," *International Journal of Bifurcation and Chaos*, vol. 28, no. 2, 1850032, 9 pages, 2018.

[10] F. Song, D. Huang, H. Zhou, H. Zhang, and I. You, "An Optimization-Based Scheme for Efficient Virtual Machine Placement," *International Journal of Parallel Programming*, vol. 42, no. 5, pp. 853–872, 2014.

[11] Z. Ai, Y. Zhou, and F. Song, "A Smart Collaborative Routing Protocol for Reliable Data Diffusion in IoT Scenarios," *Sensors*, vol. 18, no. 6, p. 1926, 2018.

[12] J. Wu, B. Cheng, M. Wang, and J. Chen, "Quality-Aware Energy Optimization in Wireless Video Communication with Multipath TCP," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2701–2718, 2017.

[13] F. Song, Y.-T. Zhou, K. Kong, Q. Zheng, I. You, and H.-K. Zhang, "Smart collaborative connection management for identifier-based network," *IEEE Access*, vol. 5, pp. 7936–7949, 2017.

[14] K. D. Rajab, "The Effectiveness and Potential of E-Learning in War Zones: An Empirical Comparison of Face-To-Face and Online Education in Saudi Arabia," *IEEE Access*, vol. 6, pp. 6783–6794, 2018.

[15] W. Guo, J. Shao, R. Lu, Y. Liu, and A. A. Ghorbani, "A Privacy-Preserving Online Medical Prediagnosis Scheme for Cloud Environment," *IEEE Access*, vol. 6, pp. 48946–48957, 2018.

[16] J. Su, F. Lin, X. Zhou, and X. Lu, "Steiner tree based optimal resource caching scheme in fog computing," *China Communications*, vol. 12, no. 8, Article ID 7224698, pp. 161–168, 2015.

[17] R. Gu, S. Zhang, Y. Ji, T. Guo, and X. Wang, "Efficient software-defined passive optical network with network coding," *Photonic Network Communications*, vol. 31, no. 2, pp. 239–250, 2016.

[18] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A Survey on Mobile Edge Computing: The Communication Perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[19] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile Edge Computing: A Survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, 2018.

[20] H. Liu, F. Eldarrat, H. Alqahtani, A. Reznik, X. de Foy, and Y. Zhang, "Mobile Edge Cloud System: Architectures, Challenges, and Approaches," *IEEE Systems Journal*, vol. 12, no. 3, pp. 2495–2508, 2018.

[21] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: a survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[22] M. Chiang and T. Zhang, "Fog and IoT: an overview of research opportunities," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 854–864, 2016.

[23] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1125–1142, 2017.

[24] S. N. Shirazi, A. Gouglidis, A. Farshad, and D. Hutchison, "The extended cloud: Review and analysis of mobile edge computing and fog from a security and resilience perspective," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2586–2595, 2017.

[25] S. Rathore, P. K. Sharma, A. K. Sangaiah, and J. J. Park, "A Hesitant Fuzzy Based Security Approach for Fog and Mobile-Edge Computing," *IEEE Access*, vol. 6, pp. 688–701, 2017.

[26] K. Bierzynski, A. Escobar, and M. Eberl, "Cloud, fog and edge: Cooperation for the future?" in *Proceedings of the 2nd International Conference on Fog and Mobile Edge Computing, FMEC 2017*, pp. 62–67, Spain, May 2017.

[27] T. D. Dang and D. Hoang, "A data protection model for fog computing," in *Proceedings of the 2nd International Conference on Fog and Mobile Edge Computing, FMEC 2017*, pp. 32–38, Spain, May 2017.

[28] C. Puliafito, E. Mingozzi, and G. Anastasi, "Fog Computing for the Internet of Mobile Things: Issues and Challenges," in *Proceedings of the 2017 IEEE International Conference on Smart Computing, (SMARTCOMP '17)*, pp. 1–6, Hong Kong, May 2017.

[29] H. Allam, N. Nassiri, A. Rajan, and J. Ahmad, "A critical overview of latest challenges and solutions of Mobile Cloud Computing," in *Proceedings of the 2nd International Conference on Fog and Mobile Edge Computing, FMEC '17*, pp. 225–229, Valencia, May 2017.

[30] Z. Tang, X. Zhou, F. Zhang, W. Jia, and W. Zhao, "Migration Modeling and Learning Algorithms for Containers in Fog Computing," *IEEE Transactions on Services Computing*, 2018.

[31] W. Zhang, Z. Zhang, and H. Chao, "Cooperative Fog Computing for Dealing with Big Data in the Internet of Vehicles: Architecture and Hierarchical Resource Management," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 60–67, 2017.

[32] K. Bilal and A. Erbad, "Edge computing for interactive media and video streaming," in *Proceedings of the 2nd International Conference on Fog and Mobile Edge Computing, FMEC '17*, pp. 68–73, Spain, May 2017.

[33] L. Liu, Z. Chang, and X. Guo, "Socially-aware Dynamic Computation Offloading Scheme for Fog Computing System with Energy Harvesting Devices," *IEEE Internet of Things Journal*, 2018.

[34] A. Hakiri, B. Sellami, P. Patil, P. Berthou, and A. Gokhale, "Managing Wireless Fog Networks using Software-Defined Networking," in *Proceedings of the IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA '17)*, pp. 1149–1156, Hammamet, October 2017.

[35] R. I. Tinini, L. C. Reis, D. M. Batista, G. B. Figueiredo, M. Tornatore, and B. Mukherjee, "Optimal Placement of Virtualized BBU Processing in Hybrid Cloud-Fog RAN over TWDM-PON," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM '17)*, pp. 1–6, Singapore, December 2017.

[36] S. Alonso-Monsalve, F. Garcia-Carballeira, and A. Calderon, "Fog computing through public-resource computing and storage," in *Proceedings of the 2nd International Conference on Fog and Mobile Edge Computing, FMEC 2017*, pp. 81–87, May 2017.

[37] D. Roca, J. V. Quiroga, M. Valero, and M. Nemirovsky, "Fog Function Virtualization: A flexible solution for IoT applications," in *Second International Conference on Fog and Mobile Edge Computing (FMEC)*, pp. 74–80, Valencia, 2017.

[38] M. Ali, N. Riaz, M. I. Ashraf, S. Qaisar, and M. Naeem, "Joint Cloudlet Selection and Latency Minimization in Fog Networks," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 4055–4063, 2018.

[39] P. Verma and S. K. Sood, "Fog Assisted-IoT Enabled Patient Health Monitoring in Smart Homes," *IEEE Internet of Things Journal*, 2018.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 7606513, 11 pages
https://doi.org/10.1155/2018/7606513

Research Article

Mobile Fog Computing-Assisted Resource Allocation for Two-Hop SWIPT OFDM Networks

Xiaofei Di,[1] **Yu Zhang**,[2,3] **Tong Liu**,[4] **Shaoli Kang**,[5] **and Yue Zhao**[6]

[1] *School of Communication Engineering, Hangzhou Dianzi University, Hangzhou 310018, China*
[2] *State Grid Energy Research Institute Co., Ltd., Beijing 102209, China*
[3] *School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China*
[4] *Beijing Computing Center, Beike Industry Park, Beijing 100094, China*
[5] *China Academy of Telecommunications Technology, Beijing 100191, China*
[6] *Epithelial Systems Biology Laboratory, Systems Biology Center, National Heart Lung and Blood Institute, National Institutes of Health, Bethesda, Maryland, USA*

Correspondence should be addressed to Yu Zhang; zhangyu2@sgeri.sgcc.com.cn

Received 2 May 2018; Accepted 8 July 2018; Published 27 September 2018

Academic Editor: Lei Yang

Copyright © 2018 Xiaofei Di 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 mobile fog computing-assisted resource allocation (RA) is studied for simultaneous wireless information and power transfer (SWIPT) two-hop orthogonal frequency division multiplexing (OFDM) networks, where a decode-and-forward (DF) relay first harvests energy from signals emitted by a source and then helps the source to forward information to its destination by using the harvested energy. Power splitting (PS) strategy is adopted at the relay and a different PS (DPS) receiver architecture is proposed, where the PS factors of all subcarriers are different. A RA problem is formulated to maximize the system's achievable rate by jointly optimizing subcarrier pairing, power allocation, and PS factors. Since the RA problem is a nonconvex problem and is difficult to solve, an efficient RA algorithm is designed. As the wireless channels are fast time-varying, the computation is performed in mobile fog node close to end nodes, instead of remote clouds. Results demonstrate that the achievable rate is significantly increased by using the proposed RA algorithm. It is also found that the computation complexity of RA algorithm of DPS receiver architecture is much lower than the existing identical PS (IPS) receiver architecture, and thus the proposed DPS architecture is more suitable for computation-constrained fog system.

1. Introduction

In the past decade, cloud computing has emerged as a new paradigm. It enables computing, storage, and network managements to centralize in the clouds, which are referred to as data centers, cellular core networks, and so on. With the clouds, vast resources can be provided to resource-constrained devices to satisfy their requirements of computing and storage. However, there is an inherent limitation for cloud computing [1], i.e., the long propagation distance from the end user to the remote cloud center, resulting in very long latency.

Recently, a new trend has been happening; that is, the computing is pushed to the network edge devices due to their progressively enhanced computation capacity. This is called mobile fog computing (MFC) or mobile edge computing (MEC) [2, 3], where the network edge devices perform computing tasks instead of remote clouds. Thanks to closer distance to end users, the latency is less and thus real-time tasks can be achieved via MFC, which is an effective supplement to cloud computing. MFC is applicable to delay sensitive tasks while cloud computing to sophisticated but delay-insensitive data processing work.

One potential application of MFC is in Internet of Things (IoT), such as wireless sensor networks (WSN). In WSN, sensor nodes are responsible for data gathering, and sink nodes for collecting and preprocessing data from surrounding sensor nodes and then delivering data to the remote clouds, which perform further complicated data processing and information mining. For WSN, resource allocation (RA)

is a key approach to improve system performance. It is carried out according to different channel states and deemed to be a real-time task, as the wireless channels are fast time-varying channels and RA needs to be processed rapidly to adapt the dynamic channels. So MFC is a more appropriate option than cloud computing.

In the field of WSN, cooperative relay communication is deemed as an important technique, as it can guarantee that the far sensor nodes can complete communication with each other via intermediate relaying sensor nodes [4, 5]. On the other hand, orthogonal frequency division multiplexing (OFDM) is employed in wireless communication networks [6]. The combination of relay and OFDM is able to significantly enhance the performance of the system [7, 8].

MFC-assisted cooperative relay systems [9, 10] and OFDM systems [11, 12] have attracted much attention and been widely investigated. In [9], a fog-enabled cooperative communication network was considered, where multiple fog nodes were configured to support two-hop transmissions, and the optimal system performance was achieved by designing time reuse patterns. In [10], cooperative fog computing for the Internet of Vehicles (IoV) was studied, where the cooperation of fog nodes was explored to enhance the system performance. In [11], a MFC-assisted multiuser OFDM network was considered, and the total consumed energy of mobile users was minimized by jointly optimizing subcarrier and CPU time allocation. In [12], the joint subcarrier and power allocation problem in an MFC-based OFDM system was investigated to minimize the maximal delay of all devices.

Meanwhile, WSN are usually energy-constrained networks, and connecting sensor nodes to power grid is impossible sometimes. Batteries can be deployed in sensor nodes, but the batteries capacity is limited and may be hard to be replaced frequently. Recently, wireless power transfer has attracted much attention, in which energy-constrained devices can harvest energy by using wireless signals emitted by system nodes with sufficient energy source. Noting that wireless signal can simultaneously carry and transfer information and energy, this is deemed as the simultaneous wireless information and power transfer (SWIPT) [13].

SWIPT has been widely investigated [14–24]. In [14], it was assumed that information decoding (ID) and energy harvesting (EH) are simultaneously carried out by using the received identical signals. Nevertheless, this is deemed not to be realized, and therefore some practical SWIPT receivers were also presented, such as time switching (TS) and power splitting (PS) in [13]. In TS receivers, ID and EH are performed in two different phases, respectively. In PS receivers, the wireless signal is split into two streams: one stream enters into the energy receiver to harvest energy, and the other enters into the information receiver to obtain information. In [16], these SWIPT receiver architectures were applied to cooperative two-hop network, and the achievable rate performance was investigated. These architectures have also been widely studied in OFDM systems; see, e.g., [17–20]. But, the existing work mainly concerns point-to-point OFDM systems. For example, in [17, 19], the performances of throughput and weighted sum-rate were investigated for multiuser OFDM networks. In [20], max-min fair resource allocation was studied for multigroup multicast OFDM systems. In [18], a new SWIPT receiver architecture was proposed, where one part of the subcarriers was used for ID, and the other part was used for EH.

Recently, some work discussed the SWIPT-enabled two-hop OFDM system. In [21, 22], the authors considered amplify-and-forward (AF) relaying protocol and the achievable information rates were maximized for two-hop MIMO-OFDM AF relay system. In [23, 24], the SWIPT-enabled two-hop OFDM decode-and-forward (DF) relay system was considered, but the subcarrier pairing over the two hops was not involved. In [15], a PS receiver architecture was considered and a RA algorithm was proposed to improve the achievable rate; however, the complexity of algorithm was so high that it was hard to be applied to computation-constrained fog system.

This paper investigates the SWIPT for a MFC-assisted two-hop OFDM network, in which a source node transmits information to a destination with the help of a DF relay. The source is assumed with fixed energy source, while the relay is an energy-constrained node and thus has to obtain energy from wireless signal emitted by the source and further forwards the information of source to destination.

The main contributions of this paper are given as follows.

Firstly, to achieve the simultaneous information and energy transmission, we adopt a different PS ratio PS (DPS) architecture, where a frequency selective power splitter splits the signal on each subcarrier into two streams and thus all subcarriers are of the different PS ratios, which can adaptively change. Further, a particular energy cooperation strategy is considered; i.e., the energy harvested on some subcarrier of the first hop is only used to forward the information received on the corresponding subcarrier, in order to reduce the excessive computational complexity. Unlike the existing work [15], it is assumed that PS receiver splits all subcarriers into two streams with identical PS ratio, which is called identical PS ratio PS (IPS) receiver architecture in this paper.

Secondly, in order to explore the system performance limit of the proposed DPS architecture, a RA optimization problem is formulated to maximize the achievable rate of the system by jointly optimizing the subcarrier pairing (SP), the PS ratios, and the PA at both source and the relay. As the problem is nonconvex and hard to solve, a low-complexity efficient RA algorithm is designed by decomposing it into three separate subproblems. The related computation is operated at the source node, which is generally a sink node of higher computation capacity in WSN.

Thirdly, extensive simulation experiments are performed to discuss the system performance. The results demonstrate that although there are some performance loss of achievable rate of the proposed DPS architecture compared with the existing IPS architecture, the computation complexity of DPS architecture is much lower than IPS architecture. So DPS architecture may be a better option for computation-constrained fog system.

This paper is organized as follows. In Section 2, the network architecture and system model are presented, and then RA optimization problem is formulated. In Section 3, an efficient RA algorithm is designed. Simulation results are

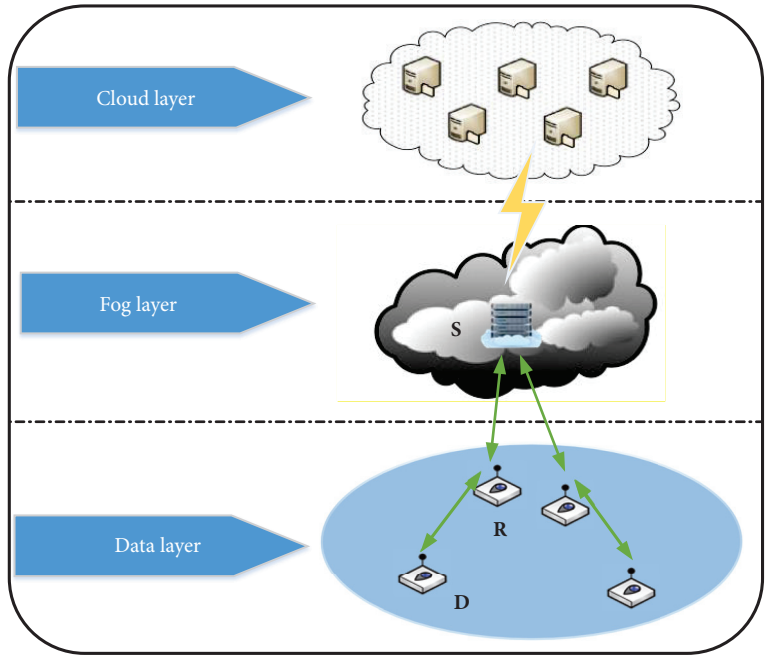


Figure 1: Network architecture of MFC-assisted two-hop OFDM system.

shown in Section 4 to discuss the performance of the DPS receiver architecture and RA algorithm. In Section 5, this paper is summarized.

2. Network Architecture and System Model

The considered network architecture is shown as in Figure 1, which is divided into three layers, i.e., data layer, fog layer, and cloud layer. Data layer comprises data nodes, which are responsible for gathering data from surrounding environment. Cloud layer contains vast resources to store and process the amount of data from data layer; meanwhile, it also sends control information to data nodes to instruct their operations. Fog layer is a bridge between data layer and cloud layer, which means that on one hand, it is responsible for collecting and preprocessing data from data layer and delivering data to cloud layer to further process it; on the other hand, it is responsible for forwarding control information from cloud layer to data layer. The information between cloud layer and fog layer is transmitted on wired channels while the information between fog layer and data layer is transmitted on wireless channels.

In this paper, we consider the transmission of control information from cloud layer to data layer. The cloud layer first sends control information to fog layer; fog layer stores the information and then forwards it to data layer. The reason of introducing the fog layer instead of directly using cloud computing is that the wireless channels are deemed to be fast time-varying channels and thus information transmission and RA task have to be performed rapidly to adapt the dynamic channels. It is worth noting that due to the enhanced computing capacity, the fog layer has the capability of performing RA algorithm according to the channel state information (CSI).

To study the information transmission from fog layer to data layer, a MFC-assisted two-hop OFDM network is considered, which consists of one source (S) in fog layer, and one destination (D) and one relay (R) in data layer, as shown in Figure 1. S desires to send information to D with the help of R. No direct link exists between S and D. S is of steady energy supply by connecting to power grid in fog layer and P_S denotes its power. R is an energy-constrained node operating in half-duplex mode and deploying DF relaying protocol, so it has to obtain energy from the signals of S and then uses the harvested energy to help S to forward information to D. The PS receiver architecture is adopted at R so that it can split the received RF signals into two streams to perform EH and ID, respectively. For such a SWIPT-enabled communication network, each transmission is based on frame of length T, which is divided into two subphases of equal length.

In the first subphase, S sends OFDM symbols to R. The received signal at R on subcarrier i can be expressed as

$$y_{r,i} = \sqrt{P_{s,i}} h_i x_i + z_{r,i}, \quad \forall i \in \{1, \ldots, N\}, \tag{1}$$

where x_i and h_i, respectively, represent transmitted symbol and channel coefficient on subcarrier i and N is the number of subcarriers. $z_{r,i}$ represents the additive white Gaussian noise (AWGN) from the antenna on subcarrier i at R, which is of zero mean and variance $\sigma^2_{r,\mathrm{att}}$. $P_{s,i}$ represents the transmission power at S on subcarrier i and satisfies

$$\sum_{i=1}^{N} P_{s,i} \le P_S, \quad P_{s,i} \ge 0, \ \forall i. \tag{2}$$

In the second subphase, using the stream for ID and EH, R, respectively, decodes the received information and harvests the energy and then reencodes the received information and forwards the reencoded information to D. Subcarrier pairing is adopted, so the information of the first hop received on subcarrier i can be transmitted on the subcarrier j in the second hop. The signal on subcarrier j received at D can be expressed as

$$y_{d,j} = \sqrt{P_{r,j}} g_j x_i + z_{d,j}, \quad \forall j \in \{1, \ldots, N\}, \tag{3}$$

where g_j is channel coefficient on subcarrier j at D, and $P_{r,j}$ is the power on subcarrier j at R. $z_{d,j}$ is AWGN from the antenna on subcarrier j at D, which is of zero mean and variance $\sigma^2_{d,\text{att}}$.

To realize SWIPT, a PS receiver architecture is proposed, where all subcarriers are of different PS (DPS) ratios and can adaptively adjust, which is called DPS architecture. To implement DPS architecture, an analog adaptive passive frequency selective power splitter is required [21, 24]. In this paper, we consider a particular energy cooperation strategy which makes the computational complexity of the RA algorithm of the DPS architecture significantly decrease and is very meaningful for some communication scenarios where the processing capacity of communication nodes is limited, such as MFC-based WSN.

Let the PS ratios θ_i^I and θ_i^E represent the fraction of the signal power used for ID and EH received on subcarrier i, respectively, which satisfy the constraints of

$$\theta_i^I + \theta_i^E = 1, \quad \theta_i^I \geq 0, \ \theta_i^E \geq 0, \ \forall i \in \{1, \ldots, N\} \tag{4}$$

Thus, the harvested energy on subcarrier i at R is given by

$$E_i = \frac{T}{2} \eta \theta_i^E |h_i|^2 P_{s,i}, \tag{5}$$

where η denotes the EH efficiency.

We consider such an *energy cooperation* strategy adopted in [21, 24], in which the energy harvested on subcarrier i of the first hop is only used to forward the information received on subcarrier i and thus the available power $P_{r,j}$ on subcarrier j at R can be inferred as

$$P_{r,j} = \frac{E_i}{T/2} = \eta \theta_i^E |h_i|^2 P_{s,i}. \tag{6}$$

The achievable information rate between S and D for DF relay system on a subcarrier pair (i, j) can be expressed as [4]

$$R_{i,j}^{DPS} = \frac{1}{2} \min \left\{ \log_2 \left(1 + \frac{\theta_i^I |h_i|^2 P_{s,i}}{\theta_i^I \sigma^2_{r,\text{att}} + \sigma^2_{r,\text{proc}}} \right), \log_2 \left(1 + \frac{|g_j|^2 P_{r,j}}{\sigma_d^2} \right) \right\}, \tag{7}$$

where $\sigma_d^2 = \sigma^2_{d,\text{att}} + \sigma^2_{d,\text{proc}}$ represents the total noise power of D on each subcarrier, $\sigma^2_{r,\text{proc}}$ and $\sigma^2_{d,\text{proc}}$, respectively, represent the power of signal processing noise on any subcarrier of R and D. In (7), the first part $\log_2(1 + \theta_i^I |h_i|^2 P_{s,i} / (\theta_i^I \sigma^2_{r,\text{att}} + \sigma^2_{r,\text{proc}}))$ represents the mutual information from S to R on subcarrier i, and the second part $\log_2(1 + |g_j|^2 P_{r,j} / \sigma_d^2)$ represents the mutual information from R to D on subcarrier j. The coefficient 1/2 in (7) is because each frame is composed of two subphases of equal length.

Substituting (6) into (7), then we can write (7) as

$$R_{i,j}^{DPS} = \frac{1}{2} \min \left\{ \log_2 \left(1 + \frac{\theta_i^I |h_i|^2 P_{s,i}}{\theta_i^I \sigma^2_{r,\text{att}} + \sigma^2_{r,\text{proc}}} \right), \log_2 \left(1 + \frac{\eta \theta_i^E |h_i|^2 |g_j|^2 P_{s,i}}{\sigma_d^2} \right) \right\}. \tag{8}$$

Thus, the achievable rate of the system can be expressed as

$$R^{DPS}(\mathscr{P}, \mathscr{S}, \boldsymbol{\theta}) = \sum_{i=1}^{N} \sum_{j=1}^{N} s_{i,j} R_{i,j}^{DPS}, \tag{9}$$

where $\boldsymbol{\theta} = \{\theta_i^I, \theta_i^E, \forall i\}$ is PS policy and satisfies the constraint (4). $\mathscr{P} = \{P_{s,i} \geq 0, | \ \forall i, j\}$ is power allocation (PA) policy and satisfies (2). $\mathscr{S} = \{s_{i,j} \in \{0, 1\} \mid \forall i, j\}$ is SP policy, which represents that if the first hop subcarrier i is matched with the second hop subcarrier j, $s_{i,j}$ =1; else $s_{i,j} = 0$. Further, one first hop (second hop) subcarrier can only match with one second hop (first hop) subcarrier. That is,

$$\sum_{j=1}^{N} s_{i,j} \leq 1, \quad \forall i, \qquad \sum_{i=1}^{N} s_{i,j} \leq 1, \quad \forall j. \tag{10}$$

With the objective of maximizing the achievable information rate of the system, by jointly optimizing the SP, the PA, and the PS ratio, the optimization problem is formulated as (**P1**):

$$\max_{\mathscr{P}, \mathscr{S}, \boldsymbol{\theta}} \quad R^{DPS}(\mathscr{P}, \mathscr{S}, \boldsymbol{\theta}) \qquad \text{s.t.} \quad (2), (4), (10). \tag{11}$$

3. Resource Allocation Design

In this section, we first describe our proposed resource allocation (RA) algorithm for problem P1 and then we shall prove that it is able to achieve the global optimal solution of problem P1.

3.1. The Proposed Resource Allocation. Our proposed RA is described as Algorithm 1, which is divided into three separate subproblems. In what follows of this subsection (Section 3.1), the detailed process of each step in Algorithm 1 is described, and its global optimality is proven in Section 3.2.

(1) The Optimal SP $\mathcal{S}^*$. The proposed SP scheme is only based on the channel power gains. Firstly, according to the channel power gains of the two hops $|h_i|^2$ and $|g_j|^2$, the first hop subcarriers and the second hop subcarriers are, respectively, sorted from highest to lowest. Next, the kth first hop subcarrier is matched with the kth second hop subcarrier, which is equivalent to the optimal $\mathcal{S}^*$ satisfying that

$$s_{i,j} = \begin{cases} 1, & \text{if } Number\left(|h_i|^2\right) = Number\left(|g_j|^2\right), \\ 0, & \text{otherwise}, \end{cases} \tag{12}$$

where $Number(|u_i|^2)$ represents the serial number of $|u_i|^2$ inside all $|u_k|^2$ for $k \in \{1, 2, \ldots, N\}$ with an degressive sorting sequence for $u \in \{h, g\}$, respectively. This scheme is called the channel gain- (CG-) sorted SP scheme. The optimality of this scheme is given as follows.

Lemma 1. *The optimal SP scheme of problem P1 is the CG-sorted SP scheme.*

Proof. To prove this lemma, two-subcarrier case is first considered and proved. Then it is further extended to general multisubcarrier case. See the Appendix for details. □

(2) The Optimal PS $\boldsymbol{\theta}^*$ with the Obtained $\mathcal{S}^*$. First, it is easily found that the problem can be decomposed into N subproblems due to the independence of each subcarrier pair. For any given subcarrier pair (i, j), the subproblem can be expressed as (**P2**):

$$\max_{\theta_i^I, \theta_i^E} \quad R_{i,j}$$
$$\text{s.t.} \quad \theta_i^I + \theta_i^E = 1, \quad \theta_i^I \geq 0, \ \theta_i^E \geq 0. \tag{13}$$

To simplify the expressions, let $A = |h_i|^2, B = \eta|g_j|^2/\sigma_d^2$ and denote $P_{s,i}$ as P_i. Since the subcarrier pair is fixed, we further drop the indexes i,j in this subsection. Thus the achievable information rate in (8) on a fixed subcarrier pair can be expressed as

$$R = \frac{1}{2} \min \left\{ \log_2 \left(1 + \frac{A\theta^I P}{\theta^I \sigma_{r,\text{att}}^2 + \sigma_{r,\text{proc}}^2} \right), \log_2 \left(1 + AB\theta^E P \right) \right\}. \tag{14}$$

It is easy to find that in (14) the first term, i.e., $\log_2(1 + A\theta^I P/(\theta^I \sigma_{r,\text{att}}^2 + \sigma_{r,\text{proc}}^2))$, is a monotonically increasing function of θ^I, and the second term, i.e., $\log_2(1 + AB\theta^E P)$, is a monotonically decreasing function of θ^I, so, to obtain the optimal solution, the two terms should be equal. Meanwhile, using $\theta^I + \theta^E = 1$, the optimal PS factor θ^I can be calculated, according to

$$B\sigma_{r,\text{att}}^2 \left(\theta^I\right)^2 + \left(1 - B\sigma_{r,\text{att}}^2 + B\sigma_{r,\text{proc}}^2\right)\theta^I - B\sigma_{r,\text{proc}}^2 = 0. \tag{15}$$

This is a quadratic equation and its two roots are given as

$$\theta^I = \frac{-\left(1 - B\sigma_{r,\text{att}}^2 + B\sigma_{r,\text{proc}}^2\right) \pm \sqrt{\left(1 - B\sigma_{r,\text{att}}^2 + B\sigma_{r,\text{proc}}^2\right)^2 + 4B^2\sigma_{r,\text{att}}^2\sigma_{r,\text{proc}}^2}}{2B\sigma_{r,\text{att}}^2}, \tag{16}$$

where only the one satisfying the constraints in problem P2 can be considered as the optimal solution. Since $\theta^I + \theta^E = 1, \theta^I, \theta^E \geq 0$, we have that $0 \leq \theta^I, \theta^E \leq 1$.

It is easy to observe that the one of the two roots

$$\theta^I = \frac{-1 + B\sigma_{r,\text{att}}^2 1 B\sigma_{r,\text{proc}}^2 - \sqrt{\left(1 - B\sigma_{r,\text{att}}^2 + B\sigma_{r,\text{proc}}^2\right)^2 + 4B^2\sigma_{r,\text{att}}^2\sigma_{r,\text{proc}}^2}}{2B\sigma_{r,\text{att}}^2} \tag{17}$$

is always less than 0, so it is discarded. For the other one, we can prove that it satisfies the above constraint.

Thus, the optimal PS factors can be given by

$$\theta^{I*} = \frac{-1 + B\sigma_{r,\text{att}}^2 - B\sigma_{r,\text{proc}}^2 + \sqrt{\left(1 - B\sigma_{r,\text{att}}^2 + B\sigma_{r,\text{proc}}^2\right)^2 + 4B^2\sigma_{r,\text{att}}^2\sigma_{r,\text{proc}}^2}}{2B\sigma_{r,\text{att}}^2}. \tag{18}$$

(1) Find the optimal $\mathcal{S}^*$ from (12).
(2) Calculate the optimal $\boldsymbol{\theta}^*$ with the obtained $\mathcal{S}^*$ from (18).
(3) Calculate the optimal $\mathcal{P}^*$ with the obtained $\mathcal{S}^*$ and $\boldsymbol{\theta}^*$ from (22).

Algorithm 1: Resource allocation algorithm.

(3) The Optimal PA $\mathcal{P}^*$ with the Obtained $\mathcal{S}^*$ and $\boldsymbol{\theta}^*$. We have obtained the optimal PS factors θ^{I*} and θ^{E*}. As the optimal PS factors are related to the channel gain of the second hop from (18), θ_i^{I*}, θ_i^{E*} are represented as $\theta_{i,j}^{I*}, \theta_{i,j}^{E*}$ for given subcarrier pair (i,j). Since the two terms are equal in (8) for optimal $\theta_{i,j}^{I*}, \theta_{i,j}^{E*}$, (8) can be transformed as

$$R_{i,j} = \frac{1}{2}\log_2\left(1 + \frac{|h_i|^2 \theta_{i,j}^{I*} P_i}{\theta_{i,j}^{I*}\sigma_{r,\text{att}}^2 + \sigma_{r,\text{proc}}^2}\right). \tag{19}$$

We denote $|h_i|^2\theta_{i,j}^{I*}/(\theta_{i,j}^{I*}\sigma_{r,\text{att}}^2 + \sigma_{r,\text{proc}}^2)$ as $\gamma_{i,j}$, and then (19) is transformed into

$$R_{i,j} = \frac{1}{2}\log_2\left(1 + \gamma_{i,j}P_i\right). \tag{20}$$

So for given $\mathcal{S}^*$ and $\boldsymbol{\theta}^*$, the PA problem can be formulated as (**P3**):

$$\begin{aligned} \max_{\mathcal{P}} \quad & \sum_{(i,j)\in\mathcal{SP}} R_{i,j} \\ \text{s.t.} \quad & (2), \end{aligned} \tag{21}$$

where $\mathcal{SP}$ is the set of subcarrier pairs. From (18), it can be easily found that the optimal PS factors are not related to PA and thus $\gamma_{i,j}$ is also not related to PA. So this problem is a classical water-filling PA problem, and we can obtain its optimal solution as

$$P_i^* = \left[\frac{1}{\nu \ln 2} - \frac{1}{\gamma_{i,j}}\right]^+, \quad \forall i, \tag{22}$$

where $[x]^+ = \max\{0, x\}$, and ν is Lagrangian multiplier and can be solved using $\sum_{i=1}^{N}[1/(\nu\ln 2) - 1/\gamma_{i,j}]^+ = P_S$.

3.2. Global Optimum of Our Proposed RA. In this subsection, we shall prove that although Algorithm 1 is divided into three separate subproblems, it can still achieve the global optimal solution of problem P1, and the result is given by the following theorem.

Theorem 2. *The RA in Algorithm 1 achieves the global optimum of problem P1.*

Proof. To prove that the RA policy in Algorithm 1 can achieve the global optimal solution of problem P1, we only need to prove that each step of Algorithm 1 maintains the global optimum. From Lemma 1, we have known that the CG-sorted SP scheme in step 1 of Algorithm 1 gives the globally optimal SP policy. The scheme is only related to channel gains, which does not require the knowledge of the optimal PS and PA. According to the derivation process in step 2 of Algorithm 1, the obtained PS is optimal under the given optimal SP, and it does not require the knowledge of optimal PA. Then in step 3 of Algorithm 1, the obtained PA is optimal under the given optimal SP and optimal PS. Since each step maintains the global optimum, Theorem 2 is proved. □

3.3. Complexity Analysis. The complexity of step 1 of Algorithm 1 depends on the adopted sorting method, which is $O(N\log N)$ if the quick-sort method is applied. Moreover, the complexity of step 2 of Algorithm 1 is $O(N)$, and the complexity of step 3 of Algorithm 1 is also $O(N)$ (the water-filling over the sorted $\gamma_{i,j}$) [25]. Thus, the total computational complexity can be expressed by $O(N\log N)$. For comparison, the computational complexity of RA algorithm for IPS architecture in [15] is $O(N\log N + MN(N+2)^q)$, where M is the number of loops in the algorithm [15], so it can be found that the computational complexity of proposed DPS architecture's RA algorithm is on the order of $M(N+2)^q/\log N$ less than the IPS architecture. So the proposed DPS architecture may be more proper for MFC-assisted networks, where the devices are of lower computation capacity.

4. Simulation Results

In this section, some simulation results are given to illustrate the performance of the presented DPS receiver architecture and RA algorithm. The noise powers are assumed follows: ${\sigma_r}^2_{\text{att}} = {\sigma_r}^2_{\text{proc}} = -33$ dBm and ${\sigma_d}^2 = -30$ dBm. The three network nodes (S, R, and D) are assumed to be placed on a straight line. The distance from S to D is reference distance and represented by d_0, where $d_0 = 10$ m. The location of R is expressed as d_r/d_0, where d_r denotes the distance from S to R. h_i and g_j are, respectively, obtained from the distribution as

$$\begin{aligned} h_i &\sim CN\left(0, \frac{1}{L\left(1+d_r\right)^{\alpha}}\right), \\ g_j &\sim CN\left(0, \frac{1}{L\left(1+\left(d_0-d_r\right)\right)^{\alpha}}\right), \end{aligned} \tag{23}$$

where α is the path loss factor and set to be 3 and L is the number of taps and set to be 4.

Firstly, we discuss the performance of our proposed Algorithm 1. For comparisons, the three other methods are also simulated, i.e., (1) OPAwoSP method: Optimal PA without SP; (2) EPAwSP method: Equal PA with SP; (3) EPAwoSP

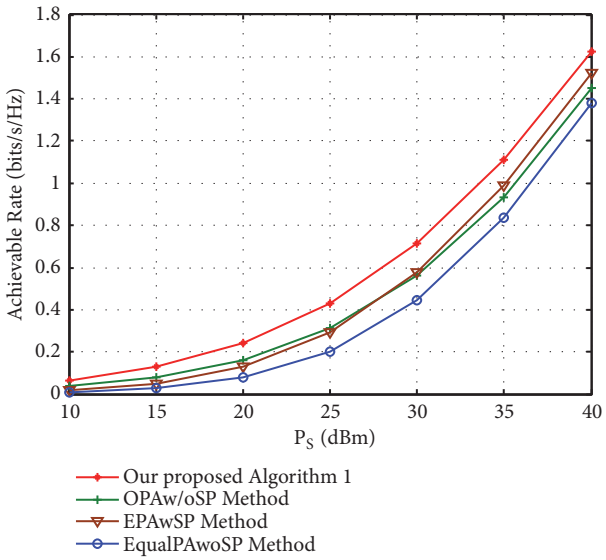


FIGURE 2: Achievable information rate versus P_S with $N = 64$ and R located at the midpoint between S and D for DPS architecture.

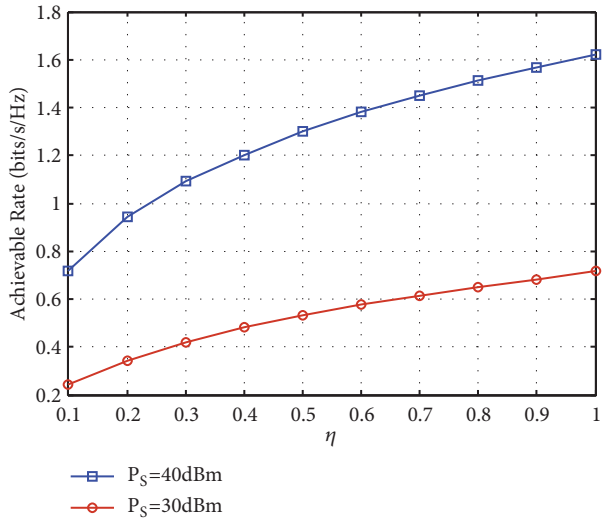


FIGURE 3: Achievable information rate versus EH efficiency η with $N = 64$ and R located at the midpoint between S and D for DPS architecture.

method: Equal PA without SP. EH efficiency $\eta = 1$. In Figure 2, we plot the achievable rates versus the total power P_S. It is easily seen that our proposed Algorithm 1 is superior to the three other methods.

We also show the effect of EH efficiency η on the achievable rate in Figure 3 and it can be found that when $\eta = 1$, the achievable rate is maximum. In the following simulations, to discuss the SWIPT-enabled system's performance limit, EH efficiency is always set to be $\eta = 1$.

Secondly, to figure out the system performance of the DPS receiver architecture, we compare the achievable information rates of the proposed DPS architectures and IPS architecture in [15]. For IPS architecture, let ρ be the PS ratio used for ID, and the rest $1 - \rho$ part is used for EH; ρ should satisfy

$$0 \le \rho \le 1. \tag{24}$$

The energy obtained by R is $E_R = (T/2)\eta(1-\rho)\sum_{i=1}^{N}|h_i|^2 P_{s,i}$ and the available power of R is $P_R = E_R/(T/2) = \eta(1-\rho)\sum_{i=1}^{N}|h_i|^2 P_{s,i}$. So the available power$P_{r,j}$ on subcarrier j at R satisfies

$$\sum_{j=1}^{N} P_{r,j} \le \eta(1-\rho)\sum_{i=1}^{N}|h_i|^2 P_{s,i}. \tag{25}$$

The achievable rate from S to D on each subcarrier pair (i, j) can be expressed as

$$R_{i,j}^{IPS} = \frac{1}{2}\min\left\{\log_2\left(1+\frac{\rho|h_i|^2 P_{s,i}}{\rho\sigma_{r,\text{att}}^2+\sigma_{r,\text{proc}}^2}\right), \log_2\left(1+\frac{|g_j|^2 P_{r,j}}{\sigma_d^2}\right)\right\}, \tag{26}$$

and thus the achievable rate of the system is given by

$$R^{IPS}(\mathcal{P},\mathcal{S},\rho) = \sum_{i=1}^{N}\sum_{j=1}^{N} s_{i,j}R_{i,j}^{IPS}. \tag{27}$$

To maximize the achievable rate of the system, an optimization problem is formulated as

$$\max_{\mathcal{P},\mathcal{S},\rho} \quad R^{IPS}(\mathcal{P},\mathcal{S},\rho) = \sum_{i=1}^{N}\sum_{j=1}^{N} s_{i,j}R_{i,j}$$
$$\text{s.t.} \quad (2),(8),(24),(25). \tag{28}$$

The solution of the problem is given in [15]. In addition, conventional non-SWIPT two-hop OFDM system is also compared in order to show the difference between SWIPT-enabled and non-SWIPT systems. For the non-SWIPT system, we use the optimal RA algorithm proposed in [8].

In Figures 4 and 5, the achievable rates of the DPS/IPS architectures and non-SWIPT system with respect to P_S and d_r/d_0 are given, respectively. In these two figures, it can be found that the achievable rate of the non-SWIPT system is higher than SWIPT-enabled IPS/DPS architectures, and DPS architecture is worse than IPS architecture.

Moreover, from Figure 5, one can find that, for the SWIPT-enabled system, when R is placed close to S or D, the system can obtain the better performance, and the proposed DPS architecture agrees with the existing IPS architecture. One can also find that, for conventional non-SWIPT system, the achievable information rate achieves maximum when R is placed at the midpoint on the line from S to D.

Finally, we also compare the average running time of IPS and DPS architectures in Figure 6. It shows that the running efficiency of the DPS architecture is far superior to the IPS architecture, which agrees with the analysis

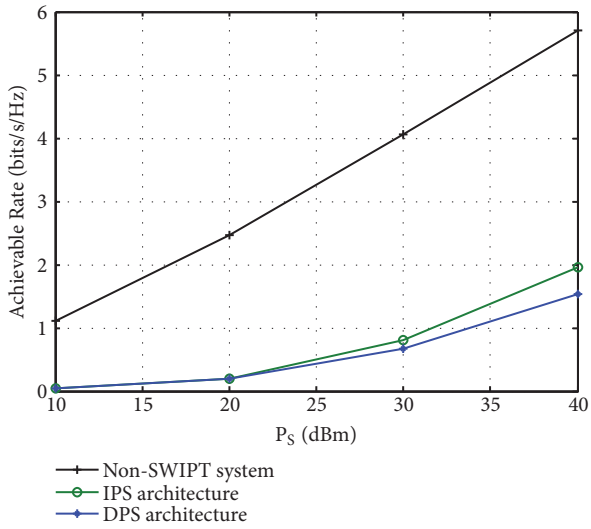


FIGURE 4: Comparison of achievable information rates of DPS architecture, IPS architecture, and non-SWIPT system versus P_S with $N = 4$ and R located at the midpoint between S and D.

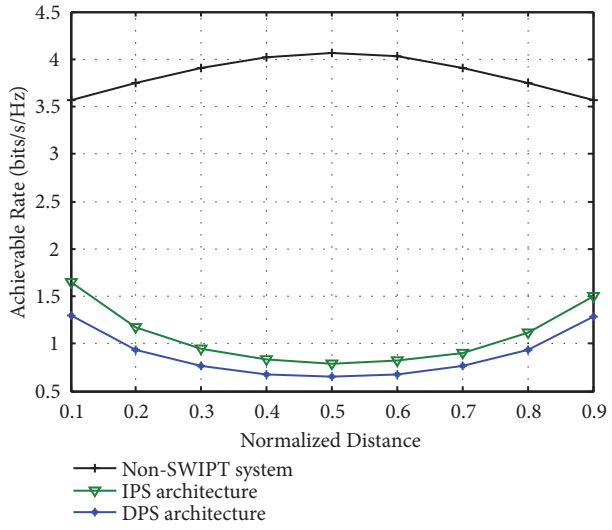


FIGURE 5: Comparison of achievable information rates of DPS architecture, IPS architecture, and non-SWIPT system versus relay location with P_S=30dBm and $N = 4$.

of computational complexity in Section 3.3, and thus, for computation-constrained MFC system, DPS architecture is a better option although there are some loss of the achievable rate compared with IPS architecture.

5. Conclusion

This paper investigated SWIPT for MFC-assisted two-hop OFDM network and proposed DPS receiver architectures. To study the system achievable rate limit, an efficient RA algorithm was given. In simulations, it was found that the achievable rate of the DPS architecture is worse than the existing IPS architecture; however, the computation complexity

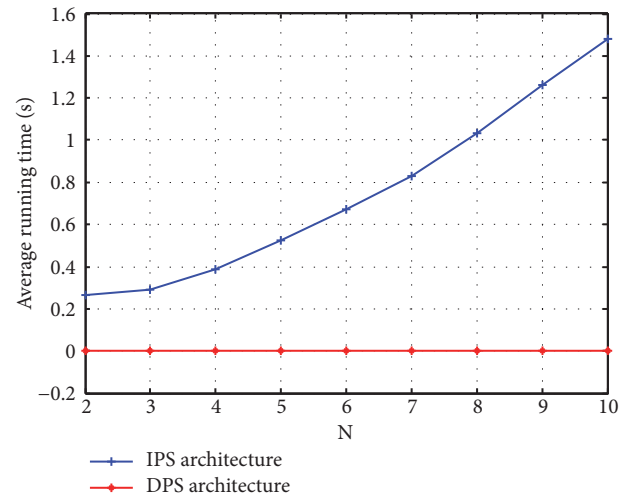


FIGURE 6: Comparison of running time of IPS and DPS architectures versus the number of subcarriers N with R located at the midpoint from S to D and P_S=10dBm.

of DPS architecture is much lower than IPS architecture. So DPS architecture may be a better option for computation-constrained MFC system.

Appendix

Proof of Lemma 1

(a) Two-Subcarrier Case (N=2). Firstly, it is assumed that two hops' channel gains satisfy $|h_1|^2 > |h_2|^2, |g_1|^2 > |g_2|^2$. Observing Algorithm 1, we know the second step and the third step can be applied to any given subcarrier pairing policy in fact, although they are derived from the optimal subcarrier pairing policy. So the achievable information rate of the system using sorted subcarrier pairs (1,1) and (2,2) can be expressed as

$$R_{\text{sort}} = \frac{1}{2}\log_2\left(1+\gamma_{1,1}P_1^*\right) + \frac{1}{2}\log_2\left(1+\gamma_{2,2}P_2^*\right), \quad \text{(A.1)}$$

and the achievable rate using nonsorted subcarrier pairs (1,2) and (2,1) can be expressed as

$$R_{\text{nonsort}} = \frac{1}{2}\log_2\left(1+\gamma_{1,2}P_1'^*\right) + \frac{1}{2}\log_2\left(1+\gamma_{2,1}P_2'^*\right), \quad \text{(A.2)}$$

where $P_1'^*, P_2'^*$ are the optimal powers for nonsorted pairing scheme.

To prove Lemma 1, we need to prove $R_{\text{sort}} > R_{\text{nonsort}}$; i.e.,

$$\left(1+\gamma_{1,1}P_1^*\right)\left(1+\gamma_{2,2}P_2^*\right) > \left(1+\gamma_{1,2}P_1'^*\right)\left(1+\gamma_{2,1}P_2'^*\right). \quad \text{(A.3)}$$

We define a new function $f(\theta_{i,j}^{I*}) = \theta_{i,j}^{I*}/(\theta_{i,j}^{I*}\sigma_{r,\text{att}}^2 + \sigma_{r,\text{proc}}^2)$, so $\gamma_{i,j} = |h_i|^2 f(\theta_{i,j}^{I*})$. For further simplifying the expressions,

let $H_i = |h_i|^2$ and $G_j = f(\theta_{i,j}^{I*})$. Note that, from (18), we can observe that for given noise power, the optimal PS factor $\theta_{i,j}^{I*}$ is only related to B, that is, to the channel gain $|g_j|^2$ of the second hop since $B = \eta|g_j|^2/\sigma_d^2$. Thus, for G_j, we only reserve the subscript j. One sees that $\gamma_{i,j} = H_i G_j$; thus (A.3) is equivalent to

$$\begin{aligned}&\left(H_1G_1P_1^* + H_2G_2P_2^* + H_1G_1H_2G_2P_1^*P_2^*\right)\\&\quad-\left(H_1G_2P_1'^* + H_2G_1P_2'^* + H_1G_2H_2G_1P_1'^*P_2'^*\right)\\&\quad>0.\end{aligned}\tag{A.4}$$

Secondly, we can prove that $G_1 > G_2$ for the assumption $|g_1|^2 > |g_2|^2$. From (18), the derivative of $\theta_{i,j}^{I*}$ with respect to B can be computed as

$$\left(\theta_{i,j}^{I*}\right)' = \frac{1}{2\sigma_{r,\text{att}}^2B^2}\left(1 - \frac{1/B - \left(\sigma_{r,\text{att}}^2 - \sigma_{r,\text{proc}}^2\right)}{\sqrt{\left(1/B - \left(\sigma_{r,\text{att}}^2 - \sigma_{r,\text{proc}}^2\right)\right)^2 + 4\sigma_{r,\text{att}}^2\sigma_{r,\text{proc}}^2}}\right).\tag{A.5}$$

One can easily find that $(\theta_{i,j}^{I*})' > 0$; that is to say, $\theta_{i,j}^{I*}$ is a monotonically increasing function of B. Meanwhile we know $B = \eta|g_j|^2/\sigma_d^2$, so with the increase of $|g_j|^2$, $\theta_{i,j}^{I*}$ increases. The increment of $\theta_{i,j}^{I*}$ will further result in the increment of $f(\theta_{i,j}^{I*})$. Thus, according to the assumption $|g_1|^2 > |g_2|^2$, we have $f(\theta_{i,1}^{I*}) > f(\theta_{i,2}^{I*})$; that is, $G_1 > G_2$.

Thirdly, for the two-subcarrier case, the explicit solutions of optimal PA can be obtained. When only total power constraint in (2) is considered and inequality constraints are ignored, the optimal P_1^*, P_2^* for sorted pairing scheme can be derived as

$$\begin{aligned}P_1^* &= \frac{P_S}{2} + \frac{H_1G_1 - H_2G_2}{2H_1G_1H_2G_2},\\P_2^* &= \frac{P_S}{2} - \frac{H_1G_1 - H_2G_2}{2H_1G_1H_2G_2}.\end{aligned}\tag{A.6}$$

Similarly, the optimal $P_1'^*, P_2'^*$ for nonsorted pairing scheme can be derived as

$$\begin{aligned}P_1'^* &= \frac{P_S}{2} + \frac{H_1G_2 - H_2G_1}{2H_1G_1H_2G_2},\\P_2'^* &= \frac{P_S}{2} - \frac{H_1G_2 - H_2G_1}{2H_1G_1H_2G_2}.\end{aligned}\tag{A.7}$$

It is worth noting that, due to nonnegative power constraint in (2), (A.6) and (A.7) are valid only for $0 \le P_1^*, P_2^*, P_1'^*, P_2'^* \le P_S$. If $P_1^*, P_2^*, P_1'^*, P_2'^*$ do not satisfy this condition, then $P_1^* = P_S, P_2^* = 0$ or $P_1'^* = P_S, P_2'^* = 0$.

So we consider the following three cases, namely, Case 1 ($0 < P_1^*, P_2^*, P_1'^*, P_2'^* < P_S$), Case 2 ($P_1^* = P_S, P_2^* = 0, P_1'^* = P_S, P_2'^* = 0$), and Case 3 ($P_1^* = P_S, P_2^* = 0, 0 < P_1'^*, P_2'^* < P_S$). For the remaining case ($0 < P_1^*, P_2^* < P_S, P_1'^* = P_S, P_2'^* = 0$), it is easy to find that it will not occur, because we can derive that P_1^* is necessarily larger than $P_1'^*$ according to our assumptions $|h_1|^2 > |h_2|^2$ and $|g_1|^2 > |g_2|^2$. Noting that here we only consider $P_1'^* > P_2'^*$, that is, $H_1G_2 - H_2G_1 > 0$, the analysis for $P_1'^* \le P_2'^*$ is similar. Then we can derive and prove (A.4) for the three cases.

Case 1. For this case, using (A.6) and (A.7), we can obtain

$$\begin{aligned}&H_1G_1P_1^* + H_2G_2P_2^* + H_1G_1H_2G_2P_1^*P_2^*\\&\quad-\left(H_1G_2P_1'^* + H_2G_1P_2'^* + H_1G_2H_2G_1P_1'^*P_2'^*\right)\\&\quad=\frac{(H_1 - H_2)(G_1 - G_2)}{2}P_S\\&\quad+\frac{\left(H_1^2 - H_2^2\right)\left(G_1^2 - G_2^2\right)}{4H_1G_1H_2G_2} > 0,\end{aligned}\tag{A.8}$$

where inequality is obtained from our assumption $H_1 > H_2$ and the obtained result $G_1 > G_2$.

Case 2. For this case, we can derive that

$$\begin{aligned}&H_1G_1P_1^* + H_2G_2P_2^* + H_1G_1H_2G_2P_1^*P_2^*\\&\quad-\left(H_1G_2P_1'^* + H_2G_1P_2'^* + H_1G_2H_2G_1P_1'^*P_2'^*\right)\\&\quad=H_1P_S(G_1 - G2) > 0,\end{aligned}\tag{A.9}$$

where inequality is obtained since$G_1 > G_2$.

Case 3. For this case, using (A.7), we can obtain

$$\begin{aligned}&H_1G_1P_1^* + H_2G_2P_2^* + H_1G_1H_2G_2P_1^*P_2^*\\&\quad-\left(H_1G_2P_1'^* + H_2G_1P_2'^*\right.\\&\quad\left.+ H_1G_2H_2G_1P_1'^*P_2'^*\right) = H_1G_1P_S\\&\quad-\left(\frac{H_1G_2 + H_2G_1}{2}P_S + \frac{H_1G_2H_2G_1}{4}P_S^2\right.\\&\quad\left.+\frac{(H_1G_2 - H_2G_1)^2}{4H_1G_2H_2G_1}\right).\end{aligned}\tag{A.10}$$

According to $0 < P_1'^*, P_2'^* < P_S$ and (A.7), one can see that P_S satisfies

$$\frac{H_1G_2 - H_2G_1}{H_1G_1H_2G_2} < P_S \le \frac{H_1G_1 - H_2G_2}{H_1G_1H_2G_2}.\tag{A.11}$$

In this interval, we can prove that (A.10) is a monotonically increasing function of P_S. So we substitute the lower

bound of the interval $(H_1G_2 - H_2G_1)/H_1G_1H_2G_2$ into (A.10), and then derive that

$$\begin{aligned} & H_1G_1P_1^* + H_2G_2P_2^* + H_1G_1H_2G_2P_1^*P_2^* \\ & - \left(H_1G_2P_1'^* + H_2G_1P_2'^* \right. \\ & \left. + H_1G_2H_2G_1P_1'^*P_2'^*\right) \\ & = \frac{2H_1\left(H_1G_2 - H_2G_1\right)\left(G_1 - G_2\right)}{2H_1G_1H_2G_2} > 0, \end{aligned} \tag{A.12}$$

where inequality is obtained from the aforementioned condition $H_1G_2 - H_2G_1 > 0$ and $G_1 > G_2$.

Since (A.10) is a monotonically increasing function of P_S in the whole interval, (A.10) is always more than 0.

In summary, it is proved that, for all cases, (A.4) always holds. So, for two-subcarrier case, Lemma 1 is proved.

(b) Multisubcarrier Case ($N > 2$). The two-subcarrier case can be generalized to the multisubcarrier case. A proof by contradiction is adopted. For an N-subcarrier relay system with $N > 2$, suppose the optimal pairing does not follow the sorted pairing rule of Lemma 1, so there are at least two pairs of incoming and outgoing subcarriers that are mismatched according to their channel gains. Without loss of generality, it is assumed that there are two pairs (i_1, j_1) and (i_2, j_2) satisfying $|h_{i_1}|^2 > |h_{i_2}|^2, |g_{j_1}|^2 < |g_{j_2}|^2$. Using the result for $N = 2$, it is found that pairing subcarrier i_1 with subcarrier j_2 and pairing subcarrier i_2 with subcarrier j_1 can achieve a higher rate than the nonsorted pairings. Hence, by using this new pairing while maintaining the other subcarrier pairs invariant, the total achievable rate can be increased. This contradicts our assumption on the optimality of a nonsorted pairing scheme.

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 there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (no. 61602034), by the Beijing Natural Science Foundation (no. 4162049), and by the Young Talents Programme of State Grid Energy Research Institute Co., Ltd. (no. XM2018020035180), Key Technologies on Two-Path Cooperative Relay Transmission for Energy-Constrained Wireless Networks.

References

[1] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," *Wireless Communications and Mobile Computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[2] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A Survey on Mobile Edge Computing: The Communication Perspective," *IEEE Communications Surveys & Tutorials*, 2017.

[3] C. Huang, R. Lu, and K. R. Choo, "Vehicular Fog Computing: Architecture, Use Case, and Security and Forensic Challenges," *IEEE Communications Magazine*, vol. 55, no. 11, pp. 105–111, 2017.

[4] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: efficient protocols and outage behavior," *IEEE Transactions on Information Theory*, vol. 50, no. 12, pp. 3062–3080, 2004.

[5] T. Wang, R. C. De Lamare, and A. Schmeink, "Alternating optimization algorithms for power adjustment and receive filter design in multihop wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 1, pp. 173–184, 2015.

[6] W. Yu and R. Lui, "Dual methods for nonconvex spectrum optimization of multicarrier systems," *IEEE Transactions on Communications*, vol. 54, no. 7, pp. 1310–1322, 2006.

[7] T. Wang and L. Vandendorpe, "Sum rate maximized resource allocation in multiple DF relays aided OFDM transmission," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 8, pp. 1559–1571, 2011.

[8] N. Kumar, S. Sharma, and V. Bhatia, "Performance Analysis of OFDM-Based Nonlinear AF Multiple-Relay Systems," *IEEE Wireless Communications Letters*, vol. 6, no. 1, pp. 122–125, 2017.

[9] S. Jin, Z. Zhu, Y. Yang, M. Zhou, and X. Luo, "Alternate distributed allocation of time reuse patterns in Fog-enabled cooperative D2D networks," in *Proceedings of the 2017 IEEE Fog World Congress (FWC)*, pp. 1–6, Santa Clara, CA, October 2017.

[10] W. Zhang, Z. Zhang, and H. Chao, "Cooperative Fog Computing for Dealing with Big Data in the Internet of Vehicles: Architecture and Hierarchical Resource Management," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 60–67, 2017.

[11] Y. Yu, J. Zhang, and K. B. Letaief, "Joint subcarrier and CPU time allocation for mobile edge computing," in *Proceedings of the 59th IEEE Global Communications Conference, GLOBECOM 2016*, USA, December 2016.

[12] M. Li, S. Yang, Z. Zhang, J. Ren, and G. Yu, "Joint subcarrier and power allocation for OFDMA based mobile edge computing system," in *Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–6, Montreal, QC, October 2017.

[13] K. Xiong, B. Wang, and K. J. R. Liu, "Rate-Energy Region of SWIPT for MIMO Broadcasting under Nonlinear Energy Harvesting Model," *IEEE Communications Letters*, vol. 16, no. 8, pp. 5147–5161, 2017.

[14] L. R. Varshney, "Transporting information and energy simultaneously," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '08)*, pp. 1612–1616, IEEE, Toronto, Canada, July 2008.

[15] X. Di, K. Xiong, Y. Zhang, and Z. Qiu, "Simultaneous wireless information and power transfer in two-hop OFDM decode-and-forward relay networks," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 1, pp. 152–167, 2016.

[16] X. Di, K. Xiong, P. Fan, and H.-C. Yang, "Simultaneous wireless information and power transfer in cooperative relay networks with rateless codes," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 2981–2996, 2017.

[17] K. Huang and E. Larsson, "Simultaneous information and power transfer for broadband wireless systems," *IEEE Transactions on Signal Processing*, vol. 61, no. 23, pp. 5972–5986, 2013.

[18] W. Lu, Y. Gong, J. Wu, H. Peng, and J. Hua, "Simultaneous wireless information and power transfer based on joint subcarrier and power allocation in OFDM systems," *IEEE Access*, vol. 5, pp. 2763–2770, 2017.

[19] X. Zhou, R. Zhang, and C. K. Ho, "Wireless information and power transfer in multiuser OFDM systems," *IEEE Transactions on Wireless Communications*, vol. 13, no. 4, pp. 2282–2294, 2014.

[20] O. T. Demir and T. E. Tuncer, "Max–Min Fair Resource Allocation for SWIPT in Multi-Group Multicast OFDM Systems," *IEEE Communications Letters*, vol. 21, no. 11, pp. 2508–2511, 2017.

[21] K. Xiong, P. Fan, C. Zhang, and K. B. Letaief, "Wireless information and energy transfer for two-hop non-regenerative MIMO-OFDM relay networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 8, pp. 1595–1611, 2015.

[22] G. Huang and W. Tu, "Wireless Information and Energy Transfer in Nonregenerative OFDM AF Relay Systems," *Wireless Personal Communications*, vol. 94, no. 4, pp. 3131–3146, 2017.

[23] Y. Liu and X. Wang, "Information and Energy Cooperation in OFDM Relaying: Protocols and Optimization," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 7, pp. 5088–5098, 2016.

[24] K. Xiong, C. Chen, G. Qu, P. Fan, and K. B. Letaief, "Group Cooperation with Optimal Resource Allocation in Wireless Powered Communication Networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3840–3853, 2017.

[25] D. P. Palomar and J. R. Fonollosa, "Practical algorithms for a family of waterfilling solutions," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, part 1, pp. 686–695, 2005.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 9340194, 6 pages
https://doi.org/10.1155/2018/9340194

Research Article

Improved Convolutional Neural Network for Chinese Sentiment Analysis in Fog Computing

Haoping Chen, Lukun Du, Yueming Lu, and Hui Gao

Key Laboratory of Trustworthy Distributed Computing and Service (BUPT), Ministry of Education, Beijing University of Posts and Telecommunications, Beijing, China

Correspondence should be addressed to Haoping Chen; chenhaopingbupt@163.com

Received 22 June 2018; Accepted 15 August 2018; Published 23 September 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Haoping 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.

Fog computing extends the concept of cloud computing to the edge of network to relieve performance bottleneck and minimize data analytics latency at the central server of a cloud. It uses edge nodes directly to perform data input and data analysis. In public opinion analysis system, edge nodes that collect opinions from users are responsible for some data filtering jobs including sentiment analysis. Therefore, it is crucial to find suitable algorithm that is lightweight in operation and accurate in predictive performance. In this paper, we focus on Chinese sentiment analysis job in fog computing environment and propose a non-task-specific method called Channel Transformation Based Convolutional Neural Network (CTBCNN) for Chinese sentiment classification, which uses a new structure called channel transformation based (CTB) convolutional layer to enhance the ability of automatic feature extraction and applies global average pooling layer to prevent overfitting. Through experiments and analysis, we show that our method do achieve competitive accuracy and it is convenient to apply this method to different cases in operation.

1. Introduction

The concept of fog computing [1, 2] is extended from cloud computing which puts a substantial amount of data analysis to edge nodes. These edge nodes are densely geographically deployed and are close to original data input. The main purpose of fog computing is to relieve network traffic load and reduce data calculation latency at the cloud. With the explosive growth of data from the Internet, fog computing is getting more and more important in Internet of Things (IoT) [3], Intrusion Detection, and many other fields. In a conceptual framework of public opinion analysis system, edge nodes are responsible for some data filtering jobs including sentiment analysis. Therefore, operations on these edge nodes are supposed to be lightweight and accurate in predictive performance.

Sentiment analysis is an important task in many real-world applications and there are many researches on it. Most researches on sentiment analysis mainly divide into two categories: unsupervised methods based on sentimental lexicon and supervised machine learning methods. The first strategy identifies polarity of text using sentiment lexicons. Reference [4] implemented sentiment value calculation and classification of microblog texts by extensible sentiment dictionary. Reference [5] combined basic emotion value lexicon and social evidence lexicon to improve traditional polarity lexicon, thus achieving significant improvement in Chinese text sentiment analysis. There are various ways to express the same opinion in Chinese. Therefore, it is impossible for any lexicon to cover all sentiment words or phrases. Furthermore, the same word in different field may have different sentimental impacts, which means lexicon is usually task-specific, while supervised machine learning methods use traditional text classification methods, such as Naive Bayes (NB) and Support Vector Machine (SVM). Pang applied several machine learning techniques to the sentiment classification problem in [6], which used movie reviews as data and the results showed that SVM outperform other methods. Feature extraction methods based on TF-IDF, Mutual Information, and Chi-Square [7] are commonly used for machine learning methods. However, classification performance varies a great deal with different feature selection methods [8]. Meanwhile, feature engineering is also task-specific. As far as we know, most sentiment analysis methods are implemented at cloud

server. It is crucial to find a suitable algorithm that works in fog computing environment. Recently, more and more researchers apply deep neural networks to sentiment analysis [9–11]. And the release of TensorFlow Lite makes it possible to run deep learning models on portable equipment.

In this work, we focus on Chinese sentiment analysis and aim at presenting a lightweight, non-task-specific method with high portability in fog computing environment without manual feature engineering. We propose a method called Channel Transformation Based Convolutional Neural Network (CTBCNN), which is an improved model of Convolutional Neural Network (CNN). Firstly, we input sentences and obtain word vectors by skip-gram model [12, 13] and keep them static. Then we utilize a new structure of called channel transformation based (CTB) convolutional layer, which enhances the capability of automatic feature extraction by considering the channel information of the output of the previous convolutional layer. Our method replaces the fully connected layer by global average pooling layer to prevent overfitting. Global average pooling was proved effective in computer vision tasks by Lin Min [14]. Inspired by Lin M's work, we study the outputs by global average pooling layer instead of fully connected layer.

The main contributions of this work are presented as follows. Firstly, we present an effective method that is suitable for fog computing environment. The method can be applied to different cases to handle different data conveniently without much human operations. Secondly, we put forward the idea of channel transformation for convolution layer, which is able to cover more information and extract more representative feature. Thirdly, our work proves the regularization effect of global average pooling layer in nature language processing task.

This work is organized as follows. Section 2 introduces the overall model of CTBCNN. Section 3 shows the structure of the stacking of CTB convolutional layers. Section 4 presents the implementation of global average pooling layer. Section 5 shows the experimental results. We conclude this work in Section 6.

2. Channel Transformation Based Convolutional Neural Network

We first present the overall model structure of CTBCNN model, which is presented in Figure 1. Compared with classic CNN [15], CTBCNN has two main differences. Firstly, CTBCNN has three CTB convolutional layers. Channel transformation is a trick of matrix transformation that allows us to implement convolution not only on height×width plane but also on height×depth plane and depth×width plane. This kind of convolution enhances capability of feature extraction by covering more information in channel depth dimension. The second difference is that we replace the fully connected layer with global average pooling layer because fully connected layer is a kind of dense connection which is prone to overfitting.

CTBCNN takes sentence vector as input and extracts feature maps from each input sentence by the three CTB convolutional layers and then utilize global average pooling over these feature maps to output the sentiment classification result.

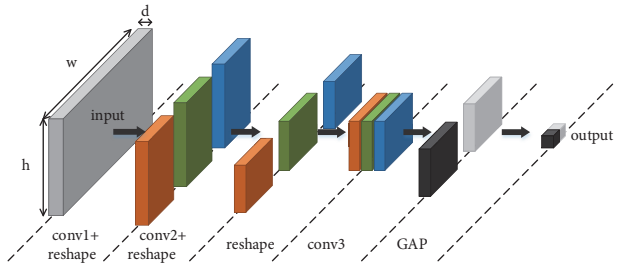


Figure 1: The overall structure of Channel Transformation Based Convolutional Neural Network. The model basically consists of three channel transformation based convolutional layers and a global average pooling (GAP) layer. This paper focuses on binary classification and the number of output categories is 2.

In the following two sections, we will present the structure of the three channel transformation based convolutional layers and global average pooling layer in detail.

3. Channel Transformation Based Convolutional Layer

In this section, we first give the definition of the shape of a matrix or a vector. In the second part we introduce the conventional convolutional layer and convolutional process. The third part presents the CTB convolutional layer on the basis of conventional convolutional layer. Finally we stack CTB convolutional layers and give the three CTB convolutional layers structure.

3.1. Definition of Shape. To make it more understandable, we define the shape of a matrix or vector using three dimensions: height, width, and depth. The input and output result can be represented by their shape in an intuitive manner of the convolutional layer and, for example, we have an image which size is h×w and has RGB 3 channels; then the shape of the image can be represented as (h, w, 3).

3.2. Conventional Convolutional Layer. The input is a sentence vector that can be represented as

$$x_{1:n} = x_1 \oplus x_2 \oplus \ldots \oplus x_n \tag{1}$$

where $x_i \epsilon R^k$ is a k dimensional word vector of the i^{th} word in sentence that consists of n words and $\oplus$ means concatenation operation. Thus the input sentence vector is similar to an "image" whose shape is (n, k, 1).

Generally in convolution process, a filter $W \epsilon R^{h_c \times k}$ like a window of h_c words is applied to generate a new feature c_i.

$$c_i = f\left(W \bullet x_{i:i+h_c-1} + b\right) \tag{2}$$

where $b \epsilon R$ is a bias term and f is an activation function. In this work we use ReLU as the activation function. From [16] we know that ReLU fuction has faster convergence speed

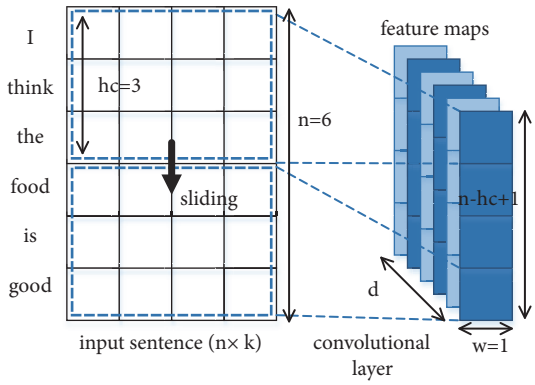


FIGURE 2: The convolution process with one input channel for an example sentence. Filter $W \epsilon R^{h_c \times k}$ extracts one feature $\epsilon R^{(n-h_c+1)\times 1}$ and with d filters we can extract d features.

of gradient descent. As a result of the convolution, a feature map c is generated where

$$c = \left[c_1, c_2, \ldots, c_{n-h_c+1}\right] \tag{3}$$

Figure 2 gives an example of conventional convolution. One feature is extracted from one filter. Assume the number of filters is d; then the shape of convolutional output is $(n - h_c + 1, 1, d)$. Intuitively, we can see that the width of output is compressed to 1, which means information loss.

3.3. Channel Transformation Based Convolutional Layer. To reduce information loss and take full advantage of information in depth dimension, namely, the channel information, we propose a method called channel transformation. Channel transformation is a reshape operation of a vector. For example, the shape of the convolution output is $(n - h_c + 1, 1, d)$, which can be transformed to $(n - h_c + 1, d, 1)$ by switch width and depth dimensions. Channel transformation provides two benefits. One is that the output of the previous convolutional can be remained "image" shape such that we can stack multiple convolutional layers. The other benefit is that we can make good use of information in depth dimension and extract feature maps with more sentimental semantics.

3.4. The Three CTB Convolutional Layers. In CTBCNN model, we have three CTB convolutional layers. Figure 3 shows the structure of the three channel transformation based convolutional layers.

In the first convolutional layer (conv1), the shape of the input vector is $(n, k, 1)$. Similar to Kim's work, we use 3 different sizes of filters.

$$\begin{aligned} &W_{11} \epsilon R^{h_{c11} * k}, \quad where\ h_{c11} = 3 \\ &W_{12} \epsilon R^{h_{c12} * k}, \quad where\ h_{c12} = 4 \\ &W_{13} \epsilon R^{h_{c13} * k}, \quad where\ h_{c13} = 5 \end{aligned} \tag{4}$$

Each size is with n_1 filters and extract n_1 feature maps. For filter W_{1i}, the shape of convolution output is $(n - h_{c1i} + 1, 1, n_1)$. The trick of channel transformation is used to switch width and depth dimensions such that we have the new shape $(n - h_{c1i} + 1,\ n_1, 1)$ as input for the next convolution layer.

In the second convolutional layer (conv2), we use 3 different filter sizes:

$$\begin{aligned} &W_{21} \epsilon R^{h_{c21} \times 1}, \quad where\ hc_{21} = n - h_{c11} + 1 \\ &W_{22} \epsilon R^{h_{c22} \times 1}, \quad where\ hc_{22} = n - h_{c12} + 1 \\ &W_{23} \epsilon R^{h_{c23} \times 1}, \quad where\ hc_{23} = n - h_{c13} + 1 \end{aligned} \tag{5}$$

Each size is with n_2 filters and extract n_2 feature maps. For filter W_{2i}, the shape of convolution output is $(1, n_1, n_2)$. The trick of channel transformation is used to switch height and depth dimensions to get the new shape $(n_2,\ n_1, 1)$. Finally we concatenate the output from 3 different sizes of filters on depth dimension and the shape of output becomes $(n_2,\ n_1, 3)$, which can be seen as an image of size $n_2 \times n_1$ with 3 channels.

In the third convolutional layer (conv3), we use filter $W_3 \epsilon R^{h_{c3} \times h_{c3}}$ to implement wide convolution, with n_3 filters and extract n_3 feature maps. Notice that the numbers of feature maps n_3 need to be the same as the output labels of the whole model. These feature maps contain sentimental semantics of the input sentence, which will be sent to global average pooling layer to capture the most important feature.

4. Global Average Pooling Layer

In classic CNN, feature maps produced by convolution layers are usually sent to max pooling layer to downsampling and then concatenated as a long vector, which is fully connected to output categories. The dense connection makes it hard to interpret how the category level information from the objective cost layer feed back to the convolution layer. Furthermore, the fully connected layers are prone to overfitting and heavily depend on dropout regularization.

For the reasons mentioned above, our method replaces the fully connected layer with global average pooling layer. Global average pooling layer enforces direct correspondences between feature maps and categories. In this way the feature maps can be intuitively interpreted as categories confidence maps. Furthermore, there is no parameter to optimize in global average pooling layer; thus overfitting is avoided.

From the comparison in Figure 4 we can see that global average pooling layer takes the average value of each feature map which can be regarded as confidence value for each category, and the resulting vector is fed directly into the Softmax function to get the probability distribution among sentiment categories.

$$P\left(y_j \mid x, \theta\right) = \frac{\exp\left(S_j\left(x, \theta\right)\right)}{\sum_{1 \le i \le |Y|} \exp\left(S_i\left(x, \theta\right)\right)} \quad 1 \le j \le |Y| \tag{6}$$

where θ represents the model parameter set. $S_j(x, \theta)$ is the average pooling result of the feature map that corresponds to category j. Y is the category space. We use stochastic gradient

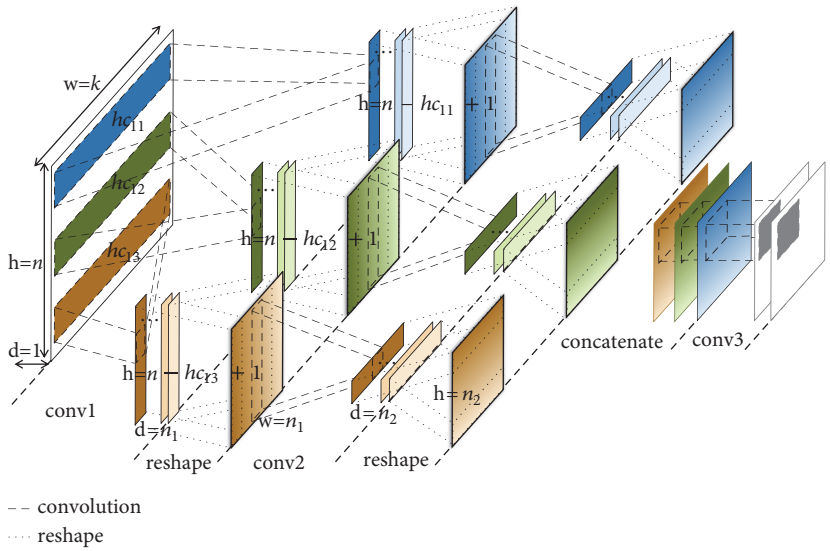
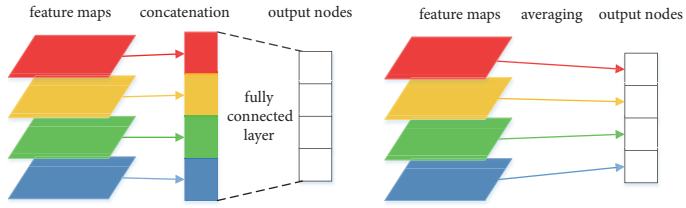


Figure 3: The structure of three channel transformation based convolutional layers. The first reshape switches the width and depth dimensions of the convolutional output. The second reshape switches the height and depth dimensions of the convolutional output.



(a) Fully connected layer (b) Global average pooling

Figure 4: The comparison between fully connected layer and global average pooling layer. Global average pooling layer feeds each feature map to corresponding category.

descent to minimize the Negative log-likelihood function of formula (6) and learn the parameter set.

5. Experiments

5.1. Overview. We have two parts of experiments. The first one performed Chinese sentiment classification tasks using CTBCNN, compared with other typical machine leaning methods and classic textCNN [15]. The second experiment focused on the regularization effect of global average pooling layer of CTBCNN model. At last we analyse the portability of our model and how it can be conveniently used in different cases.

All experiments are evaluated on two datasets: THU Hotel reviews (http://nlp.csai.tsinghua.edu.cn/~lj/.) used in [17] and ChnSentiCorp-Book (http://www.nlpir.org/?action-viewnews-itemid-77) used in [18]. Table 1 shows the details of these two datasets after duplication removal work. Both datasets are roughly equally split into positive and negative. Since there is no standard set for these dataset, we performed 10-fold cross validation and used the average accuracy metric to measure the overall performance for each method.

Initialized word vectors are 400-dimensional that were trained on 230000 articles from Chinese Wikipedia (https://dumps.wikimedia.org/zhwiki/latest/) using the skip-gram architecture. For model hyperparameters, we use 100 filters for conv1 and 100 filters for conv2. In conv3, the filter size is 3×3 with 2 filters which is in accordance with the number of categories.

Table 1: Summary of two datasets.

Dataset	Positive	Negative
THU Hotel reviews	7678	7811
ChnSentiCorp-Book	1967	1967

5.2. Sentiment Classification. To evaluate the performance of CTBCNN model, we experimented with several typical methods. All methods are experimented on two datasets with the same static word vectors. Results of our method against other methods are shown in Table 2.

The first line of Table 2 gives the results of lexicon-based baseline method used in [18, 19]. Experiment results show that most machine learning methods surpass the baseline method on both datasets. It is worth noticing that book reviews contain more less frequently used words and

Table 2: the classification results on two datasets.

Methods	Accuracy (%)	
	THU Hotel reviews	ChnSentiCorp-Book
Baseline(Lexicon-based)	80.00	71.30
CTBCNN	**92.43**	**92.81**
TextCNN	91.33	90.97
Naïve Bayes	75.26	83.20
Random Forest	80.68	84.98
SVM	89.11	86.63

Table 3: Global average pooling compared to fully connected layer.

Model	Accuracy (%)	
	THU Hotel reviews	ChnSentiCorp-Book
CTBCNN-FC without dropout	91.04	92.45
CTBCNN-FC with dropout	91.74	92.65
CTBCNN-GAP	**92.43**	**92.81**

expressions, which results in low classification accuracy. SVM outperformed other typical machine learning method, which shows that SVM is capable of handling high-dimensional features. Classic TextCNN achieves better performance than most machine learning method because of the powerful feature extraction ability of convolution layer, while CTBCNN surpasses all the other methods, which suggests that the CTB convolution layer can extract more representative feature than plain convolution layer.

5.3. The Regularization Effect of Global Average Pooling Layer. To evaluate the regularization effect of global average pooling layer, we set up a comparison experiment which replaces the global average pooling (GAP) layer of CTBCNN with a fully connected (FC) layer, while the other parts remain the same. We evaluate this model with and without dropout before the FC layer. All models are tested on the two datasets and the results are shown in Table 3.

As we can see in Table 3, for both datasets, the model with fully connected layer without dropout gives the worst performance, which is expected as the fully connected layer prone to overfitting [20] without applying any regularizer. The second model applies dropout before the fully connected layer and achieves better performance than the first one, while our model with global average pooling layer achieves the highest classification accuracy, which proved global average pooling is an effective way to avoid overfitting.

5.4. Portability of CTBCNN. From the above discussion we know that CTBCNN avoid human engineering by using CTB convolutional layer, which means CTBCNN is non-feature-specific and non-task-specific. This is helpful when different node handles different types of data in a distributed computing network. Besides, CTBCNN can be applied for multiclass classification tasks conveniently. All we need to do is make sure the number of the output feature maps of the last convolutional layer is equal to the number of categories.

6. Conclusion

Fog computing uses edge nodes to carry out a substantial amount of data analysis work. In public opinion analysis system, it is crucial to find suitable algorithm that is lightweight in operation and accurate in prediction. This work focuses on Chinese sentiment analysis in fog computing environment and proposes a non-task-specific method called Channel Transformation Based Convolutional Neural Network (CTBCNN). CTBCNN mainly consists of two parts: the three CTB convolutional layers and the global average pooling layer. CTB convolution layer is able to cover more channel information and extract more representative feature maps. And global average pooling is a regularizer to prevent overfitting. Through experiments and analysis, we show that our model do achieve competitive accuracy and it is convenient to apply this method to different cases in operation.

Data Availability

The "ChnSentiCorp-Book" data (also called "ChnSentiCorp-BK-ba-4000") used to support the findings of this study were supplied by Tan Songbo under license and so cannot be made freely available. Requests for access to these data should be made to Tan Songbo, tansongbo@software.ict.ac.cn. The "THU Hotel reviews" data used to support the findings of this study can be accessed from http://nlp.csai.tsinghua.edu.cn/~lj/.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Key R&D Program of China, Research and Application of Key Technologies

for Information Security Certification, under Grant no. 2016YFF0204001 of China Information Security Certification Center.

References

[1] D. C. Segura, R. de Souza Stabile, S. M. Bruschi, and P. S. Souza, "Providing computing services through mobile devices in a collaborative way - a fog computing case study," in *Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems*, pp. 117–121, ACM, Miami, FL, USA, November 2017.

[2] S. Park and Y. Yoo, "Network intelligence based on network state information for connected vehicles utilizing fog computing," *Mobile Information Systems*, vol. 2017, Article ID 7479267, 9 pages, 2017.

[3] D. Roca, R. Milito, M. Nemirovsky, and M. Valero, "Tackling IoT Ultra Large Scale Systems: fog computing in support of hierarchical emergent behaviors," in *Fog Computing in the Internet of Things*, pp. 33–48, Springer, Cham, Switzerland, 2018.

[4] S. Zhang, Z. Wei, Y. Wang, and T. Liao, "Sentiment analysis of Chinese micro-blog text based on extended sentiment dictionary," *Future Generation Computer Systems*, vol. 81, pp. 395–403, 2018.

[5] W. Xing, L. Haitao, and Z. Shaojian, "Sentiment analysis for Chinese text based on emotion degree lexicon and cognitive theories," *Journal of Shanghai Jiaotong University (Science)*, vol. 20, no. 1, pp. 1–6, 2015.

[6] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing-Volume 10*, pp. 79–86, Association for Computational Linguistics, Stroudsburg, Pa, USA, July 2002.

[7] L. Shouhan, X. Rui, Z. Chengqing, and H. Churen, "A framework of feature selection methods for text categorization," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pp. 692–700, Association for Computational Linguistics, 2009.

[8] W. Hongwei, Y. Pei, Y. Jiani, and L. JNK, "Text feature selection for sentiment classification of Chinese online reviews," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 25, no. 4, pp. 425–439, 2013.

[9] A. Severyn and A. Moschitti, "Twitter Sentiment Analysis with deep convolutional neural networks," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM, August 2015.

[10] L. Himabindu, R. Socher, and C. Manning, "Aspect specific sentiment analysis using hierarchical deep learning," *NIPS Workshop on Deep Learning and Representation Learning*, 2014.

[11] C. N. Dos Santos and M. Gatti, "Deep convolutional neural networks for sentiment analysis of short texts," in *Proceedings of the 25th International Conference on Computational Linguistics (COLING '14)*, 2014.

[12] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," https://arxiv.org/abs/1301.3781, 2013.

[13] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the 27th Annual Conference on Neural Information Processing Systems (NIPS '13)*, pp. 3111–3119, December 2013.

[14] L. Min, C. Qiang, and Y. Shuicheng, "Network in network," https://arxiv.org/abs/1312.4400, 2013.

[15] Y. Kim, "Convolutional neural networks for sentence classification," https://arxiv.org/abs/1408.5882, 2014.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.

[17] L. Jun and S. Maosong, "Experimental study on sentiment classification of Chinese review using machine learning techniques," in *Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering, IEEE NLP-KE 2007*, pp. 393–400, Beijing, China, September 2007.

[18] Y. Zhang, X. Xiang, C. Yin, and L. Shang, "Parallel sentiment polarity classification method with substring feature reduction," in *Trends and Applications in Knowledge Discovery and Data Mining*, J. Li et al., Ed., vol. 7867 of *Lecture Notes in Computer Science*, pp. 121–132, Springer, Berlin, Germany, 2013.

[19] P. Zhang, Z. He, and L. Tao, "Compositional polarity classification approach for product reviews," in *Proceedings of the 2014 7th IEEE Joint International Information Technology and Artificial Intelligence Conference, ITAIC 2014*, pp. 58–62, Chongqing, China, December 2014.

[20] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 1398958, 11 pages
https://doi.org/10.1155/2018/1398958

Research Article

An Engagement Model Based on User Interest and QoS in Video Streaming Systems

Xiaoying Tan,[1] Yuchun Guo ,[1] Mehmet A. Orgun,[2] Liyin Xue,[3] and Yishuai Chen[1]

[1]*Beijing Jiaotong University, China*
[2]*Macquarie University, Australia*
[3]*Australian Taxation Office, Sydney, NSW 2000, Australia*

Correspondence should be addressed to Yuchun Guo; ychguo@bjtu.edu.cn

Received 4 May 2018; Accepted 4 July 2018; Published 23 September 2018

Academic Editor: Lei Yang

Copyright © 2018 Xiaoying Tan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the surging demand on high-quality mobile video services and the unabated development of new network technology, including fog computing, there is a need for a generalized quality of user experience (QoE) model that could provide insight for various network optimization designs. A good QoE, especially when measured as engagement, is an important optimization goal for investors and advertisers. Therefore, many works have focused on understanding how the factors, especially quality of service (QoS) factors, impact user engagement. However, the divergence of user interest is usually ignored or deliberatively decoupled from QoS and/or other objective factors. With an increasing trend towards personalization applications, it is necessary as well as feasible to consider user interest to satisfy aesthetic and personal needs of users when optimizing user engagement. We first propose an *Extraction-Inference (E-I)* algorithm to estimate the user interest from easily obtained user behaviors. Based on our empirical analysis on a large-scale dataset, we then build a *QoS and user Interest based Engagement (QI-E) regression model*. Through experiments on our dataset, we demonstrate that the proposed model reaches an improvement in accuracy by 9.99% over the baseline model which only considers QoS factors. The proposed model has potential for designing QoE-oriented scheduling strategies in various network scenarios, especially in the fog computing context.

1. Introduction

The past two decades have witnessed the growth and popularity of Video-on-Demand (VoD) applications on both PCs and mobile devices, and the trend is moving from basic video offering toward better quality of user experiences (QoE) in both industry and academia [1]. In video streaming services, the traditional Mean Opinion Score (MOS) is now replaced by *user engagement*, which more directly impacts the return for investment from stakeholders of a network ecosystem [2, 3].

The requirement of a good QoE for optimizing returns has led to a rapid development of network services and the emergence of new technology, including fog computing which could achieve real-time processing and feedback of high-volume video streaming and scalability of service on low-bandwidth output data [4].

As the first step to optimize user engagement, there is an urgent need for a general engagement model that could provide insight for diverse network environments, especially for the up-to-date fog computing context. There are several studies on understanding and modeling engagement in prior works where the quality of service (QoS) together with some objective context factors, such as location, device, and temporal attributes, have always been regarded as the fundamental factors related to engagement. The relationship of user engagement and QoS, either at the application level (e.g., *startup delay, buffer frequency*, and *bitrate*) or at the network level (e.g., *throughout, signal strength*) has been widely explored in related works.

However, most of the related works ignored another important factor-subjective human factors, such as users' personalized interest in a specific video. In fact, the new era of customization raises the aesthetic and personal needs

of users, especially in mobile video services where users typically present predictable features and service demand [5]. Accordingly, user engagement, as a reflection of "the degree of delight or annoyance of the user of an application or service [6]", cannot be satisfied only by high-quality delivery. Thus, sessions with the same buffer frequency sometimes have different user engagement due to the divergence of user interest in the video content. However, most of the prior works ignore or exclude the impact of the subjective factors as it is difficult to quantify them. Some works consider video popularity as a subjective human factor in QoE models [2, 7, 8], but video popularity could only roughly describe users' average preferences but without any personalization. Only a few works [9–11] at a macrolevel evaluate subjective human factors for each viewing session from psychological and cognitive perspectives. In these works, the subjective factors are obtained by extensive experiments and surveys with a very large population of subjects; however, such experiments and survey are expensive to conduct and not suitable for streaming applications in VoD systems.

An engagement model based on both *QoS* and *user interest* is not only necessary for accurately understanding and predicting user engagement, but also beneficial for optimizing system resource allocation and for providing better personalized services. On the one hand, finding out how the subjective factor impacts user engagement could help designers to deploy appropriate bandwidth resources for optimizing user engagement. On the other hand, recommendation system (RS) could make a tradeoff decision between the QoS factors and the human factors to recommend to users the videos that they are interested in and also that have good QoS and finally that they can enjoy for a longer time. Hence, it is critical to understand the relationship between engagement and the human factors as well as QoS in order to shed light on how best to allocate resources and customize services.

A challenge prior to building such a model is how to quantify the degree of a user's interest in a video. In prior recommender systems, user interest is measured either to be explicit ratings by user study or roughly to be implicit ratings, e.g., user engagement. However, the former measurement method is accuracy but is impossible to be collected in time-sensitive applications. Instead, the later one is inaccurate, as user engagement is sometimes not the reaction to his/her pure interest in the video but also impacted by other factors, e.g., quality issues like *startup delay*.

Another challenge is how to characterize the relationships between user engagement and the two factors in a unified model to provide insight for practical applications. Intuitively, the two factors impact user engagement but not independently, which is beyond the scope of a linear regression model. For example, user interest not only impacts user engagement itself but also impacts users' patience with the QoS problem. Machine learning (ML) algorithms, e.g., decision trees and Naive Bayes [2, 12], can characterize such a dependent relationship but not in a concise formula and therefore could not provide clear insight for practical applications.

We devote this paper to respond to these two challenges. We propose an *Extraction-Inference (E-I)* algorithm to estimate the user interest from easily obtained user behaviors. Through a measurement on a real-world VoD system, we analyze the relationship of user engagement with both QoS and user interest. Based on our empirical analysis on a large-scale dataset, we build a***Q****oS and user* ***I****nterest based* ***E****ngagement (QI-E) regression model*. Our empirical evaluation shows that the incorporation of human factors brings an improvement of 9.99% in prediction accuracy over the baseline model based on only QoS factors. Finally, we discuss the application potential of the proposed model and the future work.

2. Related Work

2.1. MOS versus Engagement Metrics. User engagement instead of Mean Opinion Score (MOS) is widely employed as QoE metric in streaming applications. As a standard metric in the ITU-T recommendations, MOS is a numeric value from 1 to 5 (i.e., poor to excellent) obtained through user studies or surveys [13]. There are many analyses on its influential factors in the domain of web services, E-commerce, multimedia, and so on [7, 13, 14]. However, since conducting a survey or user study is expensive, time- consuming, and without repeatability, MOS cannot be directly used in video streaming applications [13], especially in the fog computing context where the real-time response (real-time sensing and data processing) is required [5]. These limitations motivate the development of objective metrics, such as user engagement which quantifies a user's behavior as the reaction to the level of QoE. In video applications, user engagement is usually measured as session length [8, 15], abandonment rate [12, 15], number of visits [2], or skip rate [12]. These user behavior metrics are more directly relevant to increasing opportunities for advertising and upselling, leading to greater revenues [8].

2.2. QoS Parameters. QoS parameters are often studied as primary factors related to QoE since they can be controlled by the platform at least partially [9]. The specific QoS metrics vary across different domains [7], including the metrics on the application level, e.g., startup delay, buffer frequency, buffer ratio, and bitrate [2, 8, 16] and also including the metrics on the network level, e.g., flow throughput, flow duration, handover rate, and signal strength [12, 15]. Due to uncertainty of network conditions, QoS metrics on the application level capture the quality perceived by the users more closely than those on the network level.

Sometimes QoS metrics compete with or conflict each other and require a tradeoff in system design. For example, prior works [17, 18] point out the competing relationship between the initial time and the buffer event. Moreover, the tradeoff between the bitrate and the buffer event is often studied in bitrate adaptation schemes [17].

There are also several techniques proposed to adjust QoS parameters at the client, at the server, or in the network [19]. Those techniques include bitrate adaptation [20], prefetching [21, 22], transport protocol selection [14], and cache deployment [23].

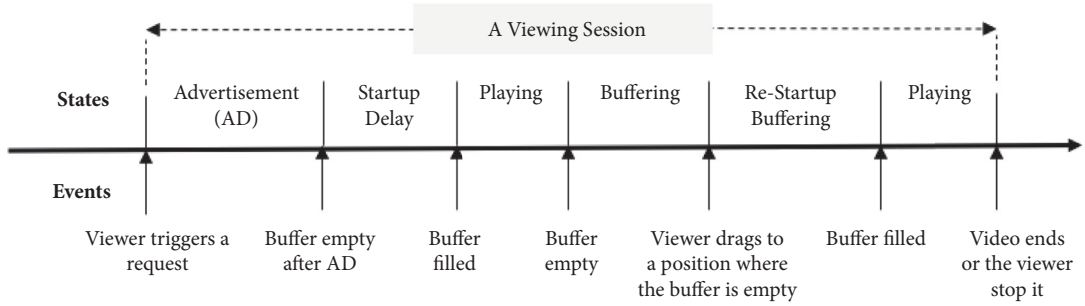


Figure 1: Illustration of a typical video streaming session.

2.3. Other Influential Factors. Recently, it has been realized that QoS factors alone cannot determine users' satisfaction and other potential "confounding" factors need to be explored [7]. The considered factors can be classified into context information such as connectivity and temporal effects [2], content attribute such as types and popularity [2], user attribute such as location [24], device, and gender [8].

However, user interests are often ignored. A few works addressed human factors (similar with user interest defined in this paper in essence) [9, 10] from psychological and cognitive perspectives and some attempted to integrate all kinds of factors including human ones [11, 25]. However, conducting such works is expensive, requiring a long period and the participation of both technicians and psychologists [9]. Identifying human subjective factors which are suitable for real-time streaming applications is still an open problem.

In terms of understanding the impact of human factors (e.g., user interest) on system design, one study [26] exploits individual interest to optimize the storage resource allocation in CDN. However, this study does not consider quality factors and hence it cannot help making a tradeoff decision on user interest and QoS.

2.4. Predicting User Interest. Predicting a user's interest in a specific item is the target problem addressed in a personalized recommendation system (RS), which is a hot topic in the face of information overload over the last decade [27, 28]. As one popular kind of recommendation algorithms, Collaborative Filtering (CF) algorithms, such as K-Nearest Neighbor, Bayesian belief nets, and Matrix Factorization, have been widely studied [27, 29–34]. As CF algorithms do not need data referring to item content, they are applicable to the video systems where explicit content descriptions of items (i.e., videos) are difficult to be obtained.

In addition, how to obtain and quantify user interest is also under study [35]. Most works ask users for explicit ratings after they purchase items, watch videos, or browse website. This method is accurate but not practical in time-sensitive applications like online VoD systems. Instead, user behaviors, such as the time spent on a page, scrolling and clicks on web pages [36, 37], time spent on a video [38, 39], and purchases in the past [40], are used as implicit ratings in some applications. User behaviors are sometimes capable of serving as reliable implicit ratings as proved in [41], but in VoD systems they are still quite noisy [42]. For example, as we address in this paper, user behavior depends on not only user interest but also the QoS during the watching time. As far as we know, such noise in implicit rating has not yet been considered in existing VoD recommendation systems.

3. Problem and Definition

This section defines the scope of the problem we focus on and then provides the definition of the metrics of the factors considered in the target model.

3.1. Problem Statement. The main objective in this paper is to propose a practical engagement model based on the objective QoS factors and the subjective individual interest factor in VoD systems. For clarity, we omit the impact of other confounding factors, such as the type of video, device, and temporal attributes.

Thus, the objective is to build a model expressed as

$$E = f(\mathbf{Q}, r) \tag{1}$$

where r, E, and $Q^k \in \mathbf{Q}$, respectively, denote user interest level, user engagement, and the value of the k-th QoS metric in a session for a user w. To make the model clear for practical applications, the dependency function $f(\bullet)$ should be in a clear and concise form.

3.2. Metrics Definition. This section defines the metrics of the factors in our model. To clarify the definitions, we first introduce the terminologies of a viewing session by Figure 1.

A typical viewing session starts when a user initiates a request and ends when he/she finishes viewing the content or changes to another one or closes the client agent. At the very beginning, most users may experience a period of advertisement and sometimes a startup delay. During viewing, users may suffer from a frozen delay due to congestion or speed limit in network. Also, users can trigger a pause or drag the process bar to a new position. If the user drags to a new position where the content has not been prefetched already, he/she may experience a restartup delay.

According to the definition of a viewing session, we now define the metrics of three factors.

3.2.1. The Metric of Engagement. We define user engagement as the *valid watching ratio* in this paper, which measures how much the video has really played. It is computed as the ratio of the playback time to the video length. Here, the playback time is referred to as the time of a session excluding startup delay, restartup delay, buffering, pause, and advertisement time. In general, the range of engagement value is restricted within 0% to 120%.

3.2.2. The Metric of QoS. In this paper, we consider the application-level QoS metrics which capture delivery-related effects on the client-side. Specifically, we focus on the following metrics.

Startup Delay. It is the time before a video starts playing and immediately after the user initiates a request, exclusive of the time taken by advertisements. It is measured in seconds.

Buffer Frequency. It is the ratio of the number of buffering events to the total time of buffering and playing. It is measured in the number of times per minute.

Buffering Ratio. It is the ratio of the time spent on buffering or restartup buffering occurring in a session to the total time of buffering and playing. It is measured in percentage.

As an extension of the study in [2], we also consider another new quality metric, *average buffer length*.

Average Buffer Length. It is the average time the user has to spend in buffering once a buffer event occurs. It is computed as the ratio of the *buffering ratio* to the *buffering frequency*. It is a new metric we propose to complement the buffer frequency.

We do not discriminate between buffering and restartup buffering event and do not discuss another usually-used quality metric, *bitrate*, due to data absence. But this omission does not lessen the value of our study because the point here is not to study the relationship between quality metrics, but to address the competing or conflicting relationships between quality and user interest. If necessary or once the data is available, the model can be extended.

Compared with network-level QoS metrics, the metrics at application level are more generally applicable in diverse network contexts. They can manage the application-level QoS metrics through respective technology, such as service selection in content delivery network (CDN) [2], prepushing scheme in P2P network [43, 44], and caching technique in fog computing [4, 45, 46], and then optimize user engagement based on the application-level QoS metrics involved in the QoE model.

3.2.3. The Metric of User Interest. User interest is a user's subjective sense of concerning with and curiosity about the content of a specific video. We regard user interest to be a nondimensional parameter between 1 to 10. As discussed earlier in this paper, it is difficult to ask users for explicit ratings and directly using implicit ratings measured by user behaviors may lead to inaccuracies. Thus, we propose an inference algorithm to obtain user interest, which will be depicted in Section 4.

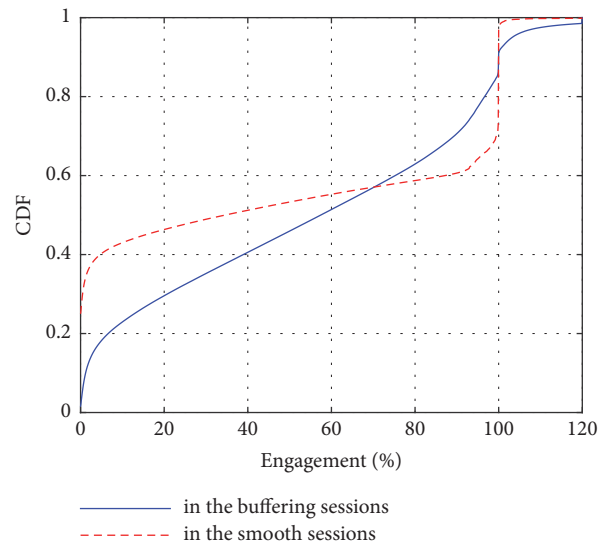


Figure 2: The distributions of user engagement in the buffering sessions and that in the smooth sessions. For clarity, we do not present the distribution of the engagement larger than 120% that are in the tiny minority.

3.3. Dataset. To build a data-driven model, we collect a large-scale dataset from the client-end of PPTV [47], a typical commercial P2P streaming system in China. The anonymous user logs range from 23th March to 28th March in 2011 with 75 million requests covering over 130,000 unique videos and 6 million users. The logs record viewing sessions of all the behavior-related, quality-related information for each user-item pair.

To filter the impact of other confounding factors (e.g., device, temporal attribute and the type of video), we only consider the sessions collected from client-end, during the periods of 7 p.m. to 12 p.m. and related to the videos of movie type in our analysis. But the method proposed here is not limited to these contexts and can be extended to other context, e.g., mobile device or other video types.

4. User Interest Inference

In this section, through the analysis of our dataset, we first show that user engagement is impacted not only by user interest but also by QoS. This means mapping the engagement time into user interest level is inaccurate. We then develop an***Extraction-Inference (E-I)*** interest estimation algorithm and evaluate it on our dataset.

4.1. Measurement and Analysis. We compare the distribution of user engagement in the sessions with/without buffering events or startup delay (called *buffering sessions*/*smooth sessions,* respectively).

As the plots given in Figure 2 show, there are significant gaps, which show that the engagement alone fails to describe a

Step I (extraction). Extract users' interest from their engagement records in the **Smooth** sessions where users did not experience any quality problems including buffering event or start-up delay.
Step II (inference). Based on the extracted interest records in Step I, infer users' interest in other sessions.

Algorithm 1: **E**xtraction and **I**nference (E-I) interest estimation algorithm.

Table 1: Inference methods of Baseline II.

Gmean	the global average of all users' interest levels in the training sessions.
Uavg/*Iavg*	the average of the interest level of the active user/the active video.
UIavg	the geometric mean of the active user's and the active video's average interest levels.
KNN	the average of the interest of the active user's K nearest neighbors who have the most similar interest in other videos with her.

user's interest accurately, as user engagement is also impacted by quality problems. As shown, the buffering event makes the distribution of the engagement more even in the entire range. Without experiencing any buffering events, most users either finish watching the entire video (accounting for 32% of the sessions) or abandon the session quite early before 5% of the entire video is viewed (for 38%); on the contrary, in the buffering sessions, these two extreme cases account for below 22% together.

The gap of the distributions in Figure 2 is attributed to the quality problems that, on the one hand, impair users' watching experience and reduce the probability of the long engagement time and, on the other hand, indicate that the users still have a certain interest in the videos rather than abandoning the sessions at the beginning even before any buffering events.

4.2. Extraction-Inference (E-I) Algorithm. We propose a heuristic***Extraction and Inference (E-I)*** *interest estimation* algorithm based on the following two assumptions: (1) given the QoS and the seeking state, a user's engagement time only depends on her interest level in the video and (2) a user's preference remains consistent during a short period, as widely accepted in recommendation systems. We develop this algorithm in two steps as shown in Algorithm 1.

First, in the selected sessions, QoS and the seeking state are determined; i.e., all the values of the QoS metrics and the seeking state are equal to zero. Accordingly, the engagement time in these sessions is decided by users' interest. We uniformly map user engagement in smooth sessions into 10 bins referred to the user's interest implicit rating (where 1 represents poor and 10 excellent).

Based on the collected users' interest in the selected sessions, we next use Matrix Factorization (MF) [33, 34], a typical Collaborative Filtering (CF) algorithm [35–38], to infer their interest in other sessions. Compared with some other typical CF algorithms, e.g., KNN algorithm [35, 36], the MF algorithm is better at dealing with the data sparsity [36] and, in our experiments, the data used for training is quite sparse. The selected sessions (used for both training and testing) only account for 23% of the sessions in our dataset.

The MF algorithm supposes that users' interest can be explained by characterizing both the users and the videos to a joint latent factor space. In this space, users and videos should be represented to be M-dimensional latent factor vectors. For a user u, provided with her vector p_u and a video v's vector q_v, her interest in this video $\hat{r}_{u,v}$ can be predicted to be the product of these two vectors in addition to the global average interest μ, the user's bias b_u, and the video's bias b_v. That is, $\hat{r}_{u,v} = \mu + b_u + b_v + p_u^T q_v$. In practice, the latent factor vectors, p_u and q_v, and the biases, b_u and b_v, are learned by the stochastic gradient descent (SGD) algorithm [14] looping through the training dataset to minimize a utility function; i.e., $\min_{b_*,q_*p_*} \sum_{\tilde{r}_{u,v}\in\tilde{I}} (\tilde{r}_{u,v} - \mu - b_u - b_v - p_u^T q_v)^2 + \lambda(\|p_u\|^2 + \|q_v\|^2 + \|b_u\|^2 + \|b_v\|^2)$. Here $\tilde{I}$ denotes the set of the historical interest records.

4.3. Evaluation. We exploit the data in smooth sessions for training and evaluation. Via 10-fold cross validation, we fix the parameters, λ and M, in the E-I algorithm to be 0.05 and 30, respectively.

For comparisons, we propose two types of baseline methods. The first type (namely, Baseline I) modifies Step I of the E-I algorithm. It directly maps user engagement into user interest in all the sessions (regardless of the quality problems). The second type (namely, Baseline II) replaces the inference algorithm, the MF algorithm, with some statistical methods and another CF algorithm, K-Nearest Neighbors (KNN), as defined in Table 1.

We evaluate the engagement prediction accuracy (measured by Root Mean Square Error (RMSE) [39]) of the algorithms on the "selected" sessions of the test dataset where the engagement records do represent users' pure interest.

According to the results listed in Table 2, E-I algorithm improves the engagement prediction accuracy by 18% on Baseline I, 27% on Gmean, 7.26% on Uavg, 8.9% on Iavg, 6.3% on UIavg, and 4.8% on KNN. The positive results of our proposed E-I algorithm confirm that it is necessary to eliminate the effect of the other relevant factors when we extract user interest from their engagement records.

5. Engagement Predictive Model

In this section, we measure the relationships of user engagement with user interest and QoS, respectively, and then propose the engagement predictive model.

Table 2: Estimation accuracy of user interest inference models and the improvement achieved by E-I algorithm on the baseline methods. The evaluation is conducted in the smooth sessions.

Methods	Baseline I	Baseline II					E-I algorithm
		Gmean	Uavg	Iavg	UIavg	KNN	
RMSE	3.63	4.07	3.19	3.25	3.16	3.11	2.96
Improvement (%)	18.4	27.2	7.2	8.9	6.3	4.8	- -

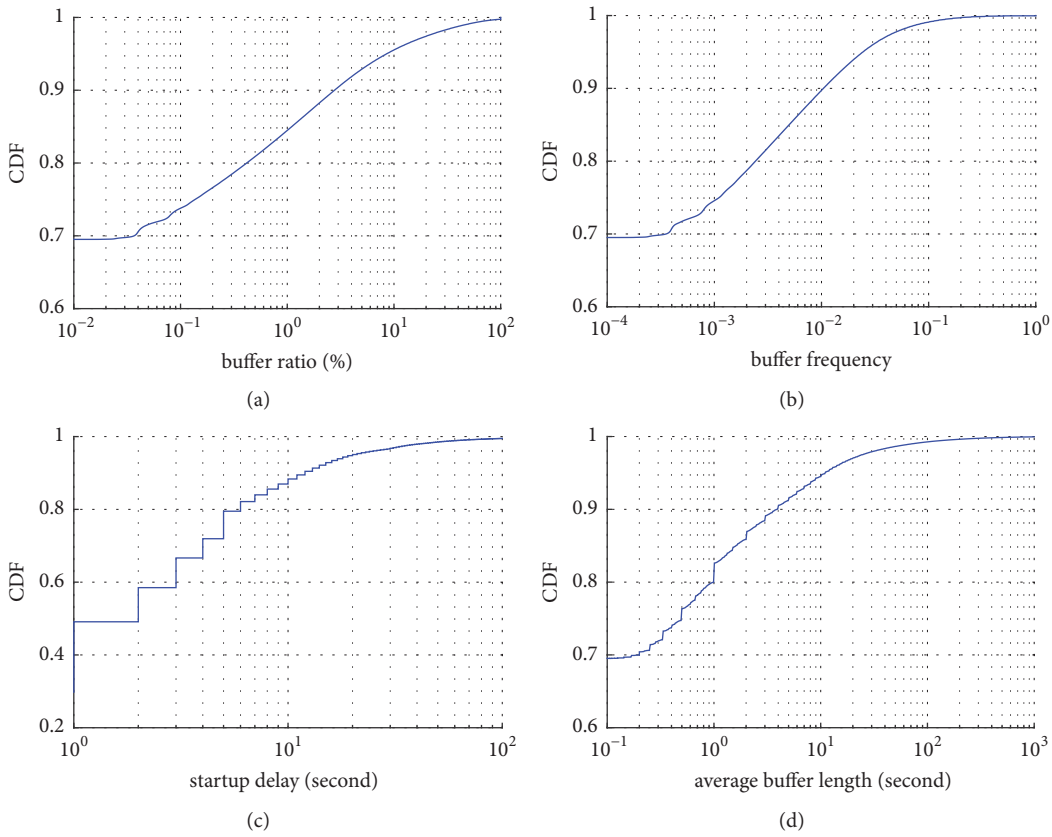


Figure 3: Cumulative distribution plots for QoS metrics.

5.1. User Engagement versus QoS Metrics

5.1.1. QoS Distribution. We first look at the distributions of the values of the various QoS metrics. As shown in Figure 3, we find that the system has a good quality generally although the quality problems are not trivial in some sessions. 43.5% of the sessions in our dataset have not endured any quality problems. Specifically, 70% of the sessions have not endured a buffer event and 50% have not experienced *startup delay*. Still, there are some sessions having endured quite poor quality situations. For example, 5% of the sessions endure a buffering ratio over 10%. 5% of the sessions have more than 5 buffering events in their 100-second playing time.

The generally good quality situation restrains the usage range of the traditional engagement models that consider QoS only. These models could not differentiate the sessions without quality problems, although the QoS metrics should be taken into account as relevant factors as shown earlier.

5.1.2. Correlation Analysis. Next, we examine the expectations of user engagement conditional on various quality metrics. Given a quality value $Q = q$, the conditional expectation of user engagement is calculated to be $\mathbb{E}(E \mid Q = q) = \sum_{e \in D(E)} eP(E = e \mid Q = q)$, where $P(E = e \mid Q = q)$ is the conditional probability and $D(E)$ is the range of the engagement value as defined in Section 3.2.

The plots of the conditional expectations are shown in Figure 4. in the dominant range of *buffer frequency* ([0, 18%] as shown in Figure 3(b)), as an example, the engagement generally decreases log-linearly as the quality becomes poorer as expected, as shown in Figure 4(b), which means that the decrease rate slows down with a larger buffer frequency. This result supports the intuition that users are less sensitive to a worse quality situation once they have suffered a bad one. The relationship between user engagement and the buffering ratio shows the similar characteristics. But with a larger average buffering length or startup delay, as shown in Figures 4(d)

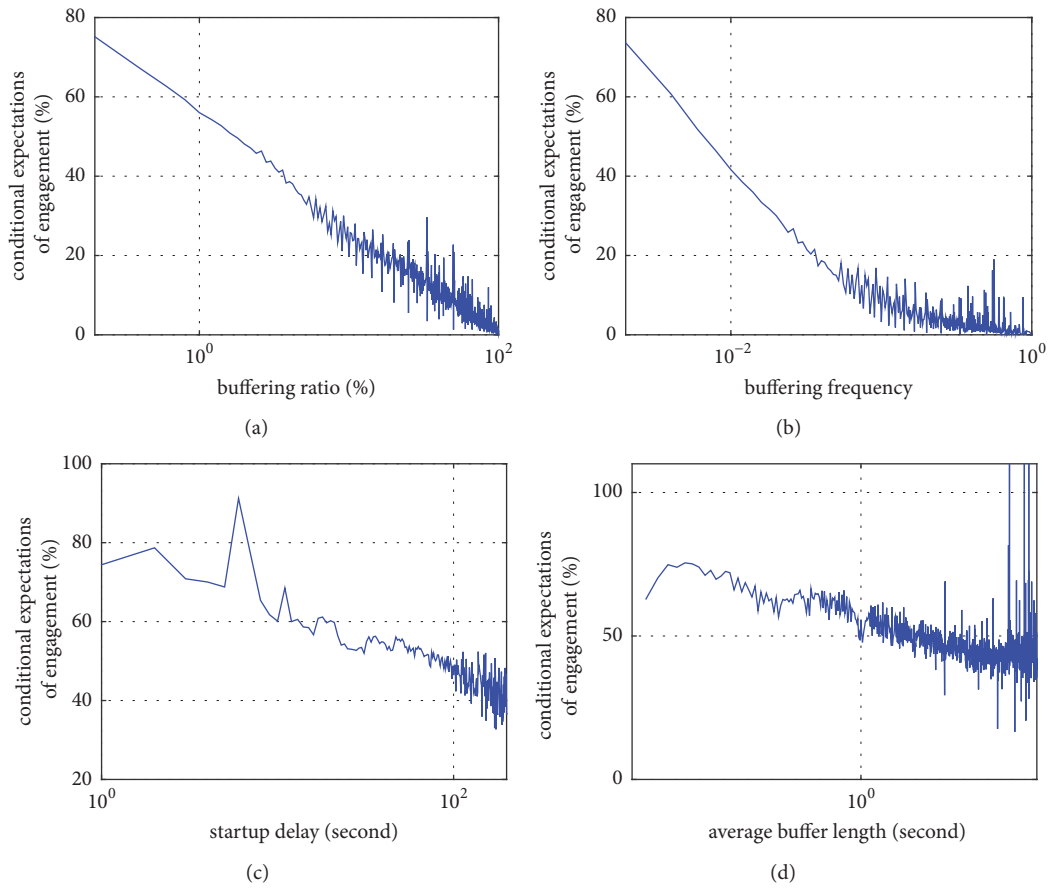


Figure 4: Expectations of user engagement against diverse QoS metrics. (a) Buffering ratio, (b) buffering frequency, (c) startup delay, and (d) average buffer length.

Table 3: Correlation coefficients of user engagement with QoS and user interest, respectively. The values of the QoS metrics are measured in logarithmic scale. ∗The correlation coefficient of user engagement with user interest will be introduced in Section 5.2, while those with QoS metrics are introduced in Section 5.1.

Correlation metric	Buffer ratio	Buffer frequency	Average buffer length	Startup delay	Interest∗
Pearson	-0.2363	-0.3332	-0.0985	-0.1195	0.3374
Spearman	-0.2273	-0.3435	-0.1034	-0.1447	0.3595

and 4(c), user engagement decreases not as smoothly, which means a weaker correlation between user engagement and these two metrics.

Furthermore, we quantify the correlation coefficients of the *logarithmical* values of the QoS metrics with the conditional expectations of engagement. To alleviate value 0 in the logarithm operators, all the QoS values are increased by 1; i.e., $Q' = \ln(Q + 1)$, where Q is the original QoS value and Q' is the logarithmical one. We employ the correlation metrics Pearson correlation coefficient t [36] and Spearman rank coefficient [39]. The first metric could identify the linear relationship with some Gaussian noise while the second one emphasizes the monotonicity between variables. As shown in Table 3, the results confirm that, among all the QoS metrics, buffer frequency and restartup buffer ratio show the strongest log-linear correlations with engagement, which means the largest weight should be assigned to these two metrics in the engagement predictive model.

5.2. User Engagement versus User Interest. We now examine the relationship of user engagement and user interest. As shown in Figure 5(a), user engagement increases linearly with an increasing interest level, except for that at the head ([1, 2]) and the tail ([9.7, 10]). Fortunately, the ranges of the head and tail only account for 0.9% of the sessions. Thus, the linear characteristic still dominates the relationship.

Moreover, the correlation coefficient between user engagement and user interest is stronger than that with most of the QoS metrics, as listed in Table 1. The results confirm that individual interest has a roughly linear dependence on the engagement.

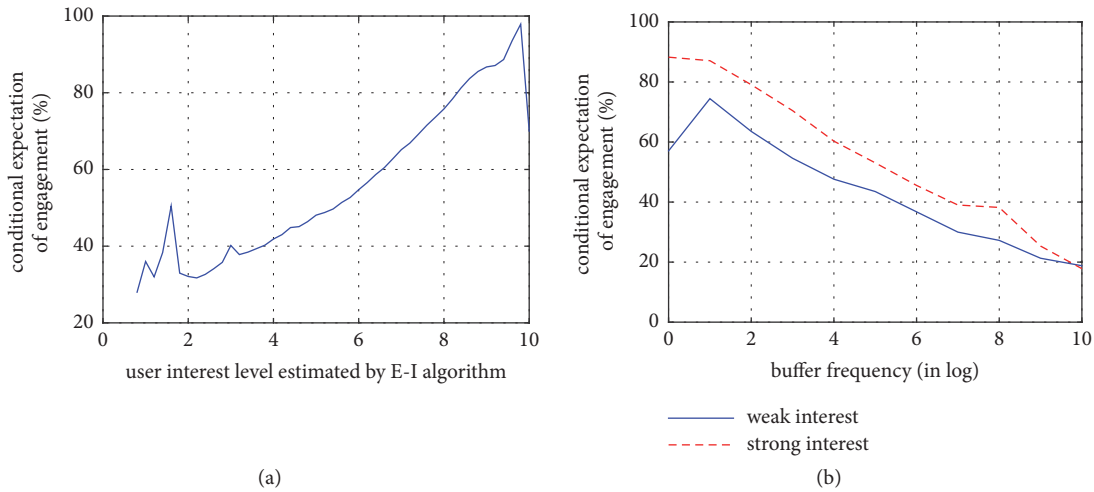


FIGURE 5: The expectations of user engagement conditional on (a) the user interest level and those on (b) the buffering frequency in the sessions where users are, respectively, strongly and weakly interested in the videos.

5.3. User Engagement versus QoS versus User Interest. Intuitively, QoS metrics and user interest may not independently impact user engagement. For example, users may have different tolerance to the quality issues when they have divergent interest degree towards the videos.

To confirm this intuition, we compare the engagement-QoS relationship under the strong-interest level and that under the weak-interest level, respectively, as shown in Figure 5(b). We use buffer frequency here as an example of the QoS metrics. In statistics, we split the logarithmical value of buffer frequency into 10 bins and denote the binned buffer frequency with a score between 0 and 10. The interval is set to be 1 score for statistic.

The boundary between the strong- and the weak-interest level is set to be the median of all the users' interest levels.

As shown, when the users are more interested in the video, the engagement decreases more quickly against an increasing buffer frequency. In other words, users with a stronger interest tend to be more sensitive to the quality problem. The significant difference between these two cases indicates the existence of the multiplicative effect of user interest and the QoS on user engagement.

6. Model Building and Evaluation

6.1. Engagement Model. According to the measurement results above, we propose a***Q****oS and user* ***I****nterest based* ***E****ngagement* ***(QI-E)*** *regression model.* In this model, for user u with an interest rating R with video v, we predict his/her engagement E for this video to be

$$E = f\left(\mathbf{Q}', R\right) = \boldsymbol{\theta}_q^T \mathbf{Q}' + \theta_r R + r\boldsymbol{\theta}_{qr}^T \mathbf{Q}' + \theta_0 \qquad (2)$$

where $\mathbf{Q}' = [Q'_1, \cdots, Q'_k, \cdots Q'_K]$ is a QoS vector where the element Q'_k is the logarithmical value of the kth QoS metric; i.e., $Q'_k = \ln(Q_k + 1)$, where Q_k is the original QoS value. All the QoS metrics are involved in the vector, and then $K = 4$. $\theta_* \in \boldsymbol{\theta}$ are the weight parameters to be learned by experiments.

Based on this model, the expectation of user engagement conditional on the kth QoS metric with a logarithmical value of $Q'^{(k)} = q$ can be derived to be

$$\mathbb{E}\left(E \mid Q'^{(k)} = q\right) = (\mathbb{E}(R) + 1)\, q + c\mathbb{E}(R) + c \qquad (3)$$

where $\mathbb{E}(R)$ is the expectation of user interest rating. $c = \sum_{\bar{k}=1,\neq k}^{K} \mathbb{E}(Q'^{(\bar{k})}) + 1$ and $Q'^{(\bar{k})}$ is the logarithmical value of the $\bar{k}$th QoS metric but not the kth one. Both of them are constants. For clarity, the parameters $\theta_* \in \boldsymbol{\theta}$ are omitted in the derivation.

From (3), we observe that the conditional expectation of user engagement $\mathbb{E}(E \mid Q'^{(k)} = q)$ is linearly proportional to the logarithmical QoS value q, which is consistent with the measurement result in Section 5.1. Furthermore, with a larger interest rating R, there is a larger slope, $\mathbb{E}(R) + 1$, in the linear relationship, which is consistent with the measurement result in Section 5.3. Similarly, through a deviation of user engagement conditional on interest rating $R = r$, it is easy to show that the model is consistent with the measurement result in Section 5.2.

6.2. Model Evaluation. We randomly select 80% of the whole dataset for training and the rest for testing. Via 10-fold cross validation, we fix the parameters θ_* in the QI-E model.

For comparison, we propose three groups of linear regression models as baselines. In the first group, denoted by BS-1, the models consider separate QoS metrics, respectively. In the second baseline model, denoted by BS-2, we take into account all the QoS metrics and the third one, denoted by BS-3, considers user interest additionally.

In Table 4(a), we first evaluate the first group baselines models. Among them, the model considering *buffer frequency* has the smallest *RMSE of 3.1553* as expected as this metric has the largest correlation coefficient with engagement as shown in Table 3.

Table 4: Performance of the engagement predictive models.

(a) Models BS-1 using each single QoS metric as input feature

Input Feature	Buffer ratio	Average buffer length	Buffer frequency
RMSE	3.1727	3.1871	3.1553

(b) Models using quality factors and human factors

Model	Input Feature	RMSE	Improvement
BS-2	All the QoS metrics	3.0926	- -
BS-3	QoS, interest	2.9846	3.5%
QI-E	QoS, interest (multiplicative item)	2.8588	7.6%

In Table 4(b), we evaluate the proposed QI-E model compared with the model that consider all the QoS metrics. As the results shown, when user interest is considered, there is an improvement of 3.5%. When the multiplicative effect of the two kinds of factors is considered in the QI-E model, the improvement climbs up to 7.6%.

The positive results confirm the effectiveness of our proposed engagement predictive model and demonstrate the necessity of understanding how the QoS factors and user interest impact user engagement. Although the experiment is conducted on a dataset from an application on PC-clients, our method and results are easy to be extended to the context of mobile clients.

7. Summary and Discussion

7.1. Summary. In this paper we have shown that, in order to optimize user engagement in VoD streaming systems directly, an effective model of engagement incorporating both user interest and perceptual quality factors in an explicit function is needed. To this end, we have proposed an Extraction-Inference (E-I) algorithm to estimate the user interest from easily obtained user behaviors. Furthermore, we have built a QoS and user Interest based Engagement (QI-E) regression model based on an experimental analysis over a large-scale dataset. This model offers an improvement in accuracy by 9.99% over the baseline model considering only QoS factors. The positive results demonstrate that user interest as well as QoS plays an important role in user engagement prediction.

7.2. Discussion for Implications. Our research on understanding and modeling user engagement can be applied to most of the up-to-date network environments, including fog computing. For example, it could help designers to make tradeoffs between QoS factors under diverse user interest through *CDN selection, streaming decisions,* and so on [48, 49].

Especially in the fog computing context, as provided with fully explored localized user features and service demand, user interest in this case is more predictable and users' personalized requirement could be better satisfied [4, 46]. For a goal of global optimization, based on our model, designers could provide the users with higher interest with a priority to the bandwidth optimization under a limited bandwidth condition. As another example, in an engagement-oriented recommendation system, designers can make a tradeoff between videos that satisfy user interest and that provide a better QoS.

In the future, we will further extend the model to consider more factors. For example, some other kinds of user behaviors like drag can help understand the user's patience. Moreover, we can design scheduling schemes specifically for the fog computing context.

Data Availability

The dataset is supported by PPTV, a commercial enterprise. The data is not for public.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Science Foundation of China under Grant nos. 61572071, 61271199, and 61301082.

References

[1] B. Li, Z. Wang, J. Liu, and W. Zhu, "Two decades of internet video streaming: A retrospective view," in *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, vol. 9(1s), p. 33, 2013.

[2] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, and H. Zhang, "Developing a predictive model of quality of experience for internet video," in *Proceedings of the ACM SIGCOMM 2013 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, SIGCOMM 2013*, pp. 339–350, China, August 2013.

[3] M. Watson, "Http adaptive streaming in practice," in *Proceedings of the In of the ACM Multimedia Systems Conference (MMSys)Keynote*, San Jose, CA, USA, 2011.

[4] S. Yi, C. Li, and Q. Li, "A survey of fog computing: concepts, applications and issues," in *Proceedings of the Workshop on Mobile Big Data (Mobidata '15)*, pp. 37–42, ACM, Hangzhou, China, June 2015.

[5] V. Pande, C. Marlecha, and S. Kayte, "A review- fog computing and its role in the internet of things," *International Journal of Engineering Research and Application*, vol. 6, no. 10, pp. 2248–96227, 2016.

[6] Recommendation ITU-T P.10/G.100 Amendment 2, Std., Jul. 2008.

[7] V. A. Siris, K. Balampekos, and M. K. Marina, "Mobile quality of experience: Recent advances and challenges," in *Proceedings of the 2014 IEEE International Conference on Pervasive Computing and Communication Workshops, PERCOM WORKSHOPS 2014*, pp. 425–430, Hungary, March 2014.

[8] S. S. Krishnan and R. K. Sitaraman, "Video stream quality impacts viewer behavior: Inferring causality using quasi-experimental designs," *IEEE/ACM Transactions on Networking*, vol. 21, no. 6, pp. 2001–2014, 2013.

[9] K. U. R. Laghari, N. Crespi, B. Molina, and C. E. Palau, "QoE aware service delivery in distributed environment," in *Proceedings of the 25th IEEE International Conference on Advanced Information Networking and Applications Workshops, WAINA 2011*, pp. 837–842, Singapore, March 2011.

[10] W. Wu, A. Arefin, R. Rivas, K. Nahrstedt, R. Sheppard, and Z. Yang, "Quality of experience in distributed interactive multimedia environments: Toward a theoretical framework," in *Proceedings of the 17th ACM International Conference on Multimedia, MM'09, with Co-located Workshops and Symposiums*, pp. 481–490, China, October 2009.

[11] K. U. R. Laghari, K. Connelly, and N. Crespi, "Toward total quality of experience: a QoE model in a communication ecosystem," *IEEE Communications Magazine*, vol. 50, no. 4, pp. 58–65, 2012.

[12] M. Z. Shafiq, J. Erman, L. Ji, A. X. Liu, J. Pang, and J. Wang, "Understanding the impact of network dynamics on mobile video user engagement," *ACM SIGMETRICS Performance Evaluation Review*, vol. 42, no. 1, pp. 367–379, 2014.

[13] M. Alreshoodi and J. Woods, "Survey on QoE\QoS correlation models for multimedia services," *International Journal of Distributed and Parallel Systems*, vol. 4, no. 3, 2013.

[14] T. Hoßfeld, R. Schatz, and U. R. Krieger, "Qoe of Youtube video streaming for current internet transport protocols," in *Measurement, Modelling, and Evaluation of Computing Systems and Dependability and Fault Tolerance*, pp. 136–150, Springer International Publishing, 2014.

[15] A. Balachandran, V. Aggarwal, E. Halepovic et al., "Modeling web quality-of-experience on cellular networks," in *Proceedings of the 20th ACM Annual International Conference on Mobile Computing and Networking, MobiCom 2014*, pp. 213–224, USA, September 2014.

[16] F. Dobrian, V. Sekar, and A. Awan, "Understanding the impact of video quality on user engagement," in *Proceedings of the ACM SIGCOMM Conference (SIGCOMM '11)*, vol. 41, pp. 362–373, August 2011.

[17] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoßfeld, and P. Tran-Gia, "A survey on quality of experience of HTTP adaptive streaming," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 469–492, 2015.

[18] Y. Qi and M. Dai, "The effect of frame freezing and frame skipping on video quality," in *Proceedings of the Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP'06. International Conference on IEEE*, pp. 423–426, 2006.

[19] J. Jiang, V. Sekar, and H. Zhang, "Improving fairness, efficiency, and stability in http-based adaptive video streaming with festive," in *Proceedings of the 8th International Conference on Emerging Networking Experiments And Technologies*, pp. 97–108, 2012.

[20] L. De Cicco, S. Mascolo, and V. Palmisano, "Feedback control for adaptive live video streaming," in *Proceedings of the of the second annual ACM conference on Multimedia systems*, pp. 145–156, 2011.

[21] Y. Xu, E. Altman, R. El-Azouzi, S. E. Elayoubi, and M. Haddad, "QoE analysis of media streaming in wireless data networks," in *NETWORKING*, pp. 343–354, Springer Berlin Heidelberg, 2012.

[22] S. Khemmarat, R. Zhou, D. K. Krishnappa, L. Gao, and M. Zink, "Watching user generated videos with prefetching," *Signal Processing: Image Communication*, vol. 27, no. 4, pp. 343–359, 2012.

[23] S. H. Shen and A. Akella, "An information-aware qoe-centric mobile video cache," in *Proceedings of the of the 19th annual international conference on Mobile computing networking*, pp. 401–412, 2013.

[24] A. Gember, A. Akella, J. Pang, A. Varshavsky, and R. Caceres, "Obtaining in-context measurements of cellular network performance," in *Proceedings of the 2012 ACM Internet Measurement Conference, IMC 2012*, pp. 287–300, USA, November 2012.

[25] A. Floris, L. Atzori, and G. Ginesu, "Addressing un-interoperability issues in QoE models: Is a layered modelling effective?" in *Proceedings of the 2014 IEEE International Conference on Communications Workshops, ICC 2014*, pp. 563–568, Australia, June 2014.

[26] M. Verhoeyen, J. D. Vriendt, and D. D. Vleeschauwer, "Optimizing for video storage networking with recommender systems," *Bell Labs Technical Journal*, vol. 16, no. 4, pp. 97–113, 2012.

[27] K. Verbert, N. Manouselis, X. Ochoa et al., "Context-aware recommender systems for learning: a survey and future challenges," *IEEE Transactions on Learning Technologies*, vol. 5, no. 4, pp. 318–335, 2012.

[28] A. S. Lampropoulos and G. A. Tsihrintzis, "A survey of approaches to designing recommender systems," in *Proceedings of the Multimedia Services in Intelligent Environments*, pp. 7–30, Springer International Publishing, 2013.

[29] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005.

[30] L. Si and R. Jin, "Flexible mixture model for collaborative filtering," *ICML*, vol. 3, pp. 704–711, 2003.

[31] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in Artificial Intelligence*, vol. 2009, 19 pages, 2009.

[32] N. Wang, T. Yao, J. Wang, and D. Y. Yeung, "A probabilistic approach to robust matrix factorization," in *Computer Vision–ECCV 2012*, pp. 126–139, Springer Berlin Heidelberg, 2012.

[33] R. Mittelman and E. L. Miller, "Fast Gauss Transforms based on a High Order Singular Value Decomposition for Nonlinear Filtering," in *Proceedings of the 2007 IEEE/SP 14th Workshop on Statistical Signal Processing*, pp. 94–98, Madison, WI, USA, August 2007.

[34] Y. Koren, "The bellkor solution to the netflix grand prize," *Netflix Prize Documentation*, 2009.

[35] M. Montaner, B. López, and J. L. de la Rosa, "A taxonomy of recommender agents on the internet," *Artificial Intelligence Review*, vol. 19, no. 4, pp. 285–330, 2003.

[36] W. Woerndl, A. Helminger, and V. Prinz, "Experiences from implementing collaborative filtering in a web 2," in *Proceedings of the International Workshop on Adaptation and Personalization for Web*, vol. 2, pp. 120–129, 2009.

[37] M. Claypool, P. Le, M. Wased, and D. Brown, "Implicit interest metrics," in *Proceedings of the 6th international conference on Intelligent user interfaces*, pp. 33–40, 2001.

[38] D. Weiß, J. Scheuerer, M. Wenleder, A. Erk, M. Gülbahar, and C. Linnhoff-Popien, "A user profile-based personalization system for digital multimedia content," in *Proceedings of the 3rd International Conference on Digital Interactive Media in Entertainment and Arts, DIMEA 2008*, pp. 281–288, Greece, September 2008.

[39] J. Davidson, B. Liebald, J. Liu et al., "The YouTube video recommendation system," in *Proceedings of the Fourth ACM Conference on Recommender Systems*, pp. 293–296, 2010.

[40] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 452–461, AUAI Press, 2009.

[41] D. O'Sullivan, B. Smyth, and D. Wilson, "Explicit vs implicit profiling-a case-study in electronic programme guides," in *IJCAI*, 2003.

[42] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Proceedings of the Data Mining, ICDM'08. Eighth IEEE International Conference on IEEE*, pp. 263–272, 2008.

[43] F. Lin and X. Lü, "QoS guaranteed pre-pushing scheme in peer-assisted streaming network," *China Communications*, vol. 11, no. 14, pp. 111–117, 2014.

[44] F. Lin, X. Zhou, X. Lü, and W. Song, "Novel pre-pushing scheme for peer-assisted streaming network based on multi-leader multi-follower stackelberg model," *Wireless Personal Communications*, vol. 80, no. 1, pp. 289–301, 2015.

[45] J. Su, F. Lin, X. Zhou, and X. Lu, "Steiner tree based optimal resource caching scheme in fog computing," *China Communications*, vol. 12, no. 8, Article ID 7224698, pp. 161–168, 2015.

[46] T. H. Luan, L. Gao, Z. Li et al., "Fog computing: focusing on mobile users at the edge," *Computer Science*, 2015.

[47] http://www.pptv.com/.

[48] H. Patrick, *"How Much Online Video Quality is Enough?" white paper, Senior Director, Product Marketing at Citrix*, Japan, 2011.

[49] D. D. Vleeschauwer and K. Laevens, "Performance of caching algorithms for IPTV on-demand services," *IEEE Transactions on Broadcasting*, vol. 55, no. 2, pp. 491–501, 2009.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 3471312, 12 pages
https://doi.org/10.1155/2018/3471312

Research Article

Adjacency-Hash-Table Based Public Auditing for Data Integrity in Mobile Cloud Computing

Wenqi Chen,[1] Hui Tian,[1] Chin-Chen Chang,[2] Fulin Nan,[1] and Jing Lu[3]

[1]*College of Computer Science and Technology, National Huaqiao University, Xiamen 361021, China*
[2]*Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan*
[3]*Network Technology Center, National Huaqiao University, Xiamen 361021, China*

Correspondence should be addressed to Hui Tian; htian@hqu.edu.cn

Received 1 May 2018; Accepted 28 August 2018; Published 13 September 2018

Academic Editor: Ke Xiong

Copyright © 2018 Wenqi 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.

Cloud storage, one of the core services of cloud computing, provides an effective way to solve the problems of storage and management caused by high-speed data growth. Thus, a growing number of organizations and individuals tend to store their data in the cloud. However, due to the separation of data ownership and management, it is difficult for users to check the integrity of data in the traditional way. Therefore, many researchers focus on developing several protocols, which can remotely check the integrity of data in the cloud. In this paper, we propose a novel public auditing protocol based on the adjacency-hash table, where dynamic auditing and data updating are more efficient than those of the state of the arts. Moreover, with such an authentication structure, computation and communication costs can be reduced effectively. The security analysis and performance evaluation based on comprehensive experiments demonstrate that our protocol can achieve all the desired properties and outperform the state-of-the-art ones in computing overheads for updating and verification.

1. Introduction

Cloud storage is one of cloud computing services and provides a way to effectively store and manage big data [1]. In recent years, more and more individuals and businesses tend to outsource their data to the cloud, since outsourcing data can render the advantages of location independent resource pooling, flexible resources, universal network access, and usage-based pricing [2–6]. Although the benefits of cloud storage services are many and huge, it also faces a lot of challenges [2, 4]. For example, the security of data sharing and storage in the same group is an urgent issue to be solved in the cloud environment [7]. In addition, data deduplication in cloud storage is also one of the vital techniques to reduce the amount of storage space and save bandwidth [8, 9]. Particularly, due to the separation of data ownership and management, cloud users (data owners) cannot verify the integrity of their data in the traditional techniques, which leads to a trust gap between cloud users and the Cloud Service Provider (CSP). In addition, Cloud storage is also faced with many internal and external security threats [10–15] (e.g., byzantine failures, hacker attacks, etc.), which may lead to cloud data corruption or loss. To solve these concerns, the cloud data auditing whose purposes are to enhance the data security of cloud storage platforms and to improve mutual trust between users and the CSP is proposed.

The most core challenge of cloud data auditing is how to efficiently check the cloud data integrity. To address this problem, a provable data possession (PDP) protocol and a proof of retrievability (PoR) protocol have been provided, respectively, by work [16] and work [17]. In typical PDP protocols, the user first generates some metadata (such as block tags) for a data file to verify integrity of the data on cloud servers. Later, the user sends the file and metadata to the cloud servers and removes them from its local storage. PDP employs a challenge-response mode for the remote verification; i.e., the CSP can generate a proof for the verifier's challenge. Compared with the former, the latter (PoR) is a complementary protocol to PDP. In initial PoR protocols, the user first encodes the data file with error-correcting code before outsourcing data to the CSP. Therefore, the user can reconstruct the entire file from the CSP's partial response. The

PoR model focuses on static data. Compared with PoR, PDP is more suitable for dynamic data auditing; see [18–28].

The existing PDP protocols can be generally divided into two categories: private auditing and public auditing. In private auditing, the user is as an only verifier to remotely verify the data integrity with low overhead. Due to no trust between the user and the CSP, the user cannot provide convincing results for verification. What is more, it is not advisable for the user to conduct the audits for their data frequently, since one of the important motivations of outsourcing data is to reduce the user's burden of storage management. To address this problem, a public auditing protocol was first provided by Ateniese et al. [16], in which an independent authorized auditor (Third Party Auditor, TPA), not only the user, can remotely verify the data integrity. Therefore, the TPA can not only provide independent audit results, but also bear the communication overhead and computation costs in the entire verification phase. The public audit for cloud storage should also achieve some security and function requirements as follows:

(i) Privacy preserving: in public auditing, the TPA on behalf of the user periodically verifies the integrity of data on the cloud servers. Thus, auditing protocols should design a mechanism to ensure that the TPA cannot derive user's data contents from the collected information during the verification phase.

(ii) Batch auditing: batch auditing is defined as the TPA can deal with auditing tasks from multiple various users simultaneously, which not only reduces the numbers of communications between the TPA and the CSP during the auditing phase, but also enhances the verification efficiency.

(iii) Dynamic auditing: in the cloud storage environments, there are a lot of various application data (financial trade, social media, etc.), which need to be updated frequently. Therefore, dynamic data auditing is a significant function for cloud storage auditing.

For the dynamic data audit, Erway el at. [19] first provided an extended PDP protocol, named as dynamic provable data possession (DPDP), which introduced a dynamic authenticated data structure, rank-based authenticated skip list, to support data updating. Later, Wang el at. [20] proposed a protocol based on the BLS signature, which utilized Merkle Hash Tree (MHT) to achieve data updating. However, the above two protocols would cause heavy computational overhead of the TPA and large communication costs during the verification phase and the updating phase. Further, [23], [26], and [27], respectively, design the dynamic authenticated data structures, Index Hash Table (IHT), Dynamic Hash Table (DHT), and Doubly Linked Info Table (DLIT) to improve audit efficiency and the structures are stored in the TPA rather than the CSP, to reduce communication costs. Though the above protocols achieve auditing effectively, the methods still have some drawbacks. In [23], updating operations incur large computational overhead, especially insertion and delete operations. Thus, [26] and [27], respectively, design the structures to overcome the above drawbacks in [23]. However, search operations in [26, 27] are relatively inefficient in the verification phase and the updating phase.

In view of above problems, this paper introduces a novel dynamic data authenticated structure adjacency-hash table (AHT) in our public auditing protocol (AHT-PA). We employ the AHT to achieve dynamic auditing. Moreover, due to AHT stored in the TPA instead of CSP, its computational overhead and communication costs are significantly less than both the protocol based on the skip list [19] and the one using MHT [20]. In the verification phase and the updating phase, AHT-PA also outperforms the protocols [23, 26, 27]. We exploit the bilinear maps and Boneh-Lynn-Shacham (BLS) signatures to support batch auditing and employ random masking to achieve privacy preserving. Our contributions can be summarized as follows:

(1) We propose a novel public auditing protocol, which can simultaneously support the essential functions: privacy preserving, batch auditing, and dynamic data auditing.

(2) We introduce a novel dynamic structure, AHT, to save data properties for dynamic data auditing. With such structure, our protocol can effectively achieve the dynamic data auditing and the data updating.

(3) We prove the security of the presented protocol and justify the auditing performance by concrete experimental comparisons with the state of the arts. The results demonstrate that our protocol can efficiently achieve secure auditing and outperform the previous ones in computational overhead and communication costs.

The rest of the paper is organized as follows: in Section 2, we review the related work concerning cloud storage auditing, particularly, regarding the dynamic data auditing. Then, we introduce the background and the necessary preliminaries for our work in Section 3. Section 4 gives the detailed description of our protocol. Section 5 presents the security proofs of our protocol, and Section 6 gives the comprehensive performance evaluations through experimental comparisons with some existing protocols. Finally, Section 7 gives the concluding remark of this paper.

2. Related Work

In recent years, many researchers have focused on cloud storage auditing. In 2007, Atenises et al. [16] proposed one of the earliest related works, "provable data possession (PDP)", which employs the based-RSA homomorphic authenticator to check the data integrity. At the same year, Juels el at. [17] presented a complementary protocol, "Proof of Retrievability (PoR)", which can not only check the correctness of data on cloud, but also ensure the retrievability of cloud data with an encoding method (error-correcting code). However, due to encoding the file before outsourcing to the CSP, the PoR model focuses on static data, such as archive data. Compared with PoR, PDP is more suitable for dynamic data auditing. As mentioned earlier, the public auditing has some advantages over private auditing. In public auditing, TPA

Table 1: Function comparison of auditing protocols.

Protocols	Public auditing	Privacy protection	Dynamic auditing	Batch auditing
PDP[16]	√	×	×	×
PoR[17]	×	⊗	×	×
IHT-PA[23]	√	√	√	⊙
DAP[22]	√	√	√	√
DPDP(skip list)[19]	×	⊗	√	×
DPDP(MHT)[20]	√	√	√	√
DHT-PA[26]	√	√	√	√
DLIT-PA[27]	√	×	√	√
AHT-PA	√	√	√	√

Note: "√" means "support"; "×" means "not support"; "⊗" means "no demand"; and "⊙" means "not mentioned".

can not only provide independent audit results, but also bear the communication overhead and computation costs for the entire verification phase. Therefore, it is considered a more practical model [15, 26]. Besides, public auditing should also achieve some security and function requirements, for example, privacy preserving, batch auditing, and dynamic auditing.

To overcome the data leakage to the TPA, Wang et al. [28] first provided a public auditing protocol for privacy preserving, where the CSP integrates the aggregate value of the data blocks with random masking. Therefore, this protocol can guarantee that the TPA cannot learn any knowledge of the user data during the verification phase. Later, [22, 23, 26] show that privacy preserving is indispensable in public auditing. Moreover, [15] and [28] are extended to preform audit tasks from multiple users simultaneously for better performance. In work [15] and work [28], the approach for batch auditing is that the CSP aggregates the data block tags generated by various users and then the TPA uses them and related block information responded from the CSP to verify the integrity of the cloud data.

For the auditing dynamic data, Erway el at. [19] provided an extended PDP protocol, named dynamic provable data possession (DPDP), which first introduced a dynamic authenticated data structure, rank-based authenticated skip list, to support data updating. Later, Wang el at. [20] proposed a protocol based on the BLS signature, which utilized Merkle Hash Tree to achieve data updating. However, the above two protocols would cause heavy computational overhead of the TPA and large communication costs during the verification phase and the updating phase. Further, [23], [26], and [27], respectively, designed the dynamic authenticated data structures, Index Hash Table (IHT), Dynamic Hash Table (DHT), and Doubly Linked Info Table (DLIT) to improve audit efficiency and to reduce communication costs by storing the structures in the TPA instead of the CSP. Though the above protocols can effectively achieve public auditing, the methods still have some drawbacks. In [23], updating operations incur large computational overhead, especially insertion and delete operations. Thus, [26] and [27], respectively, design the structure to overcome the above drawbacks in [23]. However, search operation in [26, 27] is relatively inefficient in the verification phase and the updating phase. Therefore, this paper introduces a novel dynamic data authenticated structure, adjacency-hash table (AHT), to achieve better auditing and updating efficiency.

To highlight the difference between our protocol and the existing ones, Table 1 shows comparison results of functions among them. It is clear that the presented protocol (AHT-PA) supports all the mentioned audit functions.

3. Background and Preliminaries

3.1. Problem Statement. As illustrated in Figure 1, we concentrate on designing an AHT-based public audit protocol which includes the following three entities: Users have large amounts of data and outsource their data to the cloud. Cloud Service Provider (CSP) has large-scale computing and storage devices and provides users with cloud storage services. Third Party Auditor (TPA) undertakes audit tasks for users and provides fair and objective audit results. Users outsource their data to the cloud to enjoy the reliability of data storage and high-performance services and to reduce its maintenance overhead. However, since the CSP manages their data on the cloud rather than users, users strongly desire to periodically check the integrity and correctness of their data.

As mentioned in the existing protocols [18, 19, 23, 26], the TPA is pointed out to be credible but curious. In other words, although the TPA can credibly perform the audit in the verification phase, it may be curious about the privacy information of users' data and even may try to derive the users' data contents. In addition, the CSP is considered as an untrustworthy party. For gaining benefits or maintaining their reputations, the CSP may hide the fact of data loss and even delete some data that users rarely access. In particular, the CSP may further launch three attacks to the TPA:

(i) Forge attack: the CSP may attempt to forge the data blocks and their corresponding tags to pass the audit.

(ii) Replacing attack: the CSP may attempt to pass the audit by replacing a corrupted block and its tag with another block and its corresponding tag.

(iii) Reply attack: the CSP may attempt to pass the audit using the proof messages generated previously.

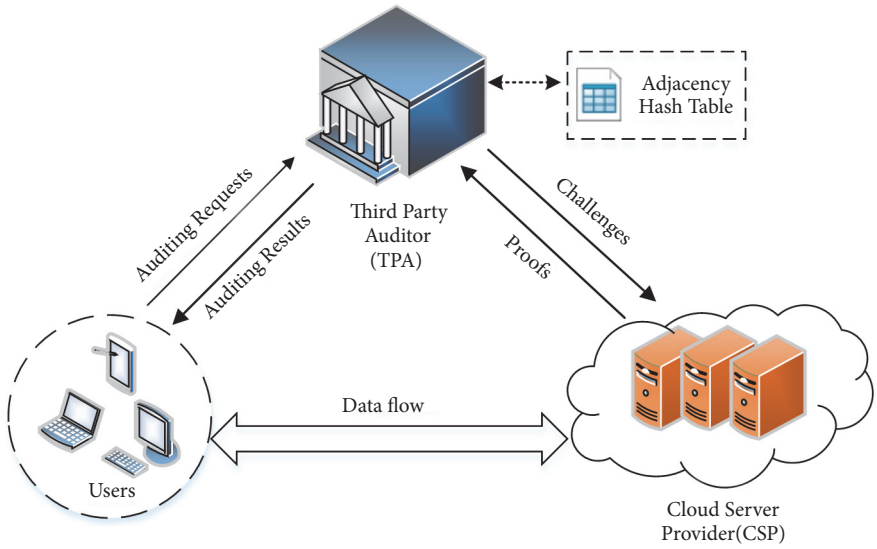


FIGURE 1: System architecture for public auditing.

To achieve the secure and efficient public auditing, our protocol aims to meet the following desired properties:

(1) Public auditing: it allows any authorized TPA to verify the correctness and integrity of user's data on the cloud servers.

(2) Blockless verification: it allows TPA to audit cloud data without retrieving the data blocks.

(3) Storage correctness: the CSP, who does not store the intact data as required, cannot pass the audit.

(4) Dynamic data audit: it allows the users to perform dynamic data operations (insertion, modification, and deletion) and achieves the efficient public auditing.

(5) Privacy preserving: it ensures that TPA cannot learn knowledge of users' data from the information collected during verification phase.

(6) Batch auditing: the TPA has the capability to deal with multiple auditing tasks from various users in a cost-effective way.

(7) Lightweight: it allows the TPA to perform public verification with minimum communication and computation costs.

3.2. Adjacency-Hash Table. As mentioned earlier, [19] and [20], respectively, introduced the authenticated skip list and the MHT to support public dynamic auditing. However, the above two protocols would cause heavy computational overhead of the TPA and large communication costs during the verification phase and the updating phase. Further, [26] and [27], respectively, designed Dynamic Hash Table (DHT) and Doubly Linked Info Table (DLIT) to improve audit efficiency and to reduce communication costs by storing the structures in the TPA rather than the CSP. Note that the DHT is a single linked table and the DLIT is a double linked table. Though the above protocols [26, 27] can achieve efficient auditing, the methods still have some drawbacks. Particularly, the search operation in [26, 27] is relatively inefficient in the verification phase and the updating phase. Therefore, this paper introduces a novel dynamic data authenticated structure, adjacency-hash table (AHT), to achieve better the auditing and updating efficiency.

The AHT is utilized by the TPA to store the latest version information (VI) of data blocks, as illustrated in Figure 2. The AHT is divided into file elements and the corresponding tables, called Adjacency Tables (AT). Every file element in file arrays includes the index number (NO_j), the file identifier (ID_j) of the given file (e.g., F_j), and a pointer indicating an AT. Each AT consists of block elements and a counter array whose every element contains a pointer indicating a block element and a value ($cValue_i$) which records the number of block elements after the corresponding pointer. Each file is organized by a file element and the corresponding AT. Each block element (e.g., the element corresponding the i-th block of the j-th file $m_{j,i}$) in the AT consists of the current version number of the block $v_{j,i}$, its time stamp $t_{j,i}$, and a pointer to the next node. Accordingly, the operations on the AHT are divided into file operations and block operations. To search the x-th ($1 \le x \le n$) block element, the TPA first determines the value a according to

$$\sum_{i=0}^{a} cValue_i < x \le \sum_{i=0}^{a+1} cValue_i, \tag{1}$$

where i is the index of the counter array ($0 \le i \le n$) and the head element whose index is 0 in the counter array is used to indicate that $cValue_0$ should be set to 0. Note that the head is not drawn in Figure 2, because it is virtual. Further, the TPA calculates the distance dis, namely,

$$dis = x - \sum_{i=0}^{a} cValue_i. \tag{2}$$

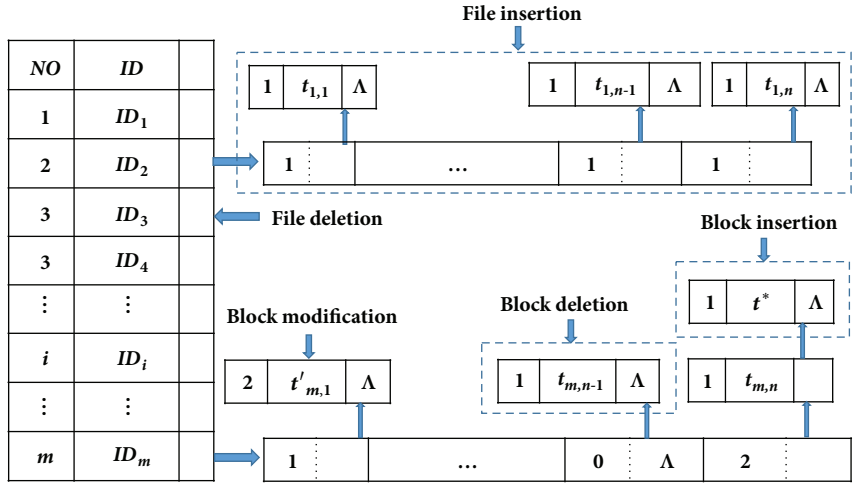


Figure 2: Adjacency-hash table (AHT).

Apparently, the x-th block element is the dis-th block element after the pointer of (a+1)-th element in the counter array. To insert a block element after an existing block, the TPA first tracks the given node (the given block element) and inserts the new node after it; the deletion of the given block element is to first track the given node and to delete it from the current AT. Besides, when the value ($cValue_i$) is equal to "0", the corresponding element in the counter array should be deleted. The search process of a file is to locate the file element according to its identifier or index; the insertion of the given file is to first insert a file element in the file array and then to construct a AT which includes related block elements; the deletion of the given file is to first remove the AT and to delete its file element; the modification of the given file is to update the file element and corresponding block elements.

3.3. Preliminaries. To facilitate understanding for readers, this section first introduces some necessary knowledge of cryptography for the presented protocol.

Bilinear Map. Let $\mathbb{G}$, $\mathbb{G}_T$ be multiplicative cyclic groups of a large prime order p. A map function e: $\mathbb{G} \times \mathbb{G} \longrightarrow \mathbb{G}_T$ with the following properties: (1) Bilinear: $\forall u, v, h \in \mathbb{G}$ and $\forall a, b \in \mathbb{Z}_p$, $e(u^a, u^b) = e(u, v)^{ab}$, and $e(u, v \cdot h) = e(u \cdot h, v) = e(u, v) \cdot e(u, h)$. (2) Computable: e is an efficient computable algorithm. (3) Nondegeneracy: if g is a generator of $\mathbb{G}$, then $e(g, g) \neq 1$.

BLS-Based Homomorphic Verifiable Authenticator (BLS-HVA). BLS-HVA is widely utilized for public auditing protocols [15, 18, 19, 22–24, 26–29], which can enable a public verifier to verify the cloud data integrity without downloading its original data. To be specific, BLS-HVAs are generated by BLS signatures in public auditing. Consequently, BLS-HVAs satisfy the properties as follows:

(i) Blockless verifiability [16]: constructing the proof in the BLS-HVA, the TPA can verify the cloud data integrity without its actual data content.

(ii) Homomorphism [16]: let $\mathbb{G}$ and $\mathbb{H}$ be multiplicative groups of a large prime order p, and "$\oplus$" and "$\otimes$" be operations in $\mathbb{G}$ and $\mathbb{H}$, respectively. If a map function f: $\mathbb{G} \longrightarrow \mathbb{H}$ satisfies homomorphism, then $\forall h_1, h_2 \in \mathbb{G}$, $f(h_1 \oplus h_2) = f(h_1) \otimes f(h_2)$.

(iii) Nonmalleability [30]: let σ_1 and σ_2 denote signatures on blocks m_1 and m_2, respectively, and β_1 and β_2 two random numbers in $\mathbb{Z}_p$. For the given block, $m' = \beta_1 m_1 + \beta_2 m_2$, a user, who does not know the private key sk, cannot generate the signature σ' of m' by combining σ_1 and σ_2.

3.4. Secure Assumptions. The security of the present protocol is based on the following assumptions.

Computational Diffe-Hellman (CDH) Assumption. Let $\mathbb{G}$ be multiplicative cyclic groups of a large prime order p. Given g^a and g^b, where g is a generator of $\mathbb{G}$, and $a, b \in \mathbb{Z}_p$, it is computationally intractable to compute g^{ab}. For any probabilistic polynomial-time adversary $\mathcal{A}$, the probability of solving the CDH problem is negligible, namely,

$$Pr\left(\mathcal{A}_{\mathrm{CDH}}\left(g, g^a, g^b \in \mathbb{G}\right) \longrightarrow g^{ab} \in \mathbb{G} : \forall a, b \underset{R}{\in} \mathbb{Z}_p\right) \leq \varepsilon. \tag{3}$$

Discrete Logarithm (DL) Assumption. Let $\mathbb{G}$ be multiplicative cyclic groups of a large prime order p. Given h (such as $h = g^a$, where g is a generator of $\mathbb{G}$, and $a \in \mathbb{Z}_p$), it is computationally intractable to compute a. For any probabilistic polynomial-time adversary $\mathcal{A}$, the probability of solving the DL problem is negligible, namely,

$$Pr\left(\mathcal{A}_{\mathrm{DL}}\left(g, h \in \mathbb{G}\right) \longrightarrow a \in \mathbb{Z}_p, \text{ s.t. } h = g^a\right) \leq \varepsilon. \tag{4}$$

4. The Proposed Protocol Based on AHT

In this section, we will present the core of our protocol based on AHT, which consists of the dynamic verification protocol

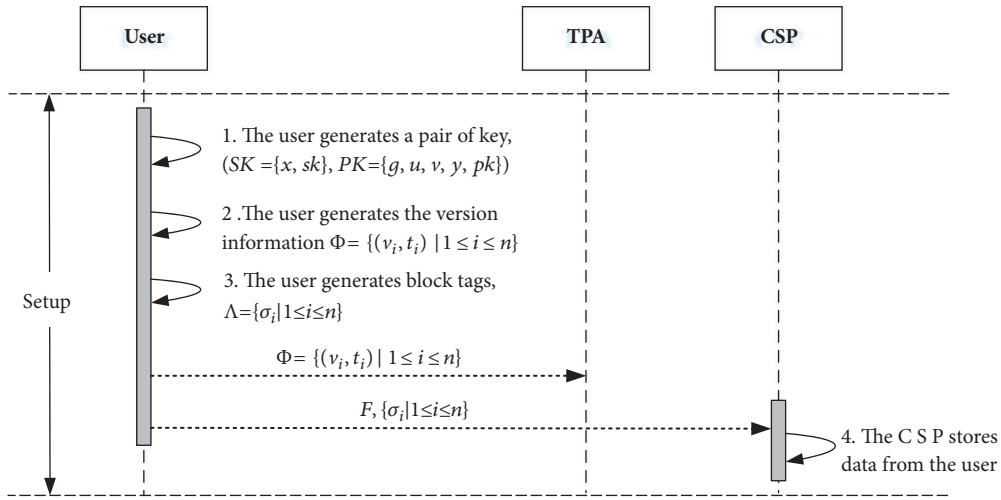


Figure 3: The workflow of the setup phase.

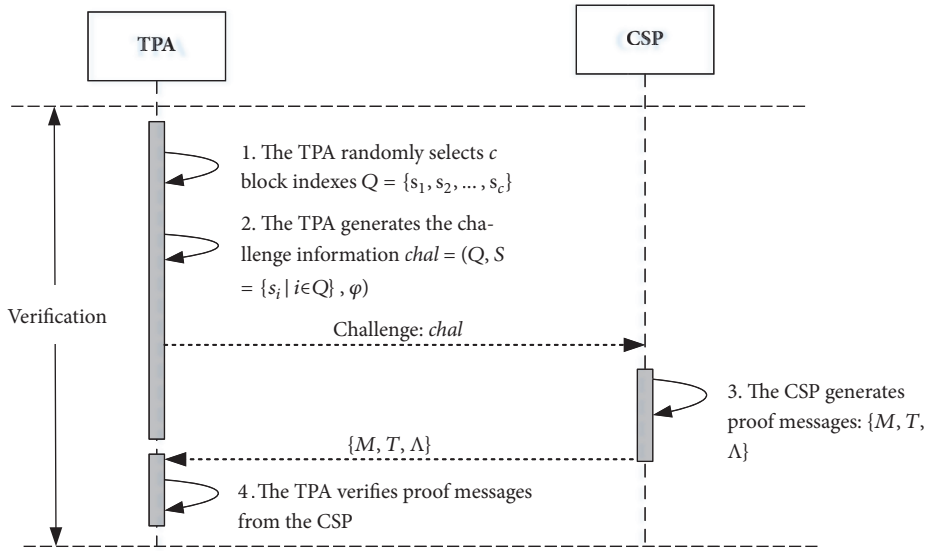


Figure 4: The workflow of the verification phase.

with privacy protection described in Section 4.1, the updating operations detailed in Section 4.2, and the batch verification protocol in Section 4.3.

4.1. Dynamic Verification with Privacy Preserving. Let $\mathbb{G}$, $\mathbb{G}_T$ be multiplicative cyclic groups of a large prime order p, and g be the generator of $\mathbb{G}$. A map function e is defined as $\mathbb{G} \times \mathbb{G} \longrightarrow \mathbb{G}_T$. $H(\cdot) : \{0,1\}^* \longrightarrow \mathbb{Z}_p$ is a secure hash function. The file outsourced to the cloud is denoted as F, which is divided into n blocks, namely, $F = \{m_1, m_2, \ldots, m_n\}$. Our dynamic auditing protocol involves the following algorithms: *KeyGen* and *TagGen* in the setup phase (see Figure 3), and *Challenge*, *ProofGen*, and *Verify* in the verification phase (see Figure 4).

KeyGen (Key Generation). The user performs *KeyGen* to generate public and secret keys, ($PK = \{g, y, v, u, pk\}$, SK = $\{x, sk\}$), where (pk, sk) is a key pair for signature, $v, u \in \mathbb{G}$ are the random elements, $x \in \mathbb{Z}_p$ is a random number, and $y = g^x$.

TagGen (Tag Generation). For each block m_i ($i = 1, 2, \ldots, n$), the user generates the signature σ_i:

$$\sigma_i = \left(u^{H(v_i \| t_i)} \cdot v^{m_i}\right)^x, \tag{5}$$

where v_i is the version number of m_i, and t_i is its time stamp, and '$\|$' is the connection operation. This signature, called block tag [16], should be uploaded to the cloud for verification along with the corresponding block. All version information (i.e., v_i and t_i) will be sent to the TPA for storing them in the

AHT. Moreover, the user generates a file tag ϑ to ensure the integrity of the file identifier ID:

$$\vartheta = ID \parallel SIG(sk, ID), \tag{6}$$

where $SIG(sk, ID)$ is the signature on ID with sk and sends it along with file identifier ID to the CSP.

Challenge. The TPA first retrieves the file tag ϑ and verifies the signature $SIG(sk, ID)$ with the public key pk. If the verification is failed, the TPA directly stops the verification by emitting FALSE; otherwise, it generates the following information: $chal = (Q = \{idx_i \mid 1 \le i \le c, c \le n\}, S = \{s_i \mid i \in Q\}, \varphi)$, where $Q = \{idx_i \mid 1 \le i \le c\}$ is the set of the block indexes to be verified, $S = \{s_i \mid i \in Q\}$ is the set of random numbers, and s_i is randomly selected form $\mathbb{Z}_p, \varphi = g^\tau$, and $\tau \in \mathbb{Z}_p$ is the random number. In particular, the TPA computes $\eta = y^\tau$ for the verification. Upon completion, the TPA sends the challenge information $chal$ to the CSP.

ProofGen (Proof Generation). While receiving the challenge, the CSP would produce a response proof for the verification, which consists of the tag proof, the block proof, and an auxiliary auditing factor. For the challenged block, the CSP generates the tag proof,

$$T = \prod_{i \in Q} e(\sigma_i, \varphi)^{s_i}, \tag{7}$$

and the block proof

$$M = \sum_{i \in Q} m_i \cdot s_i + r, \tag{8}$$

where $r \in \mathbb{Z}_p$, called random mask, is used for protecting the data privacy. Moreover, the CSP calculates the auxiliary auditing factor

$$\Lambda = e(v, y)^{-r}. \tag{9}$$

Upon completion, the CSP sends (T, M, Λ) back to the TPA as the response for the challenge.

Verify. To verify the response messages returned from the CSP, the TPA can perform the following equation:

$$\Lambda^\tau \cdot e\left(u^{\sum_{i \in Q} H(v_i \| t_i) \cdot s_i} \cdot v^M, \eta\right) = T. \tag{10}$$

If it holds, the algorithm outputs TRUE, otherwise, FALSE.

The correctness of the above verification equation can be demonstrated as follows.

$$\begin{aligned} &\Lambda^\tau \cdot e\left(u^{\sum_{i \in Q} H(v_i \| t_i) \cdot s_i} \cdot v^M, \eta\right) \\ &= e(v, \eta)^{-r} \cdot e\left(u^{\sum_{i \in Q} H(v_i \| t_i) \cdot s_i} \cdot v^{\sum_{i \in Q} m_i \cdot s_i + r}, \eta\right) \\ &= e\left(u^{\sum_{i \in Q} H(v_i \| t_i) \cdot s_i} \cdot v^{\sum_{i \in Q} m_i \cdot s_i + r} \cdot v^{-r}, \eta\right) \\ &= \prod_{i \in Q} e\left(\left(u^{H(v_i \| t_i)} \cdot v^{m_i}\right)^{x \cdot s_i}, \varphi\right) = \prod_{i \in Q} e(\sigma_i, \varphi)^{s_i} = T \end{aligned} \tag{11}$$

4.2. Dynamic Updating. To support the efficient updating operations for data blocks and files, we design the AHT in our protocol. The specific operations of block consist of block modification ($\mathscr{B}_{modify}$), block insertion ($\mathscr{B}_{insert}$), and block deletion ($\mathscr{B}_{delete}$) as follows.

Block Modification. Suppose the i-th block m_i of the file F will be modified to m'_i. The user first generates the corresponding version information (v'_i, t'_i) and then sends $U_{\text{TPA}} = (F, \mathscr{B}_{modify}, i, v'_i, t'_i)$ to the TPA. Upon receipt, the TPA updates the AHT. Simultaneously, the user generates the new signature σ'_i for m'_i according to (5) and then sends $U_{\text{CSP}} = (F, \mathscr{B}_{modify}, i, m'_i, \sigma'_i)$ to the CSP. Upon receiving, the CSP directly modifies m_i and σ_i as indicated.

Block Insertion. Suppose a new block m^* of the file F will be inserted after m_i. The user first generates the corresponding version information (v^*, t^*) and sends an insertion request $U_{\text{TPA}} = (F, \mathscr{B}_{insert}, i, v^*, t^*)$ to the TPA. Upon receiving, the TPA performs the insertion request as indicated in the AHT. Meanwhile, the user generates the new signature σ^* for m^* according to (5) and then sends $U_{\text{CSP}} = (F, \mathscr{B}_{modify}, i, m^*, \sigma^*)$ to the CSP. Once receiving the request, the CSP inserts the new block m^* after m_i and the new tag σ^* behind σ_i.

Block Deletion. Suppose the i-th block m_i of the file F will be deleted. The user sends a deletion request $U_{\text{TPA}} = (F, \mathscr{B}_{delete}, i)$ to the TPA. Upon receipt, the TPA executes the deletion request to delete the corresponding version information in the AHT. Moreover, the user sends a deletion request $U_{\text{CSP}} = (F, \mathscr{B}_{delete}, i)$ to the CSP. Upon receiving, the CSP directly deletes m_i and σ_i as indicated.

The updating operations on file include the file appending and the file deletion, which are very straightforward. We suppose that a new file F^* will be appended. The user needs to execute the algorithm *TagGen* once again. Moreover, while deleting a file F, the user first sends deletion instructions to the TPA and the CSP, respectively. Once receiving the requests, the TPA will delete the file element and its corresponding AT in the AHT, and the CSP will delete the file F and all of its tags.

4.3. Batch Verification. In reality, the TPA may simultaneously handle multiple audit tasks from different users' delegations. To achieve the minimum communication and computation costs, the batch auditing is introduced to deal with multiple auditing tasks from various users' delegations.

Suppose that the TPA sends w challenges for w users' delegations to the CSP. Once receipt, the CSP first calculates the tag proof (T_k), the data proof (M_k), and the auxiliary auditing factor (Λ_k) and then computes the aggregate tag proof T_{B} according to the following equations.

$$T_{\text{B}} = \prod_{k=1}^{w} T_k. \tag{12}$$

Finally, the CSP responds with $(T_{\text{B}}, \{M_k, \Lambda_k\}_{1 \le k \le w})$.

Table 2: Communication costs comparison.

Protocols	Verification phase	Updating phase
DPDP(skip list)[6]	$cO(\log n)$	$O(\log n)$
DPDP(MHT)[7]	$cO(\log n)$	$O(\log n)$
IHT-PA[10]	$O(c)$	$O(1)$
DHT-PA[13]	$O(c)$	$O(1)$
DLIT-PA[14]	$O(c)$	$O(1)$
AHT-PA	$O(c)$	$O(1)$

To verify the response messages, the TPA checks if the following equation holds:

$$\prod_{k=1}^{w}\left(\Lambda_k^{\tau_k}\cdot e\left(u_k^{\sum_{i\in Q_k} H(v_{k,i}\|t_{k,i})\cdot s_{k,i}}\cdot v_k^{M_k},\eta_k\right)\right)=T_{\mathrm{B}}, \tag{13}$$

where $v_{k,i}$ and $t_{k,i}$ are the version number of m_i and its time stamp for the k-th user, u_k, and v_k are the public keys of the k-th user, and τ_k, η_k, $Q_k = \{idx_{k,i} \mid 1 \le i \le c\}$ and $S_k = \{s_{k,i} \mid i \in Q_k\}$ belong to the challenge information for the k-th user. If (13) holds, the integrity of all the challenged files can be ensured. Otherwise, one or some of them are corrupted. The correctness of the above batch verification can be demonstrated as follows:

$$\begin{aligned}\prod_{k=1}^{w}\left(\Lambda_k^{\tau_k}\cdot e\left(u_k^{\sum_{i\in Q_k} H(v_{k,i}\|t_{k,i})\cdot s_{k,i}}\cdot v_k^{M_k},\eta_k\right)\right)&=\prod_{k=1}^{w}T_k\\ &=T_{\mathrm{B}}\end{aligned} \tag{14}$$

5. Security Analysis

We will evaluate the security of the presented protocol with proofs of the following theorems.

Theorem 1 (unforgeability of BLS-HVA). *In our protocol, it is computationally infeasible for any adversary to forge a valid BLS-HVA if the computational Diffe-Hellman (CDH) assumption in bilinear groups holds.*

Proof. As demonstrated in the security analysis of [21], the BLS-HVA is effectively unforgeable when the CDH problem is hard in bilinear groups [31]. Thus, the proof is omitted here. □

Theorem 2 (unforgeability of proof). *The presented protocol can efficiently resist the forging attacks generated by the CSP. In other words, it is impossible for the CSP to forge effective proofs to pass the auditing verification.*

Proof. To respond for a challenge, the CSP sends a proof message (T, M, Λ) back to the TPA. If the auxiliary auditing factor Λ is fake, the verification equation (10) does not hold, even though the other proofs are valid. As demonstrated in Theorem 1, BLS-HVAs are unforgeable. Therefore, the tag proof T cannot be forged. Finally, we just need to prove that the block proof M is unforgeable.

To prove this, we first define the following game: The TPA sends a fake proof message $P^* = (T, M^*, \Lambda)$, where

$$M=\sum_{i\in Q} m_i\cdot s_i + r \neq M^*=\sum_{i\in Q} m_i^*\cdot s_i + r. \tag{15}$$

If the CSP can still pass the verification, then he/she wins this game; otherwise, he/she does not. Assume that the CSP wins this game, then

$$\begin{aligned}&\Lambda^{\tau}\cdot e\left(u^{\sum_{i\in Q} H(v_i\|t_i)\cdot s_i}\cdot v^{M^*},\eta\right)\\ &\quad=\Lambda^{\tau}\cdot e\left(u^{\sum_{i\in Q} H(v_i\|t_i)\cdot s_i}\cdot v^{\sum_{i\in Q} m_i^*\cdot s_i+r},\eta\right).\end{aligned} \tag{16}$$

Moreover, for the valid proofs, we have

$$\begin{aligned}&\Lambda^{\tau}\cdot e\left(u^{\sum_{i\in Q} H(v_i\|t_i)\cdot s_i}\cdot v^{M},\eta\right)\\ &\quad=\Lambda^{\tau}\cdot e\left(u^{\sum_{i\in Q} H(v_i\|t_i)\cdot s_i}\cdot v^{\sum_{i\in Q} m_i\cdot s_i+r},\eta\right).\end{aligned} \tag{17}$$

According to the properties of bilinear maps, we can derive that

$$\sum_{i\in Q} m_i\cdot s_i + r=\sum_{i\in Q} m_i^*\cdot s_i + r, \tag{18}$$

which contradicts the above assumption. That is to say, the block proof is unforgeable. This accomplishes the proof of the theorem. □

The security of our protocol for resisting replacing and replay attacks is similar to the work [27]. Thus, we omit the corresponding proofs here.

6. Performance Evaluation

In this section, we will evaluate the performance of our protocol (AHT-PA) and compare it with the state of the arts.

6.1. Communication Costs. In this section, the communication costs of AHT-PA protocol are analyzed and compared during the verification phase (i.e., challenge and response) and updating phase. In the verification phase, the challenge and response messages between the TPA and the CSP bring communication overhead of $O(c)$, where c is denoted as the number of challenged blocks. Moreover, during the updating phase, the user should send an updating request to the CSP and the TPA, respectively, which costs $O(1)$.

Table 2 presents the communication costs of some protocols during the verification phase and updating phase, where

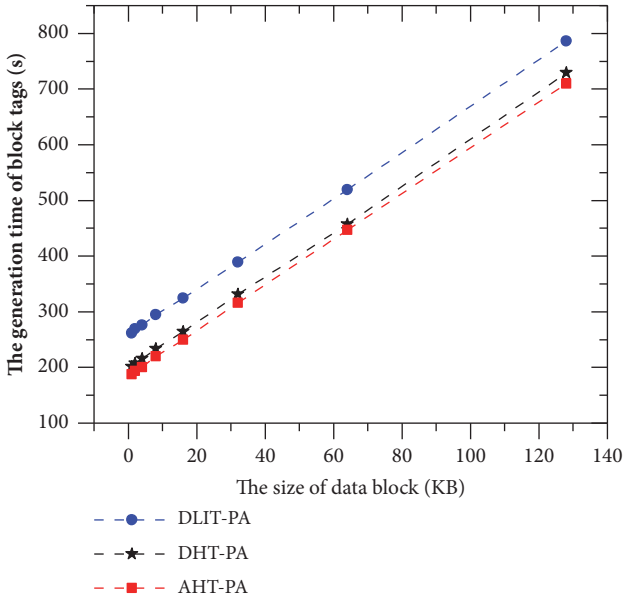


Figure 5: The tag generation time for different block sizes (the number of data blocks = 50000).

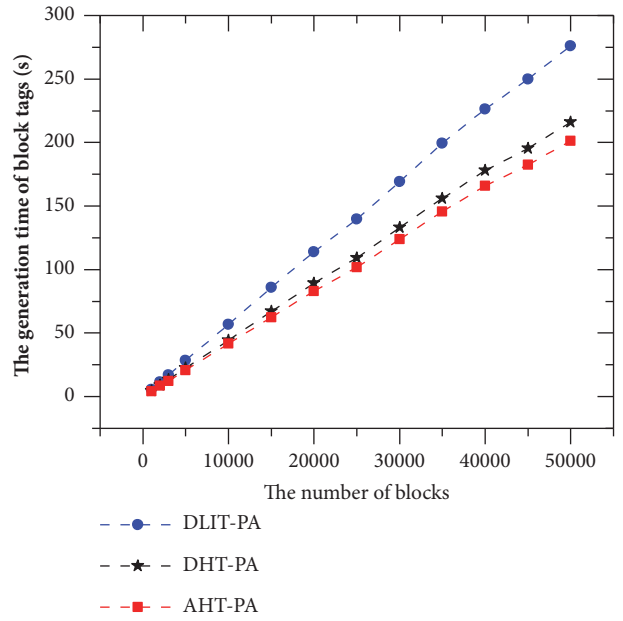


Figure 6: The time of tag generation for different numbers of blocks (block size = 4KB).

n represents the number of data blocks in a given file and c is the number of challenged blocks. Obviously, our protocol only requires a small amount of communication overhead and is substantially much efficient than the other protocols.

6.2. Computational Costs. The computational costs of AHT-PA protocol are evaluated and presented in this section. Thus, we implement all algorithms in the AHT-PA based on Pairing Base Cryptography (PBC) Library (0.5.14). The algorithms in experiments are evaluated on a DELL workstation with an Intel Xeon E3-1225v5 3.30 GHz, 16 GB DDR4-2133 ECC (2x8GB) RAM, and 2TB 7200 RPM SATA 1st HDD. We run the programs under a Linux (ubuntu 16.04.2 LTS x64) operating system, whose kernel version is 4.8.0 and use a MNT d159 curve, which has a 160-bit group order. The final results are the averages of 20 runs.

Computational Costs for Generating Tags. Figures 5 and 6 indicate the comparison results of the time of tag generation for different block sizes and for different numbers of data blocks, respectively, from which we can learn that (1) the generation time for the user is proportional to the block size or block number; (2) to deal with the same block size or same block number in the above two scenarios, AHT-PA takes less time than DHT-PA and DLIT-PA. In other words, the computation overhead of tag generation in AHT-PA is less than those in DHT-PA and DLIT-PA.

Computational Costs for Verification. Figure 7 indicates the experimental results of the verification time for different numbers of challenged blocks, from which we can learn that the verification time increases rapidly with the number of challenge blocks in the DHT-PA and DLIT-PA, but the verification time of AHT-PA remains stable and is much less than the previous two protocols.

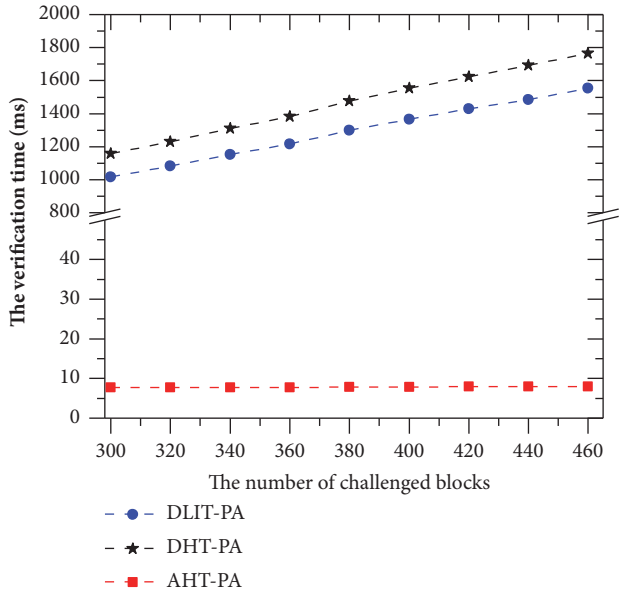


Figure 7: The verification time for different numbers of challenged blocks (the number of total data blocks = 50000, the block size = 4KB).

Computational Costs for Batch Auditing. In the batch auditing scenario, we will evaluate the performance of AHT-PA and compare it with DHT-PA and DLIT-PA. The comparison results, as shown in Figure 8, demonstrate that (1) three protocols can simultaneously handle various audits from multiple users and (2) at the same number of auditing tasks, the average audit time per task in AHT-PA is significantly less than in DHT-PA and DLIT-PA. That is to say, the batch auditing protocol in AHT-PA is much more efficient than those in DHT-PA and DLIT-PA.

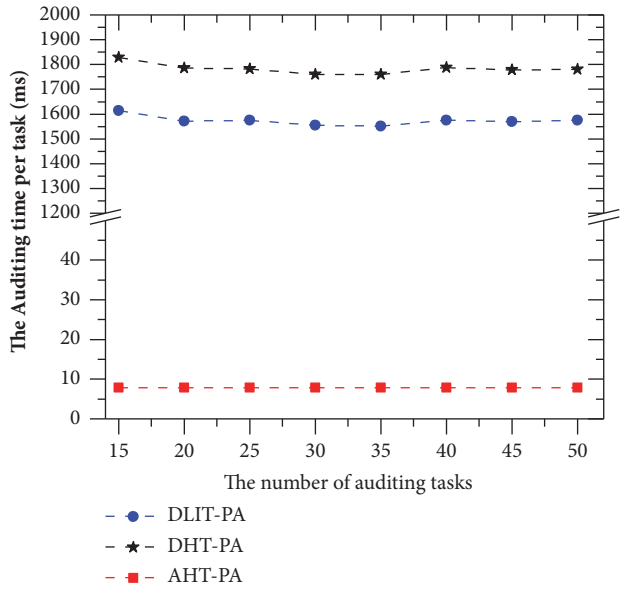


FIGURE 8: The average audit time per task for the various numbers of auditing tasks (the number of challenged blocks = 460, the total number of data blocks for each file = 50000, and the block size = 4KB).

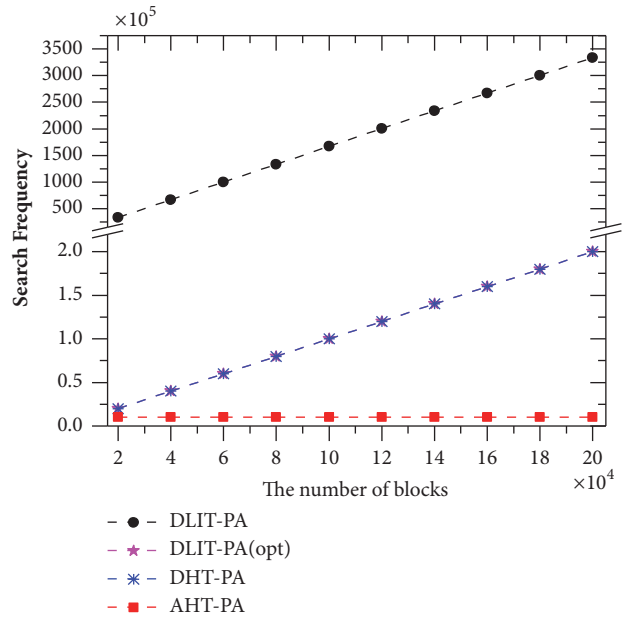


FIGURE 9: The search frequency under 5000 challenged blocks.

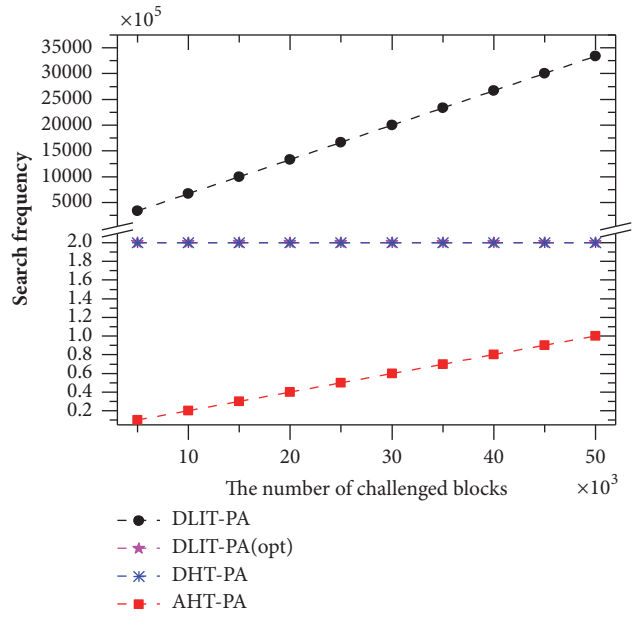


FIGURE 10: The search frequency under 2×10^5 data blocks.

Search Efficiency. We design two experiments on a single file to evaluate block-search efficiency of AHT-PA. The first experiment is performed under various total numbers of data blocks from 2×10^4 to 2×10^5 with 5000 challenged blocks, and another is performed under various numbers of challenged blocks with 2×10^5 data blocks. In the two experiments, we add an extra comparison item for DLIT-PA, named DLIT-PA (opt), whose results were obtained by first sorting the index set of challenged blocks to get its ascending set and then counting the corresponding frequency for searching block elements. As is well known, to locate a block element in the single linked list or double linked list, it is necessary to visit the whole list from the first element in the first search round. After that, if we previously sort the index set of the required blocks, the searching element in the single linked list or double linked list can start from the current element to the next element behind the current element. In this sense, in terms of search frequency, DLIT-PA (opt) is more efficient than DLTI-PA and identical to DHT-PA.

The results of the experiment under 5000 challenged blocks are shown in Figure 9, from which we can learn that AHT-PA outperforms the other protocols. Moreover, the larger the number of data blocks, the greater the search frequency gap. Figure 10 gives the experimental results under 2×10^5 data blocks. Apparently, the search frequency in AHT-PA slowly rises with the increasing number of challenged blocks. However, for the same number of challenged blocks, the search frequency in AHT-PA is much less than the ones in other protocols.

In summary, the public auditing protocol proposed in this paper can achieve better performance. To be specific, in the tag generation phase and verification phase, the computation cost is less than those of the state of the arts, while achieving better block-search efficiency in the verification phase and the updating phase.

7. Conclusions

In this paper, we present a novel auditing protocol for cloud storage. Differing from the state of the arts, we design a new structure, called adjacency-hash table, to support efficient data updating as well as reduce computational costs. Moreover, to achieve privacy preserving, our protocol employs the random masking technique to prevent the TPA from

learning the users' data contents in the verification phase. Sufficient formal proofs indicate that our protocol is secure. The theoretical analysis and the experimental results show that our protocol is feasible and efficient and outperforms the state of the arts in both the computation overhead and the communication costs.

Data Availability

The 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

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. U1405254 and U1536115, the Natural Science Foundation of Fujian Province of China under Grant No. 2018J01093, the Program for New Century Excellent Talents in Fujian Province University under Grant No. MJK2016-23, the Program for Outstanding Youth Scientific and Technological Talents in Fujian Province University under Grant No. MJK2015-54, the Promotion Program for Young and Middle-Aged Teachers in Science and Technology Research of Huaqiao University under Grant No. ZQN-PY115, the Opening Project of Shanghai Key Laboratory of Integrated Administration Technologies for Information Security under Grant No. AGK201710, Subsidized Project for Postgraduates' Innovative Fund in Scientific Research of Huaqiao University, and Program for Science & Technology Innovation Teams and Leading Talents of Huaqiao University under Grant No. 2014KJTD13.

References

[1] H. Wang, Z. Zheng, L. Wu, and P. Li, "New directly revocable attribute-based encryption scheme and its application in cloud storage environment," *Cluster Computing*, vol. 20, no. 3, pp. 2385–2392, 2017.

[2] M. N. O. Sadiku, S. M. Musa, and O. D. Momoh, "Cloud computing: Opportunities and challenges," *IEEE Potentials*, vol. 33, no. 1, pp. 34–36, 2014.

[3] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, 2009.

[4] K. Yang and X. Jia, "Data storage auditing service in cloud computing: challenges, methods and opportunities," *World Wide Web*, vol. 15, no. 4, pp. 409–428, 2012.

[5] Z. Xia, X. Wang, X. Sun, Q. Liu, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 340–352, 2016.

[6] Z. Fu, X. Sun, Q. Liu, L. Zhou, and J. Shu, "Achieving efficient cloud search services: multi-keyword ranked search over encrypted cloud data supporting parallel computing," *IEICE Transactions on Communications*, vol. E98B, no. 1, pp. 190–200, 2015.

[7] J. Shen, T. Zhou, X. Chen, J. Li, and W. Susilo, "Anonymous and Traceable Group Data Sharing in Cloud Computing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 4, pp. 912–925, 2018.

[8] J. Li, Y. K. Li, X. Chen, P. P. C. Lee, and W. Lou, "A Hybrid Cloud Approach for Secure Authorized Deduplication," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 5, pp. 1206–1216, 2015.

[9] J. Li, X. Chen, X. Huang et al., "Secure distributed deduplication systems with improved reliability," *IEEE Transactions on Computers*, vol. 64, no. 12, pp. 3569–3579, 2015.

[10] Q. Jiang, J. Ma, and F. Wei, "On the security of a privacy-aware authentication scheme for distributed mobile cloud computing services," *IEEE Systems Journal*, 2016.

[11] D. A. B. Fernandes, L. F. B. Soares, J. V. Gomes, M. M. Freire, and P. R. M. Inácio, "Security issues in cloud environments: A survey," *International Journal of Information Security*, vol. 13, no. 2, pp. 113–170, 2014.

[12] L. F. B. Soares, D. A. B. Fernandes, J. V. Gomes, M. M. Freire, and P. R. M. Inácio, "Cloud security: State of the art," *Security, Privacy and Trust in Cloud Systems*, vol. 9783642385865, pp. 3–44, 2014.

[13] Z. Fu, X. Wu, C. Guan, X. Sun, and K. Ren, "Toward Efficient Multi-Keyword Fuzzy Search over Encrypted Outsourced Data with Accuracy Improvement," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 12, pp. 2706–2716, 2016.

[14] Q. Jiang, M. K. Khan, X. Lu, J. Ma, and D. He, "A privacy preserving three-factor authentication protocol for e-health clouds," *Journal of Supercomputing*, vol. 72, no. 10, pp. 3826–3849, 2016.

[15] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," *IEEE Transactions on Computers*, vol. 62, no. 2, pp. 362–375, 2013.

[16] G. Ateniese, R. Burns, R. Curtmola et al., "Provable data possession at untrusted stores," in *Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07)*, pp. 598–609, Virginia, Va, USA, November 2007.

[17] A. Juels and B. S. Kaliski Jr., "Pors: proofs of retrievability for large files," in *Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07)*, pp. 584–597, ACM, Alexandria, VA, USA, November 2007.

[18] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward secure and dependable storage services in cloud computing," *IEEE Transactions on Services Computing*, vol. 5, no. 2, pp. 220–232, 2012.

[19] C. Erway, A. Küpçü, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in *Proceedings of the Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS '09)*, pp. 213–222, ACM, Chicago, Ill, USA, November 2009.

[20] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 5789, pp. 355–370, 2009.

[21] Q.-A. Wang, C. Wang, K. Ren, W.-J. Lou, and J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 5, pp. 847–859, 2011.

[22] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 9, pp. 1717–1726, 2012.

[23] Y. Zhu, G.-J. Ahn, H. Hu, S. S. Yau, H. G. An, and C.-J. Hu, "Dynamic audit services for outsourced storages in clouds," *IEEE Transactions on Services Computing*, vol. 6, no. 2, pp. 227–238, 2013.

[24] C. Liu, J. Chen, L. T. Yang et al., "Authorized public auditing of dynamic big data storage on cloud with efficient verifiable fine-grained updates," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 9, pp. 2234–2244, 2014.

[25] H. Jin, H. Jiang, and K. Zhou, "Dynamic and Public Auditing with Fair Arbitration for Cloud Data," *IEEE Transactions on Cloud Computing*, vol. 6, no. 3, pp. 680–693, 2018.

[26] H. Tian, Y. Chen, C.-C. Chang et al., "Dynamic-Hash-Table Based Public Auditing for Secure Cloud Storage," *IEEE Transactions on Services Computing*, vol. 10, no. 5, pp. 701–714, 2017.

[27] J. Shen, J. Shen, X. Chen, X. Huang, and W. Susilo, "An efficient public auditing protocol with novel dynamic structure for cloud data," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 10, pp. 2402–2415, 2017.

[28] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in *Proceedings of the IEEE INFO-COM*, pp. 525–533, March 2010.

[29] H. Shacham and B. Waters, "Compact proofs of retrievability," in *Advances in Cryptology—ASIACRYPT 2008: Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security, Melbourne, Australia, December 2008*, vol. 5350 of *Lecture Notes in Computer Science*, pp. 90–107, Springer, Berlin, Germany, 2008.

[30] B. Wang, B. Li, and H. Li, "Panda: Public auditing for shared data with efficient user revocation in the cloud," *IEEE Transactions on Services Computing*, vol. 8, no. 1, pp. 92–106, 2015.

[31] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in *Advances in Cryptology—ASIACRYPT 2001*, vol. 2248 of *Lecture Notes in Computer Science*, pp. 514–532, Springer, Berlin, Germany, 2001.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 7209475, 9 pages
https://doi.org/10.1155/2018/7209475

Research Article

A Fog Computing Security: 2-Adic Complexity of Balanced Sequences

Wang Hui-Juan and Jiang Yong

The Information Security Department, The First Research Institute of the Ministry of Public Security of P.R.C., Beijing 100084, China

Correspondence should be addressed to Wang Hui-Juan; whj409@163.com

Received 9 January 2018; Accepted 5 March 2018; Published 9 September 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Wang Hui-Juan and Jiang Yong. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In the fog computing environment, the periodic sequence can provide sufficient authentication code and also reduce the power consumption in the verification. But the periodic sequence faces a known full-cycle attack threat in fog computing. This paper studies the 2-adic complexity attack ability of the periodic balance sequence in the fog computing environment. It uses the exponential function as a new approach to study the 2-adic properties of periodic balance sequence and presents that the 2-adic complexity of the periodic balanced sequence is not an attacking threat when used in fog computing.

1. Introduction

Fog computing is a decentralized computing architecture compared to cloud computing and is currently used primarily for mobile and portable devices. Due to the current proliferation of IoT devices, the main advantage of fog computing is the ability to quickly provide scalable, decentralized solutions. Between data sources and cloud infrastructure, fog computing mainly processes and stores data. Fog computing can improve computational performance by reducing the amount of processing and storage that extra data consumes. Fog computing has real-time responsiveness and offers a cost-effective, flexible deployment of hardware and software in computing system deployments. Fog platform also faces a lot of network security issues. Such as code injection attacks (such as SQL injection), session and cookie hijacking (posing as legitimate users), illegal direct data access unsafe references, malicious redirect and driver attacks, web attacks, and other cyberattacks. Due to the relatively small computing resources (memory, processing, and storage) of the fog computing system, there is no security protection that can consume a large amount of secure authentication storage as cloud computing does. Fog computing should be defined for a broader range of ubiquitous connected devices, which requires the fog server to generate a large number of security codes at one time and a relatively low computational load during verification. For secure communications and authentication, stream ciphers are recognized as fast certification, which require less computation and storage capacity. AES-based cipher type mentioned in [1] is an encryption algorithm which is suitable for fog platforms. But fog calculation of data encryption security needs to consider stream cipher antiattack performance. In the fog computing environment using stream ciphers, the security verification and data transmission should be considered between the length of the password and the verification algorithm. The safety of some fog calculations strongly depends on the security of the sequence itself by weakening the verification algorithm. In this case, the fog server will distribute a large amount of security codes, and it is easier for an attacker to collect large numbers of plain-texts and cipher-texts so that he may filter out full-period encrypted sequences. Currently, there are many attacks on the known periodic sequences in which a common one is the 2-adic complexity attack.

For cryptographic applications, a good pseudorandom generator must be infeasible to find the corresponding initial state. Hence many modern stream ciphers are designed by combining the output sequences in various nonlinear ways. Goresky and Klapper first introduced feedback with carry shift registers (FCSRs) as shown in Figure 1 , which are a

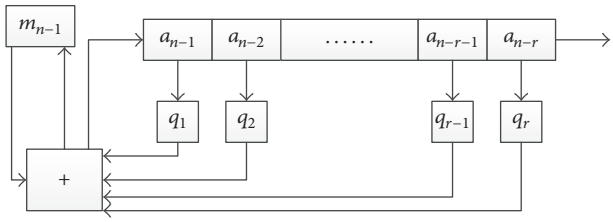


Figure 1: Feedback with carry shift registers.

Begin
Input a until the first nonzero a_{k-1} *is found*
$\alpha = a_{k-1} \cdot 2^{k-1}$
$f = (0, 2)$
$g = (2^{k-1}, 1)$
While there are more bits do
  In put a new bit a_k
$\alpha = \alpha + a_k \cdot 2^k$
  If $\alpha g_2 - g_1 \equiv 0 \ (mod\ 2^{k+1})$ *then*
$f = 2f$
  else if $\Phi(g) < \Phi(f)$ *then*
    Let d be odd and minimize $\Phi(f + dg)$
$\langle g, f \rangle = \langle f + dg, 2g \rangle$
  else
    Let d be odd and minimize $\Phi(g + df)$
$\langle g, f \rangle = \langle g + df, 2f \rangle$
  fi fi
$k = k + 1$
odd
Return g
End

Algorithm 1: Rational approximation algorithm.

class of nonlinear sequence generators by [2], and used the arithmetic in the 2-adic number to analyze this stream generator. For the security of the stream, rational approximation algorithm given in [2] is an important adaptive synthesizing algorithm against FCSRs, as shown in Algorithm 1, by which if a key-stream can be generated by a short FCSR, then this FCSR can be efficiently determined from a small subsequence of the key-stream. Therefore, the rational approximation algorithm sets up a new measure of key-stream security and is referred to as 2-adic complexity. For the properties of FCSRs, it is well known that any strictly periodic sequence can be generated by an FCSR. Then any binary sequence with low 2-adic complexity is insecure for cryptographic applications. Although some properties of 2-adic complexity had been proven, such as the expected value and variance of 2-adic complexities of periodic binary sequences and the 2-adic complexity of m-sequence, the 2-adic complexity of binary sequences has not been quite clear. This paper studies one function of periodic balance sequence which can against the 2-adic complexity attack in the fog computing environment.

This paper involves the exponential function and the structure principle of FCSR for the study of the 2-adic properties and 2-adic complexity of balanced binary sequences. For a binary balanced periodic sequence, we give a relationship with its 2-adic integer, the length of period, and 2-adic complexity and show that the 2-adic complexity is bigger than the half period of the sequence when its 2-adic number approaches half. Moreover, it is indicated that the 2-adic complexity of the binary balanced sequence is affected by the register bit values of the FCSR. In the following sections we only consider the binary strictly periodic sequences, and we denote them as periodic sequences for simplicity.

2. Preliminary

In this section we briefly review some basic facts about feedback with carry shift register (FCSR) and 2-adic number. The FCSR is a feedback with r-stages shift register and its auxiliary memory contained nonnegative integer. Assume an odd integer q has the binary representation as $q + 1 = q_1 \cdot 2 + q_2 \cdot 2^2 + \cdots + q_r \cdot 2^r$. Then the r-stages connections of FCSR are given by the bits $\{q_1, q_2, \ldots, q_r\}$. The FCSR with connection integer q is described as follows:

(1) Take an integer sum $\sigma_t = \sum_{k=1}^{r} q_k a_{t-k} + m_{t-1}$.

(2) Shift the contents one step to the right, outputting the right bit a_{t-r}.

(3) Place $a_t = (\sigma_t) \bmod 2$ into the left most cell of the shift register.

(4) Replace the memory integer m_{t-1} with $m_t = (\sigma_t - a_t)/2 = \lfloor \sigma_t/2 \rfloor$.

The number of bits in the connection number coincides with the size of the basic register. For strictly periodic sequences, the extra memory is small and we can ignore it, but the eventually periodic sequence may require the amount of memory. In this paper, we just consider the strictly periodic sequences, and then we denote that the 2-adic complexity of sequences is to measure the number of bits in the basic FCSR. In the study of the output sequence of a given FCSR, we usually use the arithmetic in the 2-adic integer.

A 2-adic integer is form power series $\alpha = \sum_{t=0}^{\infty} a_t \cdot 2^t$, with $a_t \in \{0, 1\}$, and a fact is that number -1 is represented by $-1 = 1 + 2 + 2^2 + 2^3 + \cdots$. Then, the negative integer $-q$ is associated with the product

$$-q = \left(1 + 2 + 2^2 + 2^3 + \cdots\right) \cdot \left(q_0 + q_1 \cdot 2 + q_2 \cdot 2^2 + \cdots + q_r \cdot 2^r\right). \quad (1)$$

Moreover, the multiplication of 2-adic integer also has unique inverse if the integer q is an odd integer. Thus the 2-adic integer contains every rational number p/q, provided q is odd.

Proposition 1 (see [2]). *There is a one-to-one correspondence between rational numbers* $\alpha = p/q$ *(where q is odd) and eventually periodic binary sequences* $\underline{a}$*. We define the rational number* α *as the 2-adic expansion of the binary sequences* $\underline{a}$*. The sequence* $\underline{a}$ *is strictly periodic if and only if* $\alpha \leq 0$ *and* $|\alpha| < 1$.

If a strictly sequence $\underline{a}$ is generated by an FCSR with connection integer q, then the 2-adic integer $\alpha = a_0 + a_1 \cdot 2 + a_2 \cdot 2^2 + \cdots$ of binary sequence $\underline{a}$ has the following association.

Proposition 2 (see [2]). *Let a periodic sequence $\underline{a} = a_0, a_1, a_2, \ldots$ be generated by an FCSR with connection integer q and the 2-adic representation of sequence $\underline{a}$ is $-p/q$. Then one has*

$$\frac{\sum_{t=0}^{T-1} a_t 2^t}{2^T - 1} = \frac{p}{q}. \tag{2}$$

From the above description about 2-adic integer and FCSR, the 2-adic complexity of periodic sequence $\underline{a}$ can be regarded as $\psi(\underline{a}) = \lfloor \log_2^q \rfloor$.

The binary sequences of 2-adic complexity can be got from rational approximation algorithm [2]. If the 2-adic complexity of a sequence is greater than half the period, then this sequence is resistant to 2-adic rational approximation attacks.

3. Main Results

In this section we mainly prove Theorem 7, and some lemmas are given to support the main result proof.

Lemma 3 (see [2]). *Suppose a periodic sequence $\underline{a} = a_0, a_1, a_2, \ldots$ is generated by an FCSR with connection integer q. Let $r = 2^{-1} \in Z/(q)$ be the (multiplicative) inverse of 2 in the ring $Z/(q)$ of integer modulo q. Then there exists $A \in Z/(q)$ such that, for all $t = 0, 1, 2, \ldots$, one has $a_t = (A \cdot r^t \bmod q) \bmod 2$.*

In this paper, we just consider the balanced binary strictly periodic sequence. Then the sequence $\underline{u} = \{u_t = p \cdot 2^{-t} \bmod q\}_{t=0}^{\infty}$ in a period of length T satisfies the fact that the number of even integers equals the number of odd integers. We assume that another sequence $\underline{v} = \{v_t\}_{t=0}^{\infty}$ over $Z/(q)$ in a period of length T is bilateral symmetry with $(q-1)/2$. In the following analysis of this paper, we introduce the exponential function $e_q(*) = e^{(2\pi i \cdot *)/q}$ as the tool to prove the main theorems. It is easy to get $e_q(q) = e^{(2\pi i \cdot q)/q} = 1$ and $e_q(q/2) = e^{(\pi i \cdot q)/q} = -1$. Since $u_t = p \cdot 2^{-t} \bmod q$, we have

$$e_q(u_t) = e^{(2\pi i \cdot u_t)/q} = e^{(2\pi i \cdot p \cdot 2^{-t})/q} = e_q\left(p \cdot 2^{-t}\right). \tag{3}$$

Lemma 4. *Let the sequence $\underline{v}$ over $Z/(q)$ be a periodic sequence as described above; one has known that the sequence $\underline{v}$ in a period of length of T satisfies the following equation:*

$$\sum_{t=0}^{T-1} \sum_{b=1}^{q-1} \sum_{x=0}^{(q-1)/2} e_q\left(b \cdot \left(\frac{v_t}{2} - x\right)\right) = \frac{T(q-1)}{4}. \tag{4}$$

Proof. In a period of length T of the sequence $\underline{v}$, the number of odd integers equals the number of even integers and T is an even. When $v_t \in Z/(q)$ is an even integer, we assume $v_t = 2k_t$ over $Z/(q)$, and we have

$$\begin{aligned} &\sum_{b=1}^{q-1} \sum_{x=0}^{(q-1)/2} e_q\left(b \cdot \left(\frac{v_t}{2} - x\right)\right) \\ &= \sum_{b=0}^{q-1} \sum_{x=0}^{(q-1)/2} e_q\left(b \cdot \left(\frac{v_t}{2} - x\right)\right) - \frac{q+1}{2} \\ &= \sum_{b=1}^{q-1} \sum_{x=0, x_t \neq k}^{(q-1)/2} e_q\left(q \cdot (k_t - x)\right) + q - \frac{q+1}{2}. \end{aligned} \tag{5}$$

Since $e_q(q \cdot (k_t - x)) = e^{(q \cdot (k_t - x) \cdot 2\pi i)/q} = 1$, then we have $\sum_{b=1}^{q-1} \sum_{x=0}^{(q-1)/2} e_q(b \cdot (v_t/2 - x)) = (q-1)/2$ for v_t as an even integer. When $v_t = 2k_t + 1$ is an odd integer, we get another v_s with $v_s = q - 1 - v_t$, and

$$\begin{aligned} &\sum_{b=0}^{q-1} \sum_{x=0}^{(q-1)/2} \left(e_q\left(b \cdot \left(\frac{v_t}{2} - x\right)\right) + e_q\left(b \cdot \left(\frac{v_s}{2} - x\right)\right)\right) \\ &= \sum_{b=0}^{q-1} \sum_{x=0}^{(q-1)/2} \left(e_q\left(b \cdot \left(\frac{v_t}{2} - x\right)\right)\right. \\ &\left. + e_q\left(b \cdot \left(\frac{q-1}{2} - \frac{v_t}{2} - x\right)\right)\right). \end{aligned} \tag{6}$$

As the variable $(q-1)/2 - x \in [0, (q-1)/2]$,

$$\begin{aligned} &\sum_{b=0}^{q-1} \sum_{x=0}^{(q-1)/2} \left(e_q\left(b \cdot \left(\frac{v_t}{2} - x\right)\right) + e_q\left(b \cdot \left(\frac{v_s}{2} - x\right)\right)\right) \\ &= \sum_{b=0}^{q-1} \sum_{x=0}^{(q-1)/2} \left(e_q\left(b \cdot \left(\frac{v_t}{2} - x\right)\right) + e_q\left(b \cdot \left(x - \frac{v_t}{2}\right)\right)\right) \\ &= \sum_{x=0}^{(q-1)/2} \frac{e_q\left(q \cdot (v_t/2 - x)\right) - 1}{e_q\left(v_t/2 - x\right) - 1} \\ &\quad + \frac{e_q\left(q \cdot (x - v_t/2)\right) - 1}{e_q\left(x - v_t/2\right) - 1}. \end{aligned} \tag{7}$$

Since $e_q(q \cdot (v_t/2 - x)) - 1 = e_q(q \cdot (x - v_t/2)) - 1 = -2$, we have

$$\begin{aligned} &\sum_{x=0}^{(q-1)/2} \frac{e_q\left(q \cdot (v_t/2 - x)\right) - 1}{e_q\left(v_t/2 - x\right) - 1} + \frac{e_q\left(q \cdot (x - v_t/2)\right) - 1}{e_q\left(x - v_t/2\right) - 1} \\ &\quad = 2 \cdot \frac{q+1}{2} = q + 1. \end{aligned} \tag{8}$$

Thus

$$\begin{aligned} &\sum_{b=1}^{q-1} \sum_{x=0}^{(q-1)/2} \left(e_q\left(b \cdot \left(\frac{v_t}{2} - x\right)\right) + e_q\left(b \cdot \left(\frac{v_s}{2} - x\right)\right)\right) \\ &= \sum_{b=0}^{q-1} \sum_{x=0}^{(q-1)/2} \left(e_q\left(b \cdot \left(\frac{v_t}{2} - x\right)\right) + e_q\left(b \cdot \left(\frac{v_s}{2} - x\right)\right)\right) \\ &\quad - q - 1 = q + 1 - q - 1 = 0. \end{aligned} \tag{9}$$

Then, from the above analysis, we get

$$\sum_{t=0}^{T-1}\sum_{b=1}^{q-1}\sum_{x=0}^{(q-1)/2} e_q\left(b\cdot\left(\frac{v_t'}{2}-x\right)\right) = \frac{T}{2}\sum_{b=1}^{q-1}\sum_{x=0}^{(q-1)/2} e_q\left(b\cdot\left(\frac{k_t}{2}-x\right)\right) = \frac{T(q-1)}{4}. \tag{10}$$

□

The sequences v_t have a little limit in Lemma 4, and the sequence v_t in a period of length T satisfies

$$\{v_t\}_{t=0}^{T-1} = \left\{\frac{q-1}{2}-\frac{T}{2}+1, \frac{q-1}{2}-\frac{T}{2}+2, \ldots, \frac{q-1}{2}+\frac{T}{2}\right\}, \tag{11}$$

when $(q-1)/2 \equiv (T/2) \bmod 2$, and v_t in a period of length T satisfies

$$\{v_t\}_{t=0}^{T-1} = \left\{\frac{q-1}{2}-\frac{T}{2}, \ldots, \frac{q-1}{2}-1\right\} \cup \left\{\frac{q-1}{2}+1, \ldots, \frac{q-1}{2}+\frac{T}{2}\right\}, \tag{12}$$

when $(q-1)/2 \equiv (T/2+1) \bmod 2$.

Lemma 5. *For any positive integer N, one has*

$$\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{N-1} e_q\left(2b\cdot\left(\frac{v_t}{2}-x\right)\right)\right| < q\left(\frac{1}{\pi}\ln^{\cot(\pi/q)}+\frac{1}{6}\right). \tag{13}$$

Proof. If $(q-1)/2 \equiv (T/2) \bmod 2$, we have

$$\{v_t\}_{t=0}^{T-1} = \left\{\frac{q-1}{2}-\frac{T}{2}+1, \frac{q-1}{2}-\frac{T}{2}+2, \ldots, \frac{q-1}{2}+\frac{T}{2}\right\}. \tag{14}$$

That is,

$$\begin{aligned}&\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{N-1} e_q\left(2b\cdot\left(\frac{v_t}{2}-x\right)\right)\right| = \left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1} e_q\left(b\cdot(v_t-2x)\right)\right| \\ &= \left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=(q-1)/2-T/2+1}^{(q-1)/2+T/2} e_q\left(b\cdot(t-2x)\right)\right| \\ &\le \sum_{b=1}^{(q-1)/2}\left|\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1} e_q\left(b\cdot(t-2x)\right)\right|.\end{aligned} \tag{15}$$

If $(q-1)/2 \equiv (T/2+1) \bmod 2$, we have

$$\{v_t\}_{t=0}^{T-1} = \left\{\frac{q-1}{2}-\frac{T}{2}, \ldots, \frac{q-1}{2}-1\right\} \cup \left\{\frac{q-1}{2}+1, \ldots, \frac{q-1}{2}+\frac{T}{2}\right\}. \tag{16}$$

That is,

$$\begin{aligned}&\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1} e_q\left(b\cdot(v_t-2x)\right)\right| \\ &= \left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=(q-1)/2-T/2+1}^{(q-1)/2+T/2} e_q\left(b\cdot(t-2x)\right) - \sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2} e_q\left(b\cdot\left(\frac{q-1}{2}-2x\right)\right)\right| \\ &= \left|\sum_{b=1}^{(q-1)/2} e_q\cdot\left(b\cdot\left(\frac{q-1}{2}-\frac{T}{2}-1\right)\right)\sum_{x=0}^{(q-1)/2}\sum_{t=1}^{T+2} e_q\left(b\cdot(t-2x)\right) + \sum_{b=1}^{(q-1)/2} e_q\left(b\cdot\left(\frac{q-1}{2}\right)+\frac{q}{2}\right)\sum_{x=0}^{(q-1)/2} e_q\left(b\cdot(-2x)\right)\right|.\end{aligned} \tag{17}$$

Since $|e_q(b\cdot((q-1)/2-T/2-1))| \le 1$ and $|e_q(b\cdot((q-1)/2)+q/2)| \le 1$,

$$\{v_t\}_{t=0}^{T-1} = \left\{\frac{q-1}{2}-\frac{T}{2}, \ldots, \frac{q-1}{2}-1\right\} \cup \left\{\frac{q-1}{2}+1, \ldots, \frac{q-1}{2}+\frac{T}{2}\right\}; \tag{18}$$

these are satisfying

$$\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1} e_q\left(b\cdot(v_t-2x)\right)\right| \le \sum_{b=1}^{(q-1)/2}\left|\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T+2} e_q\left(b\cdot(t-2x)\right)\right|. \tag{19}$$

Next we consider the formula

$$\sum_{b=1}^{(q-1)/2}\left|\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{N} e_q\left(b\cdot(t-2x)\right)\right|. \tag{20}$$

We first have the inequality

$$\left|\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{N} e_q\left(b\cdot(t-2x)\right)\right| = \left|\frac{e_q(b\cdot N+1)-1}{e_q(b)-e_q(-b)}\right| \le \frac{1}{\left|\sin\left(2\pi b/q\right)\right|}. \tag{21}$$

It follows that

$$\begin{aligned}
&\sum_{b=1}^{(q-1)/2}\left|\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{N}e_q\left(b\cdot(t-2x)\right)\right| \\
&\quad\le \frac{1}{\left|\sin\left(2\pi/q\right)\right|}+\sum_{b=2}^{(q-1)/2}\frac{1}{\left|\sin\left(2\pi b/q\right)\right|} \\
&\quad< \frac{1}{\left|\sin\left(2\pi/q\right)\right|}+2\int_{2}^{d((q-1)/4)t}\left|\csc\left(2\pi x/q\right)\right|d_x \\
&\quad< \frac{1}{\left|\sin\left(2\pi/q\right)\right|}+\frac{q}{\pi}\ln^{\cot(2\pi/q)}.
\end{aligned} \tag{22}$$

Then

$$\begin{aligned}
&\sum_{b=1}^{(q-1)/2}\left|\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{N}e_q\left(b\cdot(t-2x)\right)\right| \\
&\quad< q\left(\frac{1}{\pi}\ln^{\cot(\pi/q)}+\frac{1}{6}\right).
\end{aligned} \tag{23}$$

Thus we get the conclusion

$$\begin{aligned}
&\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{N-1}e_q\left(2b\cdot\left(\frac{v_t}{2}-x\right)\right)\right| \\
&\quad< q\left(\frac{1}{\pi}\ln^{\cot(\pi/q)}+\frac{1}{6}\right).
\end{aligned} \tag{24}$$

□

Lemma 6. *The binary strictly periodic sequence $\underline{a}$ corresponds to the 2-adic integer $\alpha = -p/q$, where the integer q is odd and primitive with p. The complement sequence $\underline{a}'$ of $\underline{a}$ has the 2-adic representation $\alpha' = -(q-p)/q$.*

Proof. From the description about the 2-adic integer, we have known that $-1 = 1 + 2 + 2^2 + 2^3 + \cdots$ and the sequence $\underline{1} = 1, 1, 1, \ldots$ have the 2-adic representation -1. The complement sequences $\underline{a}$ and $\underline{a}'$ satisfy $\alpha + \alpha' = -1$. Then, we have $-p/q + \alpha' = -1$; that is, $\alpha' = -(q-p)/q$. □

Theorem 7. *Let $\underline{a} = a_0, a_1, a_2, \ldots$ be a binary balanced period sequence with period T, $\underline{u} = \{u_t = p \cdot 2^{-t} \bmod q\}_{t=0}^{\infty}$ is the exponential of the sequence $\underline{a}$, the elements in $\underline{u} = \{u_t = p \cdot 2^{-t} \bmod q\}_{t=0}^{\infty}$ are symmetry with $(q-1)/2$, and the 2-adic integer of sequence $\underline{a}$ has the property*

$$\frac{\pi}{12}\cdot T < \log_2^q. \tag{25}$$

Proof. Let sequence $\underline{a} = a_0, a_1, a_2, \ldots$ with the period T and corresponding sequence $\underline{u} = \{u_t = p \cdot 2^{-t} \bmod q\}_{t=0}^{\infty}$ as the described sequence $\underline{v}$; we have

$$\begin{aligned}
&\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}e_q\left(2b\cdot\left(\frac{u_t}{2}-x\right)\right)\right| \\
&\quad=\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}e_q\left(2b\cdot\left(\frac{v_t}{2}-x\right)\right)\right|.
\end{aligned} \tag{26}$$

The sequences $\underline{v}$ have the following inequality:

$$\begin{aligned}
&\left|\sum_{t=0}^{T-1}\sum_{b=1}^{q-1}\sum_{x=0}^{(q-1)/2}e_q\left(b\cdot\left(\frac{v_t}{2}-x\right)\right)\right| \\
&\quad\le\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}e_q\left(2b\cdot\left(\frac{v_t}{2}-x\right)\right)\right| \\
&\qquad+\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}e_q\left((2b-1)\cdot\left(\frac{v_t}{2}-x\right)\right)\right|.
\end{aligned} \tag{27}$$

Then

$$\begin{aligned}
&e_q\left((2b-1)\left(\frac{v_t}{2}-x\right)\right) \\
&\quad=e_q\left(b\cdot(v_t-2x)\right)\cdot e_q\left(-1\cdot\left(\frac{v_t}{2}-x\right)\right).
\end{aligned} \tag{28}$$

We have

$$\begin{aligned}
&\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}e_q\left((2b-1)\cdot\left(\frac{v_t}{2}-x\right)\right)\right| \\
&\quad=\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}e_q\left(b\cdot(v_t-2x)\right)\right. \\
&\qquad\left.\cdot\,e_q\left(-1\cdot\left(\frac{v_t}{2}-x\right)\right)\right|.
\end{aligned} \tag{29}$$

That is,

$$\begin{aligned}
&\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}e_q\left(b\cdot(v_t-2x)\right)e_q\left(-1\cdot\left(\frac{v_t}{2}-x\right)\right)\right| \\
&\quad=\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}\left(e_q\left(b\cdot(v_t^e-2x)\right)\cdot\left(e_q\left(-\left(\frac{v_t^e}{2}-x\right)\right)-1\right)+e_q\left(b\cdot(v_t^e-2x)\right)\right)\right.
\end{aligned}$$

$$+\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}\left(e_q\left(b\cdot\left(v_t^o-2x\right)\right)\left(e_q\left(-\left(\frac{v_t^o}{2}-x\right)\right)+1\right)-e_q\left(b\cdot\left(v_t^o-2x\right)\right)\right)\Bigg|$$

$$=\left|\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{(q-1)/2}\frac{e_q\left(\left(\left(q+1\right)/2\right)\left(v_t^e-2x\right)-e_q\left(v_t^e-2x\right)\right)}{e_q\left(v_t^e-2x\right)-1}\left(e_q\left(v_t^e-2x\right)+1\right)+\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\cdot\left(v_t^e-2x\right)\right)\right.$$

$$\left.-\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\cdot\left(v_t^o-2x\right)\right)\right|=\left|\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}(-1+1)+\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\cdot\left(v_t^e-2x\right)\right)-\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\right.\right.$$

$$\left.\left.\cdot\left(v_t^o-2x\right)\right)\right|=\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\cdot\left(v_t^e-2x\right)\right)-\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\cdot\left(v_t^o-2x\right)\right)\right|, \tag{30}$$

v_t^e are defined as the even numbers in a period of $\underline{v}$, v_t^o are defined as the odd numbers in a period of $\underline{v}$, and then the inequality can be expressed as

$$\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}e_q\left(\left(2b-1\right)\cdot\left(\frac{v_t}{2}-x\right)\right)\right|$$

$$=\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}e_q\left(b\cdot\left(v_t-2x\right)\right)\right. \tag{31}$$

$$\left.-2\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\cdot\left(v_t^o-2x\right)\right)\right|.$$

We have

$$\left|\sum_{t=0}^{T-1}\sum_{b=1}^{q-1}\sum_{x=0}^{(q-1)/2}e_q\left(b\cdot\left(\frac{v_t}{2}-x\right)\right)\right|$$

$$\le 2\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}e_q\left(b\cdot\left(v_t-2x\right)\right)\right| \tag{32}$$

$$+2\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\cdot\left(v_t^o-2x\right)\right)\right|.$$

Since

$$\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\cdot\left(v_t^o-2x\right)\right)\right|$$

$$\le\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T/4}e_q\left(b\cdot\left(\left(2t+1\right)2x\right)\right)\right|, \tag{33}$$

then

$$\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\cdot\left(v_t^o-2x\right)\right)\right|$$

$$\le\sum_{b=1}^{(q-1)/2}\left|\frac{\sin\left(\left(T\right)b\pi/2q\right)}{\sin\left(2b\pi/q\right)\sin\left(b\pi/q\right)}\right|; \tag{34}$$

that is,

$$\left|\frac{\sin\left(\left(T\right)b\pi/2q\right)}{\sin\left(2b\pi/q\right)\sin\left(b\pi/q\right)}\right|\le\left(\cos^{T/4}\left(\frac{b\pi}{q}\right)\right.$$

$$\left.+\cos^{T/4-1}\left(\frac{b\pi}{q}\right)+\cdots+\cos\left(\frac{b\pi}{q}\right)\right)\cdot\frac{1}{\sin\left(b\pi/q\right)}. \tag{35}$$

So

$$\left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t}e_q\left(b\cdot\left(v_t^o-2x\right)\right)\right|$$

$$\le\sum_{b=1}^{(q-1)/2}\left(\cos^{T/4}\left(\frac{b\pi}{q}\right)+\cos^{T/4-1}\left(\frac{b\pi}{q}\right)+\cdots\right. \tag{36}$$

$$\left.+\cos\left(\frac{b\pi}{q}\right)\right)\cdot\frac{1}{\sin\left(b\pi/q\right)}\le\frac{q}{\pi}\cdot\left(\sum_{t=1}^{T/4}\frac{1}{t}\right)\ln^{\cot(\pi/q)}$$

$$\le\frac{q}{\pi}\cdot\ln^{T/4}\log_2^q.$$

As Lemma 3 $(T/4)(q-1)<(q/\pi)\cdot\log_2^q+(2q/\pi)\log_2^q$, we have $(\pi/12)\cdot T<\log_2^q$. □

If the connection integer q of balanced sequence $\underline{a}$ is large enough and satisfies $\ln^{\cot(\pi/q)}<\lfloor\log_2^q\rfloor$, we can have the following corollary.

Corollary 8. *Let $\underline{a}=a_0,a_1,a_2,\ldots$ be a period balanced sequence with period T, $\underline{u}=\{u_t=p\cdot 2^{-t}\bmod q\}_{t=0}^{\infty}$ is the exponential of the sequence $\underline{a}$, and the elements in $\underline{u}=\{u_t=p\cdot 2^{-t}\bmod q\}_{t=0}^{\infty}$ are symmetry with $(q-1)/2$, then the 2-adic complexity of sequence $\underline{a}$ has*

$$\frac{\pi}{12}\cdot T<\psi\left(\underline{a}\right). \tag{37}$$

The balanced binary sequence described in Theorem 7 is resistant to 2-adic attack, but the higher sequence requirements are difficult to achieve. In general, when considering the complexity, it cannot get its exponential representation.

How to get a relatively broad condition to reflect the relationship between 2-adic complexity and periodicity of binary balanced sequences is the problem we need to consider.

Lemma 9. *Let $\underline{a}$ be the balanced strictly periodic sequence of a binary sequence and correspond to the 2-adic integer p/q. Assume that the sequence $\underline{u} = \{u_t = p \cdot 2^{-t} \bmod q\}_{t=0}^{\infty}$ with $a_t = u_t \bmod 2$. Then, there must exist another sequence $\underline{v}$ over $Z/(q)$ satisfying*

$$\left|\sum_{b=0}^{(q-3)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1} e_q\left(\left(b+\frac{1}{2}\right)\cdot(v_t-2x)\right)\right| < \sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}\left|\sum_{b=0}^{(q-3)/2} e_q\left(\left(b+\frac{1}{2}\right)\cdot(u_t-2x)\right)\right|. \tag{38}$$

Proof. When $u_t = 2k_t + 1$ is an odd integer,

$$\left|\sum_{b=0}^{(q-3)/2} e_q\left(\left(b+\frac{1}{2}\right)\cdot(u_t-2x)\right)\right| = \left|\frac{1}{2\sin\left(\left((2k_t+1-2x)/q\right)\pi\right)}\right|. \tag{39}$$

When $u_t = 2k_t$ is an even integer,

$$\left|\sum_{b=0}^{(q-3)/2} e_q\left(\left(b+\frac{1}{2}\right)\cdot(u_t-2x)\right)\right| = \left|\frac{1}{2\cos\left(\left((2k_t-2x)/q\right)\pi\right)}\right|. \tag{40}$$

The sequence $\underline{u}$ in a period of length T has the same number of even integers and odd integers.

If $|2\sin(((2k_t+1-2x)/q)\pi)| = |2\cos(((2k_t-2x)/q)\pi)|$ and $|1/2\sin(((2k_t+1-2x)/q)\pi)| + |1/2\cos(((2k_t-2x)/q)\pi)|$ have a minimum value, then it can be arrived at the minimum value when u_t are bilateral symmetry with $(q-1)/2$. So

$$\left|\sum_{b=0}^{(q-3)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1} e_q\left(\left(b+\frac{1}{2}\right)\cdot(v_t-2x)\right)\right| < \sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}\left|\sum_{b=0}^{(q-3)/2} e_q\left(\left(b+\frac{1}{2}\right)\cdot(u_t-2x)\right)\right|, \tag{41}$$

where the sequence $\underline{v}$ in a period of length of T is also bilateral symmetry with $(q-1)/2$. □

Lemma 10. *Let the binary strictly periodic sequence $\underline{a} = a_0, a_1, a_2, \ldots$ be generated by an FCSR with connection integer q and the correspondence 2-adic integer is $-p/q$. Then, the sequence $\underline{u} = \{u_t = p \cdot 2^{-t} \bmod q\}_{t=0}^{\infty}$ satisfies*

$$\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}\left|\sum_{b=0}^{(q-3)/2} e_q\left(\left(b+\frac{1}{2}\right)\cdot(u_t-2x)\right)\right| < \frac{q}{\pi}T\left|\log_2^{p/(q-p)}\right|. \tag{42}$$

Proof. From the definition of $e_q(a)$, we have

$$\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}\left|\sum_{b=0}^{(q-3)/2} e_q\left(\left(b+\frac{1}{2}\right)\cdot(u_t-2x)\right)\right| = \sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}\left|\frac{e_q\left((q-3)/2+1+1/2\right)(u_t-2x)-e_q\left((1/2)(u_t-2x)\right)}{e_q(u_t-2x)-1}\right| = \sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}\left|\frac{1}{1\pm e_q(u_t/2-x)}\right|. \tag{43}$$

Then, when u_t is an even integer, $|1 + e_q(u_t/2 - x)| = 2|\cos((u_t - 2x)\pi/2q)|$, and when u_t is an odd integer, $|1 - e_q(u_t/2 - x)| = 2|\sin((u_t - 2x)\pi/2q)|$.

We have known that

$$\sum_{x=0}^{(q-1)/2}\left|\frac{1}{\sin\left((u_t-2x)\pi/2q\right)}\right| < \frac{q}{\pi}\left|\ln^{\tan(u_t-q)/2q} - \ln^{\tan(u_t)/2q}\right|. \tag{44}$$

As q is large integer, we get

$$\sum_{x=0}^{(q-1)/2}\left|\frac{1}{1\pm e_q(u_t/2-x)}\right| < \frac{q}{\pi}\left|\log_2^{u_t} - \log_2^{q-u_t}\right|. \tag{45}$$

Then we have

$$\sum_{t=0}^{T-1}\sum_{x=0}^{(q-1)/2}\left|\frac{1}{1\pm e_q(u_t/2-x)}\right| < \frac{q}{\pi}\sum_{t=0}^{T-1}\left|\log_2^{u_t/(q-u_t)}\right|. \tag{46}$$

□

From Lemma 3, $a_t = u_t \bmod 2$ is the complement with the binary sequence $\{a_t' = (q-u_t) \bmod 2\}_{t=0}^{\infty}$. Since the one-to-one correspondence between the binary sequence and 2-adic integer, we have $a_t' = (p-u_t) \bmod 2 = ((q-p)\cdot 2^{-t} \bmod q) \bmod 2$.

We have $u_t = p\cdot 2^{-t} - n_t q$, $q - u_t = (q-p)\cdot 2^{-t} - n_t' q$, and $n_t' \approx ((q-p)/p)n_t$. Then

Table 1: The security of binary sequences.

	L-sequence	m-sequence	AES balance
Period	2^{e-1} bit	$2^e - 1$ bit	T
Occurrence	$A_0 = A_1$	$A_0 = A_1 - 1$	$A_0 = A_1$
$E(C(\tau))$	0	0	—
$V(C(\tau))$	$O(2^{e-1}/\ln^{2^{e+1}})$	$O(2^e - 1/\ln^{4(2^e-1)})$	—
2-adic complexity	$2^{e-2} \succ \varphi$	$\varphi \succ 2^{e-1} - 1$ (Test)	$\varphi \succ T/2 - 1$ (Test)

Table 2: Hardware performance comparison.

	Critical path (ns)	Frequency (MHz)	Throughout (Mbps)	Total logic elements	Total registers	Initialization cycles
Trivium	4.623	216.31	216.31	650	321	1152
AES	5.232	199.43	216.31	632	344	1146
FFCSR-2	5.452	183.42	1467.36	622	420	182
FCSR-SS	5.950	168.07	1344.56	771	506	182

$$\begin{aligned}
&\sum_{t=0}^{T-1} \left|\log_2^{u_t/(q-u_t)}\right| \\
&\quad = T\left|\log_2^{p/(q-p)}\right| \\
&\qquad + \left|\log_2^{\prod_{t=0}^{T-1}((p(q-p)2^{-t}-(q-p)qn_t)/(p(q-p)2^{-t}-pqn'_t))}\right| \\
&\quad \approx T\left|\log_2^{p/(q-p)}\right|.
\end{aligned} \tag{47}$$

So we have

$$\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}\left|\frac{1}{1 \pm e_q(u_t/2 - x)}\right| < \frac{q}{\pi}T\left|\log_2^{p/(q-p)}\right|. \tag{48}$$

Thus, we get the conclusion

$$\begin{aligned}
&\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1}\left|\sum_{b=0}^{(q-3)/2} e_q\left(\left(b + \frac{1}{2}\right)\cdot(u_t - 2x)\right)\right| \\
&\quad < \frac{q}{\pi}T\left|\log_2^{p/(q-p)}\right|.
\end{aligned} \tag{49}$$

Theorem 11. *Let $\underline{a} = a_0, a_1, a_2, \ldots$ be a balanced strictly periodic sequence; if the connection integer α satisfies $1/3 \le |\alpha| \le 3/5$, then its correspondence 2-adic integer, its period T, and the 2-adic complexity $\psi(\underline{a})$ satisfy*

$$T\left(\frac{\pi}{4} - \left|\log_2^{\alpha/(1-\alpha)}\right|\right) < \psi(\underline{a}) + \frac{1}{6}. \tag{50}$$

Proof. From Lemmas 3 and 9, we have known that

$$\begin{aligned}
\frac{T(q-1)}{4} &= \left|\sum_{t=0}^{T-1}\sum_{b=1}^{q-1}\sum_{x=0}^{(q-1)/2} e_q\left(b\cdot\left(\frac{v_t}{2} - x\right)\right)\right| \\
&\ge \left|\sum_{b=0}^{(q-3)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1} e_q\left(\left(b + \frac{1}{2}\right)\cdot(v_t - 2x)\right)\right| \\
&\quad + \left|\sum_{b=1}^{(q-1)/2}\sum_{x=0}^{(q-1)/2}\sum_{t=0}^{T-1} e_q(b\cdot(v_t - 2x))\right|.
\end{aligned} \tag{51}$$

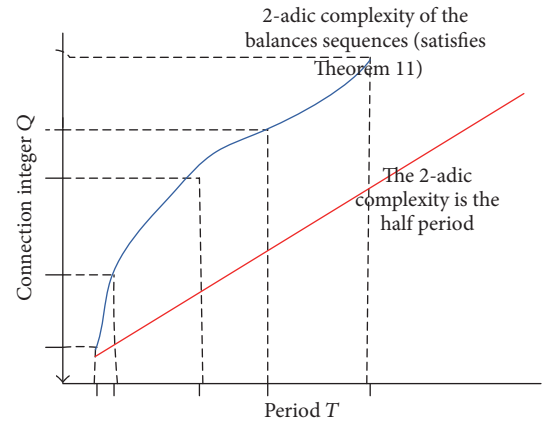


Figure 2: 2-adic complexity of binary sequences.

From Lemmas 5 and 10, we have

$$\frac{T(q-1)}{4} < q\left(\frac{1}{\pi}\log_2^q + \frac{1}{6}\right) + \frac{q}{\pi}T\left|\log_2^{p/(q-p)}\right|. \tag{52}$$

As $\alpha = -p/q$, then we get the conclusion

$$T\left(\frac{\pi}{4} - \left|\log_2^{\alpha/(1-\alpha)}\right|\right) < \psi(\underline{a}) + \frac{1}{6}. \tag{53}$$

Theorem 11 needs the connection integer α to satisfy $1/3 \le |\alpha| \le 3/5$. □

Note that the sequences of which the second half of one period is the bitwise complement of the first half are also the balanced sequences but their 2-adic complexities do not correspond to this result. The 2-adic complexity of these sequences with the analysis in this article inconformity is due to the bit proportion and distribution in a period of sequences, and it is well known that their 2-adic complexity is smaller than their half period because of the bitwise complement. However, through the experiment (Figure 2), their 2-adic complexity (except the long sequences) is approximated with their half period but we have not got faithful and accurate proving to analyze this result.

4. Conclusion

The vigorous development of fog calculation is increasing the security requirements on it. Stream ciphers are undoubtedly the most suitable (see Table 1) among the nodes in the situation of lightweight security encryption, and the security of the stream cipher directly affects the communication security of the fog computing nodes (see Table 2). In this correspondence, lower bounds of the 2-adic complexity of binary periodic sequences are presented, and they are influenced by the length of encryption sequences in fog computing. However, the tighter lower bounds are not determined, so better results are desirable.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] P. Mahajan and A. Sachdeva, "A study of encryption algorithms, de sand for security," *Global Journal of Computer Science and Technology*, vol. 13, no. 15, pp. 15–22, 2013.

[2] M. Goresky and A. Klapper, "Arithmetic crosscorrelations of feedback with carry shift register sequences," *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, vol. 43, no. 4, pp. 1342–1345, 1997.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 2734219, 8 pages
https://doi.org/10.1155/2018/2734219

Research Article

Immune Scheduling Network Based Method for Task Scheduling in Decentralized Fog Computing

Yabin Wang, Chenghao Guo, and Jin Yu

Science and Technology on Information Systems Engineering Laboratory, Nanjing 21007, China

Correspondence should be addressed to Yabin Wang; 517129269@qq.com

Received 2 May 2018; Accepted 15 July 2018; Published 2 September 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Yabin 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.

Fog computing has changed the distributed computing rapidly by including the smart devices widely distributed at the network edges. It is able to provide less latency and is more capable of decreasing traffic jam in the network. However, it will bring more difficulties for resource managing and task scheduling especially in a decentralized ad hoc network. In this paper, we propose a method that takes advantages of the immune mechanism to schedule tasks in a decentralized way for fog computing. By using forward propagation and backward propagation in the ad hoc network, the power of distributed schedulers is used to generate the optimized scheduler strategies to deal with computing nodes overloaded and achieve the optimal task finishing time reducing. The experiment results show that our approach can beat similar methods.

1. Introduction

With the number of smart wearable devices increasing and in-vehicle systems continuing improving, more and more users depend on mobile computing to deal with affairs in everyday life. Many mobile applications rely on offloading remote resources to complete their tasks. The remote resources mainly consist of large-scale computing clusters within a data center. These cloud computing clusters are also used for storing data.

User applications access the cloud center by using access points to exchange data between the data center and the user applications by using the core network. The computing capability of the edge network makes the access points being able to provide computing and storage services [1]. Edge computing nodes communicate with cloudlets. Cloudlets can also exchange information including control/management data (e.g., computing nodes and applications state) with each other to manage data and processes. Data storage and analysis is one hop away from the place where data are produced and consumed. This infrastructure can benefit different types of user applications. These application types are as follows:

(i) Applications that need to be run in low latency, e.g., surveillance applications and tactical applications will benefit from being only one hop away to the cloudlet.

(ii) If applications relying on the cloud to analyze large amount of data deploy themselves in a fog and the analyzing procedure is performed in the cloudlet one hop away, shorter time of delay and response time can be achieved. Less data traffic can also be achieved this way.

(iii) Large amount of unprocessed raw data collected by many edge devices need not to be stored in the remote cloud for long term. Data can be filtered, analyzed, and aggregated to generate knowledge and less amount of data need to be stored. The knowledge can be also used for other edge devices. The fog can reduce the network traffic from edge to the remote cloud in the above both cases.

Fog computing is able to provide less latency and is more capable in decreasing traffic jam in the network. But this will bring more difficulties for resource managing and task scheduling. More new challenges need to be overcome. For example, if a fog node is overloaded, on which node and when to respond to corresponding requests processed by the overloaded nodes should be determined. Generated strategies must consider the mobility of data and computing nodes [2]. This is even more difficult when the decisions are made in a fully decentralized ad hoc network in fog

computing [3]. However, most of the existing scheduling methods depend on a centralized controller to perform task scheduling which is not suitable for the ad hoc scenario [4]. In this paper, we first introduced the immune mechanism which has the characteristics of self-organization, cooperation, and robustness to reschedule tasks to deal with the problem of overload in fog computing.

The rest of the paper is organized as follows. The related work to this paper is introduced in Section 2. Proposed approach is introduced in Section 3. Experiments and the results are discussed in Section 4 and followed by the conclusion in Section 5.

2. Related Work

Previous researches about scheduling in a distributed scenario can be classified into 2 classes as follows [5].

(i) Deterministic methods: these methods use some characteristics of a problem to solve it.

(ii) Nondeterministic methods: random search methods techniques are used.

The deterministic methods include list-scheduling algorithms and clustering algorithms [6–8]. List-scheduling algorithm first creates one priority list of tasks. Then one task is selected from the list and assigned to one node based on some heuristics. List-scheduling algorithms can be further classified into dynamic and static list-scheduling according to whether the initial priority list changes during the executing of the algorithm. The priority list will not be changed in the executing of static list scheduling. The drawbacks of list-scheduling are that they will not achieve consistent results for different problems. The reason is that some specific heuristics are not persistent to problems with different properties. The most popular list-scheduling methods include Modified Critical Path (MCP) [8], Mapping Heuristic (MH) [9], and Dynamic Critical Path (DCP) [10]. Clustering algorithms group tasks with high data dependency together. The group is called a cluster task. The group of task will be assigned to computing nodes that are adjacent to each other. By this, the makespan can be reduced. Clustering methods first assume that unbounded number of computing nodes can be used and then the number decreases to the realistic number [11]. The most popular cluster methods are Dominant Sequence Clustering (DSC) [12].

The nondeterministic includes population-based algorithms such as PMC_GA [13]. The method takes longer time to find scheduling solutions compared with the above methods. The advantage of PMC_GA is to be suitable for wider range of problems compared with the above-mentioned methods.

3. Proposed Approach

Our approach effectively uses the ability of geographically distributed computing nodes to make schedule decisions. The scheduler is distributed in each computing node. Each scheduler will generate scheduling polices. Each scheduler will corporate with each other to generate better scheduling policies.

The scheduler in the computing node consists of an immune scheduling network. The network uses immune mechanism to balance the scalability and scheduling quality. The network is a distributed cooperative scheduling framework, and the local scheduler in each computing node makes scheduling decisions independently. The local scheduler collaborates with one another and synchronizes information. Tasks are allocated to local schedulers, and local schedulers communicate with each other regularly, exchanging information of each computing node's network and computing load. The framework avoids conflict scheduling decisions caused by fully decentralized scheduling and avoids centralized bottleneck and single point failure of schedulers.

First we will describe the concepts in the immune mechanism and the immune scheduling network and then describe the detailed procedure of our approach.

3.1. Immune Mechanism. The immune system is one of the most important systems in our body. It is able to protect our body from the pathogens and abnormal behaviour of some elements within our body [4]. One of the most important constituents of the immune system is lymphocytes. Lymphocytes are also called white blood cells. These cells are generated in the bone marrow.

The natural immune system protects our body from the harmful foreign elements and abnormal behaviour of our body. It is one of the most important organs in our body. One of the most important constituents of our immune system is blood cells. Blood cells are created in the bone marrow. They exist in the blood and lymph system and perform immunological functions. The main population of lymphocytes consists of B and T cells [14].

B cells and T cells have receptors. Receptors of B cells will recognize pathogens and then the B cell will clone and differentiate itself to be proliferated. Among the cloned cells, some are antibodies. Antibodies are responsible for combating with antigens. Some of the cloned cells are responsible for memorizing the type of attacks so that it will get a faster reaction when facing the same attack for the second time. The affinity between each antibody and each antigen will be calculated. The affinity will be improved by mutating antibodies. Better antibodies for combating with specific antigens can be selected by choosing antibodies with higher affinities. The essential parts of immune mechanism are the cloning, mutation, and selection.

The immune network theory [15] is another important concept. The core idea in immune network theory is that antibodies will stimulate and recognize each other. An antibody will be stimulated and cloned when recognizing another antibody. The recognized antibody will be suppressed. This procedure will avoid producing inordinate antibodies and make the whole immune system stable. The immune network, clone, and selection lead to diversity of population and improved searching for optimized solutions.

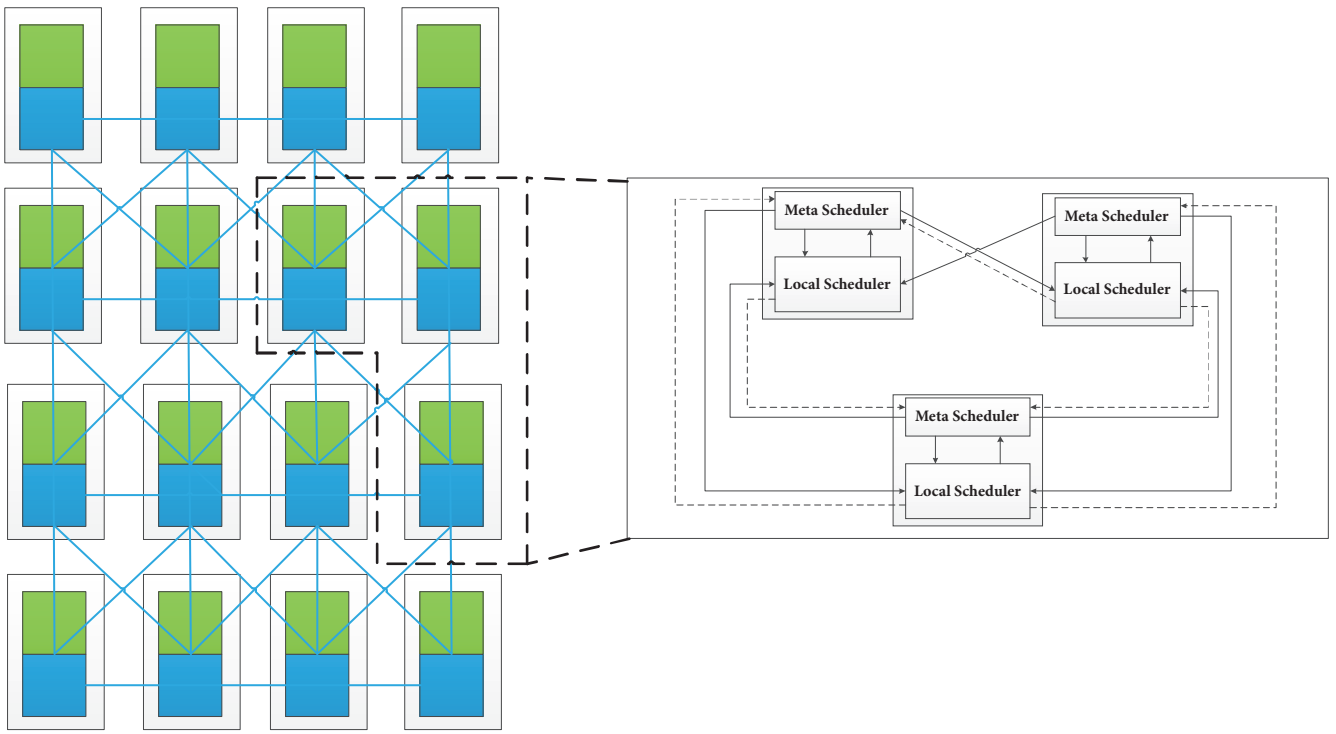


Figure 1: The architecture of immune scheduling network.

3.2. Immune Scheduling Network. The immune scheduling network as shown in Figure 1 consists of meta-schedulers and local schedulers. The network uses the immune network to implement the scheduling. The immune network is based on the interaction between antibodies and the communication between different kinds of immune cells. The immune network regulates the mutual promotion and inhibition between the antibodies to match the antigen and the antibody.

The main function of B cells is to produce antibodies. Immune networks rely mainly on antibodies to attack invasive antigens to protect organisms. The B cell is equivalent to the local scheduler in the immune scheduling network. The local scheduler exists in every computing node, which produces the equivalent of the scheduling strategy, and the antigen is equivalent to the problem that the scheduling policy needs to solve such as minimizing the execution time and minimizing the network traffic. The antibody is a scheduling strategy relative to the problem to be solved.

Meta-scheduler collects the tasks of the surrounding nodes to occupy the resource information (information including CPU, memory, bandwidth, hard disk storage, etc.), and broadcasts the information to adjacent nodes, and the adjacent nodes learn continuously to obtain more and more nonadjacent computing nodes.

Local scheduler learns the information of meta-scheduler and generates antibody according to immune mechanism (scheduling strategy), and the scheduling strategy of different nodes will choose the best solution.

3.3. Detailed Procedure. When some computing nodes suffer from overloading or damaging, two steps are implemented.

3.3.1. Nonspecific Immunization Stage. This stage is an emergency treatment stage. The nodes around the overloaded nodes quickly evaluate their own resource status to determine whether their resource can execute the tasks. If the evaluation result is yes, the tasks will be assigned to a qualified node randomly by the task requester fast. After this stage the nodes around overloaded node may also suffer from overload. Then the second procedure needs to be performed to balance the amount of tasks of the nodes in the network.

3.3.2. The Specific Immunization Phase. The schedulers working as immune cells need to identify the characteristics of the scheduled tasks working as antigen. The scheduler needs to generate a targeted scheduling strategy based on the type of task in the overloaded node. Different types of task correspond to different optimization target functions (the affinity function of antibodies and antigens). The target function includes minimizing the execution time.

After determining the objective function, the information of overloaded node needs to be broadcast nodes by nodes to inform the nodes not adjacent to the overloaded node, as nodes can only sense the nodes around them. The informed nodes may activate the local scheduler to generate scheduler strategies. Schedulers need to use the immune learning methods to generate some antibodies according to the resources of adjacent nodes. The strategies generated by different schedulers will be compared and the best one will be selected as the final scheduling strategy. Immune learning is complex if all the schedulers in the network all perform immune learning; it will cost too much time. In our approach, we will use path selection to select path that the overloaded

information being broadcast so that only a subset of scheduler needs to be activated.

This kind of distributed scheduling will cause a few nodes to be dispatched frequently, causing overheating [16]. Immune suppression mechanism can avoid this situation.

3.4. Forward Propagation and Backward Propagation. The propagation process is composed of forward propagation process and back propagation process. In the forward propagation process, the activation signal is transported nodes by nodes. If the scheduler is activated, the scheduler will use the immune mechanism to generate antibodies. The key to forward propagation is to find a scheduler activating path that can generate the optimized antibodies. In the backward propagation process, all the antibodies generated by activated schedulers will be selected, cloned, mutated, and compared to generate the final antibodies.

In order to find the optimized scheduler activated path, our approach uses the objective function to evaluate the local and surrounding resources and assign a score S_1 to it. S_1 will be distributed to the surrounding nodes. The surrounding nodes will use the same way for evaluation to get S_2. $|S_1 - S_2|$ will be used as input to the local activation function. The activation function determines whether the scheduler is activated or not.

The aim of the activation function is to evaluate whether the node and its surrounding nodes are eligible to be added to a list of qualified antibodies. The idea of the activation function is that if S_1 is high, scheduler with higher scores than S_1 should be activated. In order to get more appropriate schedulers, schedulers with approximate but less score are still activated. If S_1 is low, the situation is the opposite. In order to deal with these two situations, the activation function must be designed carefully.

In computational network, the activation function is the heart of the neural network [17]. Each node in the neural network receives the input value and passes the input value to the next layer. The input node directly transfers the value of the input attribute to the next layer (hidden layer or output layer). In the neural network, there is a functional relationship between the input and output of the hidden layer and the output layer node, which is called the activation function.

The output of each node in the neural network is based on the definition of the relevant activation function. The activation function is sometimes called the processing unit function or the compression function. It is used to input a set of inputs on the input arc. The activation function is also called the ignition rule, which makes it connect with the work of the human brain. When a neuron's input is large enough, it will ignite, that is, to send electrical signals from its axon (output connection). Similarly, in artificial neural networks, the output rule is generated when the input exceeds a certain standard, which is the idea of ignition rules. When dealing with only two value outputs, the output is either 0 or 1 depending on whether the neuron should be ignited [18]. In our research, the activation function will determine whether a node will create antibodies. Bekir [19] studied different action activation functions and found the best function Tanh which is used in our approach. The function is represented as

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{1}$$

Figure 2 shows an example of finding a scheduling activating path in forward propagation. There are 11 computing nodes composing an ad hoc fog network. Each box represents a node. The green rectangle inside a box represents a task running in the node. A blue box indicates that the computing node is activated. Now node A is overloaded. The surrounding nodes B, C, D, and J cannot process all the tasks in A, so the specific immunization phase has to start. D and J get higher scores after evaluation using objective function. D and J are activated (a). The objective function will be described in detail in the following section. E gets higher scores than D so E is activated (b). G and F get lower scores than E. I and K get similar high scores than J. G and F are not activated and I and K are activated (c). L has the same high scores as I does, so L is activated.

In the forward propagation procedure, each activated scheduler uses the immune mechanism described in Section 3.3 to generate the best antibodies. The generated antibodies will be transported back to the overloaded computing node in backward propagation. During the transportation the generated antibodies will be further compared and the best ones remained. Figure 3 shows an example of backward propagation. When antibodies are transformed from I to J, the generated antibodies will be compared using the objective function. The antibodies with the highest scores will remain.

3.5. Objective Function. Evaluating an antibody involves evaluating the computing capability of each node, bandwidth of communication between pairs of nodes, and other properties, e.g., communication methods of protocols between processors. Some researches [4] considered the environment homogenous. In the real fog environment, the nodes may have different types of processors and communication bandwidth. In our research, heterogeneous environment, processors, and message transferring ways are assumed. We also assumed that the nodes will suffer from failure. The notation is described as follows. P indicates processors. p_i indicates the ith processor. P_c indicates the number of processors.

The aim of scheduling tasks is to minimize the makespan. The aim can be formulated as

$$f : t_s \longrightarrow N_c \tag{2}$$

The function creates a map between a task and a computing node. The notations are denoted by

$$T_A = \left\{ t_j \in T \mid f\left(t_j\right) = n_i \right\} \tag{3}$$

T_A indicates the whole task set that has been assigned to the computing node n_i. The completion time of the whole task in node i is presented as

$$c_{i=} \left\{ rft\left(t_j\right) \right\} \quad where\ t_j \in T_A \tag{4}$$

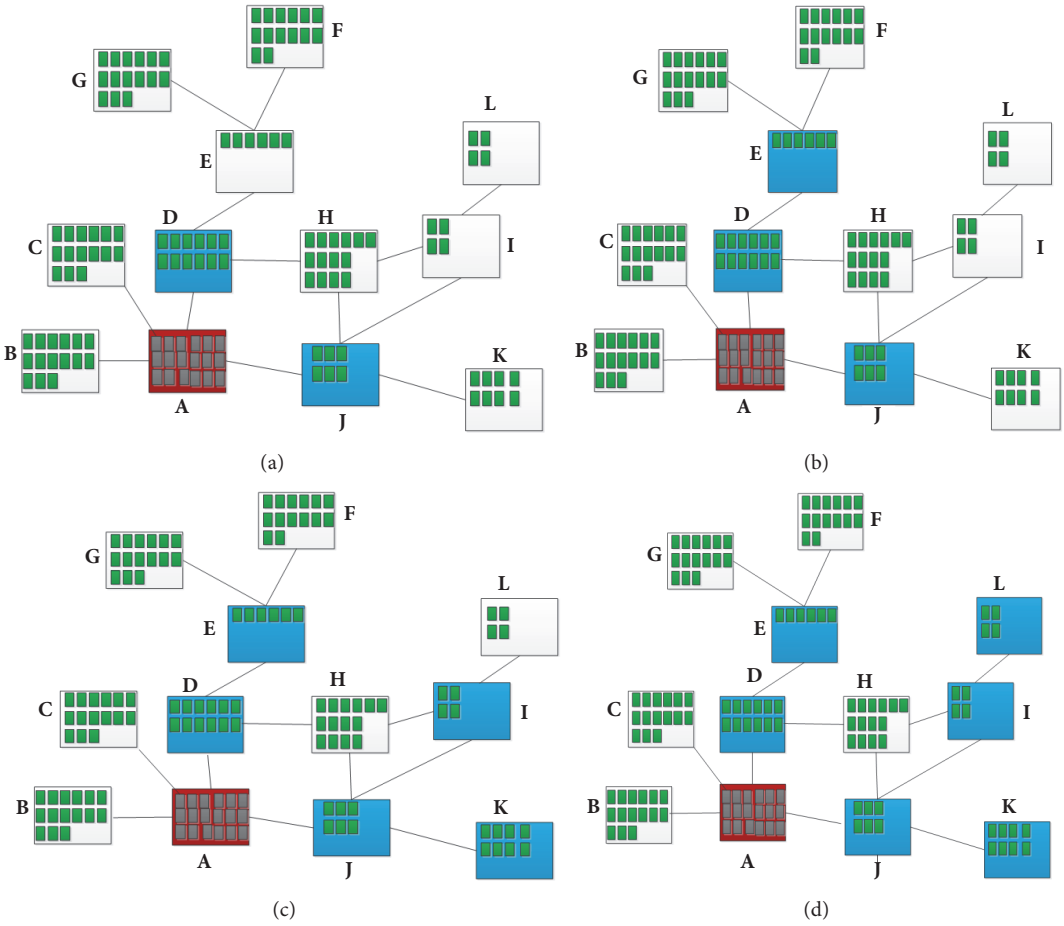


Figure 2: An example of selecting path in forward propagation.

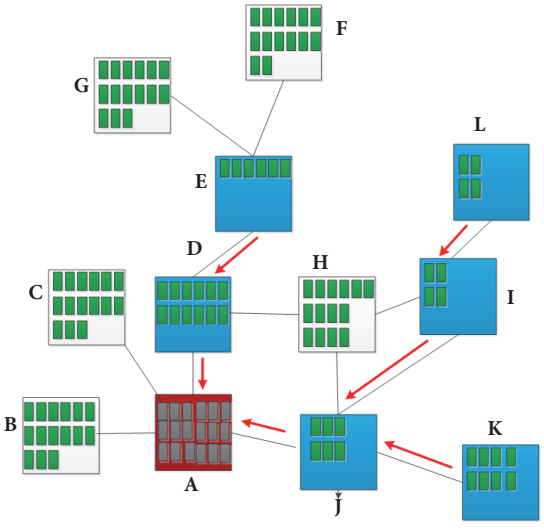


Figure 3: An example of backward propagation.

t_j is the last executed task that is assigned to the computing node i. $rft(t_j)$ is the realistic finish time of task t_j.

$$rft\left(t_j\right) = \left\{rst\left(t_j\right) + et\left(t_j, f\left(t_j\right)\right)\right\} \tag{5}$$

$et(t_j, n_i)$ is the time of task t_j executing on node i. rst is the realistic starting time of a task. A task is ready if all its predecessor tasks have been executed and the data the task required have been ready. At this moment, the task started. In this paper, the time is called the first start time (fst). fst is formally represented as

$$fst\left(t_i\right) = \max\left\{rft\left(t_j\right) + w_{ij} * C\cos\left(f\left(t_i\right), f\left(t_j\right)\right) \mid t_j \in pre\left(t_i\right)\right\} \tag{6}$$

In this equation, C cos is represented as

$$C\cos\left(n_i, n_j\right) = \begin{cases} 0, & if\ i = j \\ r_{ij}, & if\ i \neq j \end{cases} \tag{7}$$

rij is the communication cost between computing nodes i and j. When calculating the C cos, the distance between f(ti) and f(tj) is also considered.

rst(tj) indicates the time the task is ready to be started and there is no other task with higher priority to be processed. rst(tj) is represented as

$$rst\left(t_j\right) = \left\{fst\left(t_j\right) + waittime\left(t_j\right)\right\} \tag{8}$$

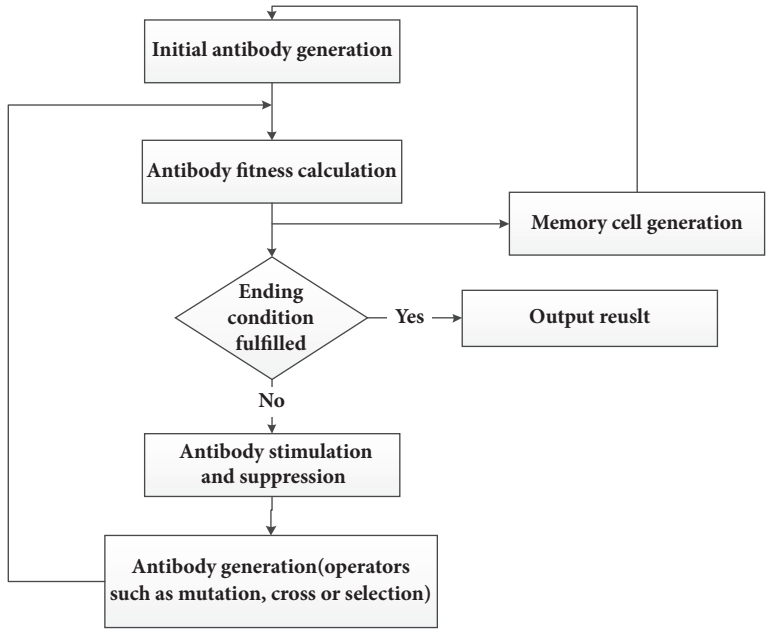


Figure 4: The detailed procedure.

Waittime(Tj) is the time that task Tj which is ready waits for task with higher priority that is ready to be finished. The makespan is represented as

$$C\max = \max\left(c_i\right) \quad i = 0, 1, \ldots n_i \tag{9}$$

The objective function of our approach is

$$obj = \frac{cm - \min\left(cm\right)}{\max\left(cm\right) - \min\left(cm\right)} \tag{10}$$

In this function, cm = C max. min (cm) and max (cm) are the minimum and maximum value of all the solutions.

It is important to note that the time cost of each the task running on each node and the time cost of communication between nodes need to be tested and recorded beforehand. The time cost will be input of our objective function.

The algorithm started with a set of population generated at random. The population will be further improved in several iterations. Each antibody represents a candidate for the solutions of the scheduling. The algorithm encodes the antibody as a string consisting of integers. The indexes of the integers in the string indicate the index number of tasks. The value of an integer indicates the computing node that the corresponding task will be assigned to.

3.6. Antibody Generating. The basic idea of generating antibodies is as follows:

(1) Using the basic operation of genetic algorithm and the way of population evolution to generate the optimal antibody

(2) Based on the mechanism of lymphocyte interaction in the immune network to regulate antibody affinity, to suppress similar antibodies

(3) Using immune memory to accelerate the convergence of the algorithm

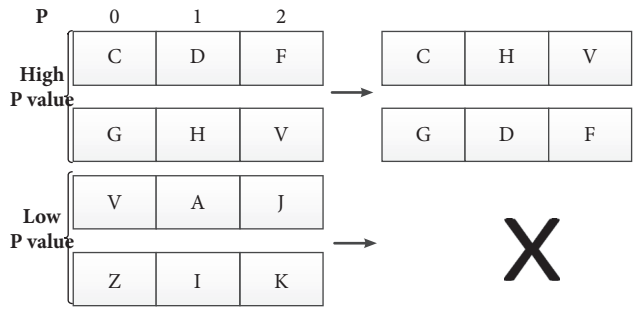


Figure 5: Antibody stimulation and suppression.

The detailed procedure is shown in Figure 4.

Initial antibody generation: it generates a random population. The population of antibody determines the order of executing the tasks. During the execution, the makespan can be calculated. After this is the clonal selection procedure. The clonal selection results in the search in the solution space and gets the best solution from the whole space. The key is the tradeoff between exploration and exploitation [4]. The clonal selection procedure includes antibody fitness calculation, antibody stimulation and suppression, and antibody generation.

As shown in Figure 5, an antibody is represented as an array where the index of each element represents a task index. The elements in the array represent the node to which the corresponding task needs to be assigned to. In Figure 5, numbers 0, 1, and 2 tasks are dispatched to C, D, and F nodes, respectively.

Antibody fitness calculation: it calculates fitness of each antibody using the objective function. The antibody with the highest fitness is retained. After several iterations of stimulation and inhibition, the best antibodies are retained.

The immune-remove phase follows the clonal selection. The elitism is provided by the immune phase. To make the next population more diverse, the elitism and random population insertion is performed. The algorithm will be iterated for K times. K is set in the begging. The output of the algorithm is the best antibody of all the populations and the order of executing the task.

Memory cell generation: after the fitness of each antibody calculated, the maximum fitness of the total population will be calculated. If the maximum fitness value is greater than the maximum fitness value provided by the immune network, the corresponding individual is added to the antibody memory table as a new good antibody. The maximum fitness value will be also added to the antibody adaptation table; otherwise, the immune memory process started. According to the historical antibody table, the immune memory uses the chaos multiplication to find the individual with highest fitness value to work as the better immune antibody [20].

Antibody stimulation and suppression: many highly frequent mutations will destroy the affinity between antibodies and antigens. It is necessary to selectively increase the number of high affinity antibodies to solve this problem. The replication operation of the next generation is carried out according to the replication probability calculated in the antibody production operator.

In a population, the expected reproductive probability of each individual is determined by both the affinity between antibody and antigen and the antibody concentration.

$$P = \alpha \frac{A_v}{\sum A_v} + (1 - \alpha) \frac{C_v}{\sum C_v} \quad (11)$$

α is a constant. Av indicates the fitness of an individual. Cv indicates the concentration of the individual. The higher the Av and Cv are, the greater the expected reproductive rate will be. A larger value of α encourages the generation of individuals with high fitness and suppresses the individuals with high concentration as shown in Figure 5.

Antibody generation: different operators have different global search capability [21]. Qiuzhen Lin [22] proposed a new operator which gained a wider diversity by including a suitable parent selection strategy. This operator is used to make the scheduler strategies more diverse to make the more optimized antibody to be generated in the earlier iterations. In order to generate more optimized antibodies, each antibody has to be cloned for Cc times. Cc is a positive integer value and works as one of the inputs of our approach. Each cloned antibody will be mutated. Random elements in the antibody will be selected and on which the operator described above needs to be performed. After cloning the fitness of each clone is calculated. The clone with the highest fitness is added to the antibody population.

4. Experiment and Result

In our experiment, proposed approach is compared with well-known scheduling algorithms including MCP [8], DSC [9], MD [8], DCP [10], and PMC_GA. A popular parameters setting method used in [4, 11] is used in our experiment. The parameter setting is described in Table 1.

Table 1: Parameter setting.

Parameter	value
Number of antibodies	400
Number of clones	50
Selection rate	0.25

Table 2: Experiment results for application with 8 tasks.

Approaches	MCP	DSC	MD	DCP	PMC_GA	Proposed
Number of nodes	8	8	8	8	8	8
Finish time (s)	26	23	29	27	20	13
Number of iterations	52	47	54	50	39	28

Table 3: Experiment results of application with 18 tasks.

Approaches	MCP	DSC	MD	DCP	PMC_GA	Proposed
Number of nodes	8	8	8	8	8	8
Finish time (s)	530	460	461	443	443	320
Number of iterations	120	106	105	100	101	88

Table 4: Experiment results of application with 50 tasks.

Approaches	PMC_GA	Proposed
Number of nodes	8	8
Finish time (s)	719	692

Two kinds of Gaussian Elimination application [11] are used for scheduling, one containing 9 tasks and the other containing 18 tasks. In order to compare proposed approach with other approaches a simulation experiment is conducted. In the experiments random node is selected to be made overloaded. Then the scheduling procedure is triggered. The finish times of all the tasks are recorded and compared. The results for application with 8 tasks are shown in Table 2. It is evident that our approach can achieve smaller finish time including the time of scheduling tasks.

Table 3 shows the results of application with 18 tasks. It is obvious that our approach can achieve lower finish time and smaller number of iterations.

In order to evaluate the performance of our approach on more complex application with 50 tasks, different conditions are considered based on some popular used task graph database [23]. The communication cost of tasks is added to make the simulation more realistic. The best approach of all the traditional ones is compared with our approach. The termination condition is set according to the study of Vahid Majid Nezhad et al. [22]. The termination condition is that the fitness remains unchanged in 10 iterations. The results are shown in Table 4. It is evident that our approach is still better than PMC_GA in more complex applications.

5. Conclusions

In this paper, a decentralized immune scheduler network based method is proposed to assign tasks in an ad hoc network for fog computing. This method takes advantage of

the immune mechanism to scheduler tasks in a decentralized way in an ad hoc network for fog computing. In the proposed method, by using propagation and backward propagation in the ad hoc network, the power of distributed schedulers is used to generate the optimized scheduler strategies to deal with computing node overloaded and achieve the optimal task finishing time decrease. The experiment results show that our approach can beat similar methods.

Data Availability

The experimental data used to support the findings of this study have not been made available because the data are related to our lab's commercial secrets.

Conflicts of Interest

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

Acknowledgments

The study is funded by the Pre-Research on the Equipment of the Army Information System Project [no. 3511080401].

References

[1] L. F. Bittencourt, J. Diaz-Montes, R. Buyya, O. F. Rana, and M. Parashar, "Mobility-Aware Application Scheduling in Fog Computing," *IEEE Cloud Computing*, vol. 4, no. 2, pp. 26–35, 2017.

[2] C. Mouradian et al., "A Comprehensive Survey on Fog Computing: State-of-the-art and Research Challenges," *IEEE Communications Surveys & Tutorials*, vol. 99, p. 1, 2017.

[3] K. A. Hummel and G. Jelleschitz, "A robust decentralized job scheduling approach for mobile peers in ad-hoc grids," in *Proceedings of the 7th IEEE International Symposium on Cluster Computing and the Grid, CCGrid 2007*, pp. 461–468, May 2007.

[4] M. Sanei and N. M. Charkari, "Hybrid heuristic-based artificial immune system for task scheduling," *International Journal of Distributed & Parallel Systems*, vol. 2, no. 6, 2011.

[5] Y.-K. Kwok and I. Ahmad, "Static scheduling algorithms for allocating directed task graphs to multiprocessors," *ACM Computing Surveys*, vol. 31, no. 4, pp. 406–471, 1999.

[6] B. Kruatrachue and T. Lewis, "Grain size determination for parallel processing," *IEEE Software*, vol. 5, no. 1, pp. 23–32, 1988.

[7] G. C. Sih and E. A. Lee, "Compile-time scheduling heuristic for interconnection-constrained heterogeneous processor architectures," *IEEE Transactions on Parallel and Distributed Systems*, vol. 4, no. 2, pp. 175–187, 1993.

[8] M. Wu and D. D. Gajski, "Hypertool: a programming aid for message-passing systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 1, no. 3, pp. 330–343, 1990.

[9] H. El-Rewini and T. G. Lewis, "Scheduling parallel program tasks onto arbitrary target machines," *Journal of Parallel and Distributed Computing*, vol. 9, no. 2, pp. 138–153, 1990.

[10] Y. Kwok and I. Ahmad, "Dynamic critical-path scheduling: an effective technique for allocating task graphs to multiprocessors," *IEEE Transactions on Parallel and Distributed Systems*, vol. 7, no. 5, pp. 506–521, 1996.

[11] R. Hwang, M. Gen, and H. Katayamaa, "A comparison of multiprocessor task scheduling algorithms with communication costs," *Computers & Operations Research*, vol. 35, no. 3, pp. 976–993, 2008.

[12] T. Yang and A. Gerasoulis, "DSC: scheduling parallel tasks on an unbounded number of processors," *IEEE Transactions on Parallel and Distributed Systems*, vol. 5, no. 9, pp. 951–967, 1994.

[13] Y. C. Lee and A. Y. Zomaya, "An Artificial Immune System for Heterogeneous Multiprocessor Scheduling with Task Duplication," in *Proceedings of the 2007 IEEE International Parallel and Distributed Processing Symposium*, pp. 1–8, Long Beach, CA, USA, March 2007.

[14] D. Dasgupta and F. Nino, "Immunological computation : theory and applications," in *Longman*, p. 140, 2008.

[15] N. K. Jerne, "Towards a network theory of the immune system," *Annales Dimmunologie*, vol. 125, no. (1-2), p. 373, 1974.

[16] Y. Huang, N. Bessis, P. Norrington, P. Kuonen, and B. Hirsbrunner, "Exploring decentralized dynamic scheduling for grids and clouds using the community-aware scheduling algorithm," *Future Generation Computer Systems*, vol. 29, no. 1, pp. 402–415, 2013.

[17] P. Ramachandran, B. Zoph, and Q. V. Le, *Searching for Activation Functions*, 2017.

[18] I. H. Witten and E. Frank, *Data Mining. Practical Machine Learning Tools & Techniques with Java Implementations*, vol. 13, 2005.

[19] B. Karlik and A. V. Olgac, *Performance Analysis of Various Activation Functions in Generalized MLP Architectures of Neural Networks*, 42, Cambridge University Press, 2010.

[20] D. E. Goldberg, "Genetic algorithms in search, optimization, and machine learning," *Choice Reviews Online*, vol. 27, no. 02, pp. 27-0936–27-0936, 1989.

[21] W. Gong, Á. Fialho, Z. Cai, and H. Li, "Adaptive strategy selection in differential evolution for numerical optimization: an empirical study," *Information Sciences*, vol. 181, no. 24, pp. 5364–5386, 2011.

[22] Q. Lin et al., "A novel hybrid multi-objective immune algorithm with adaptive differential evolution," *Computers & Operations Research*, vol. 62, pp. 95–111, 2015.

[23] "Standard Task Graph Set," http://www.kasahara.elec.waseda.ac.jp/schedule.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 6283957, 10 pages
https://doi.org/10.1155/2018/6283957

Research Article

A Mobile Fog Computing-Assisted DASH QoE Prediction Scheme

Hongyun Zheng[1], **Yongxiang Zhao**[1], **Xi Lu**[1,2] **and Rongzhen Cao**[1,3]

[1]*School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China*
[2]*Department of Information Technology, Nanjing Bank, Nanjing 210008, China*
[3]*Department of Communication Patent Examination, Patent Office of State Intellectual Property Office, Beijing 100081, China*

Correspondence should be addressed to Hongyun Zheng; hyzheng@bjtu.edu.cn

Received 4 May 2018; Revised 28 July 2018; Accepted 5 August 2018; Published 28 August 2018

Academic Editor: Xiaowen Gong

Copyright © 2018 Hongyun Zheng 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.

Video service has become a killer application for mobile terminals. For providing such services, most of the traffic is carried by the Dynamic Adaptive Streaming over HTTP (DASH) technique. The key to improve video quality perceived by users, *i.e.*, Quality of Experience (QoE), is to effectively characterize it by using measured data. There have been many literatures that studied this issue. Some existing solutions use probe mechanism at client/server, which, however, are not applicable to network operator. Some other solutions, which aimed to predict QoE by deep packet parsing, cannot work properly as more and more video traffic is encrypted. In this paper, we propose a fog-assisted real-time QoE prediction scheme, which can predict the QoE of DASH-supported video streaming using fog nodes. Neither client/server participations nor deep packet parsing at network equipment is needed, which makes this scheme easy to deploy. Experimental results show that this scheme can accurately detect QoE with high accuracy even when the video traffic is encrypted.

1. Introduction

Video service has become a killer application for mobile terminals and most of video traffic is carried by the Dynamic Adaptive Streaming over HTTP (DASH) technique. Mobile video traffic, which accounted for 60% of the total mobile traffic in 2016, is expected to rise to 78% by 2021 [1]. This significant growth is accompanied by the wide adoption of DASH standards [2, 3]. DASH has distributed video on large scale owing to the reuse of the existing HTTP infrastructure and ability to penetration through firewall. It offers video viewers the possibility of avoiding video playout interruptions in case of variations in terms of network conditions and adaptive changes in video bit rate.

In order to provide better Quality of Experience (QoE) for video users, network operators have to understand and monitor the video quality perceived at users, which has become a hot topic in recent years [4]. Some existing solutions use measuring mechanisms at client/server side to probe the QoE [5, 6]. However, these solutions are unfeasible for network providers because they are not able to access the measuring results at client/server side. Some other solutions investigated how to measure the QoE inside the network. These network-based solutions relied on deep packet inspection (DPI) or deep packet parsing (DPP) to evaluate the QoE [7–9]. However, more and more video services are being encrypted in order to protect user privacy [10], which means that these solutions will not work well soon.

Mobile Fog Computing (MFC), which brings the computing capabilities close to mobile users, provides a potential solution to probe users' QoEs at network edge without weakness of the above work [11]. In this paper, we propose a Fog-assisted Real-time QoE Prediction (FRQP) approach to enable network provider to predict users' QoEs with slightly increased computing power. Specifically, we deploy a probe mechanism at fog nodes to observe bidirection video traffic which enables us to infer users' QoEs according to the temporal features of the bidirection video traffic. FRQP is based on the network-measured traffic, which means that FRQP can work well without client/server's participation. FRQP also does not need deep packet parsing since it only observes packet header information.

Our contributions in this paper are as follows: (1) we design an MFC-assisted architecture, which uses fog computing capability to predict QoE from two-way traffic; (2) we for the first time divide the normal playing duration of a video into two subphases so as to effectively predict user QoE; (3) we creatively introduce the concept of request distance to characterize the density of request packets so as to avoid false detection of rebuffering events; (4) we conduct experiments and the results validate the effectiveness of our proposed approach.

The remainder of this paper is organized as follows. Section 2 presents the scenario under study and how DASH works. Section 3 presents the challenge in QoE prediction. Section 4 describes our QoE prediction method. Section 5 presents our experiments and evaluation results. Section 6 introduces related work, and conclusions and future work are given in Section 7.

2. Application Scenario and Working Mechanism of DASH

2.1. Application Scenario. In this paper, we propose a fog-assisted real-time QoE predicting approach to enable network provider to predict users' QoE with slightly increased computing power. Its application scenario is shown in Figure 1. In the figure, the mobile user device hosts a DASH client and connects to Access Node (AN), which can be a wireless network such as Wi-Fi and LTE. The AN, as a fog node, connects to DASH server via backhaul network, *e.g.*, the Internet. Both fog node and backhaul network are managed by network operators.

We deploy a probe mechanism at fog nodes to observe the bidirection video traffic which enables us to infer users' QoE according to the temporal features of the bidirection video traffic. Specifically, it periodically collects the bidirection video traffic and predicts users QoE from the traffic. The details of the prediction algorithm will be described in Section 4. The fog node can report the predicted QoE results to cloud that can allocate network resource accordingly in order to improve user QoE. In this paper, we do not care how the fog node communicates with the cloud but focus on how to predict user QoE.

The benefit of deploying probe mechanism at fog node is as follows. The probe mechanism needs accurate observation of temporal feature of bidirection video traffic to infer the user QoE. The nearer the probe mechanism is located away from client, the less chance cross traffic interferes with the video traffic.

2.2. Working Mechanism of DASH. In order to better understand the idea behind our proposed approach, we will first describe the working mechanism of DASH in this subsection.

Figure 2 shows an abstracted model of DASH delivery system. The DASH server encodes the video file into multiple versions with different qualities and slices each video file into video chunks with the same playtime. The DASH server uses HTTP to provide video services. When a client pushes the start button, it sends the server a HTTP GET message for fetching the corresponding video chunks. The fetched chunks will be kept in local buffer at the client. When the buffer has received enough number of chunks, the local player will start to play on the screen by withdrawing the chunks from the buffer continuously.

After initialization, the video client enters steady state with normal video playing, which is usually divided into two states [12]: ON and OFF, as shown in the down part of Figure 3. With the client fetching chunks from server continuously, the number of chunks in buffer, which we call buffer size, also increases. Once the buffer size reaches the max threshold, the client will stop downloading and this state of client is called OFF state. When the buffer size drops to the min threshold because of the player's continuous withdrawing of chunks from the local buffer, the client begins to send request messages to the server for starting downloading again. The state in which the client fetches data from server continuously is ON state. From Figure 3 it can be seen that, corresponding to the working states, the traffic between the client and server exhibits ON-OFF pattern and the ON-OFF pattern is regular. Accordingly, the buffer size at the client oscillates regularly (see the top part of Figure 3).

In a video viewing session, the video playing will be interrupted when the buffer is drained out. If this happens, the playback will be frozen and the client will enter initialization to locally accumulate enough video chunks again. The video interruption is referred to as rebuffering or frozen event. Rebuffering event is an important factor affecting end-user perceived QoE [13, 14].

3. Design Challenge

Because of the importance of rebuffering event in video streaming services, in this paper we use rebuffering event as the key metric for characterizing QoE at users. This work aims to use probe mechanism at fog node to predict whether a rebuffering event occurs at client. According to the above-introduced DASH working mechanism, it seems that we can easily predict user's QoE by monitoring the ON-OFF pattern of traffic between user and DASH server. For example, in Figure 3, it can be seen that video traffic presents apparent ON-OFF pattern. In ON state, a client continuously issues HTTP request messages to video server and the server continuously sends data chunks to the client. In OFF state, there exists no traffic between the client and server. From these observations, we can reasonably assume that video is played smoothly if the probe mechanism observes ON-OFF traffic between client and server.

Unfortunately, in reality, inferring the occurrence of rebuffering event is not easy. Next, we will set up experiments in a lab controlled environment to show the challenge.

The experiment settings are as follows. We started a DASH client under an interface with sufficient bandwidth to enable a DASH client to play video smoothly; we also limited the bandwidth of terminal interface in a certain time period to produce rebuffering events. In the experiments, we captured the packets transferred between the client and DASH server and also buffer size at the client. Specifically, we split time into fixed sampling intervals such that there is at most one request falling into a sample interval and counted

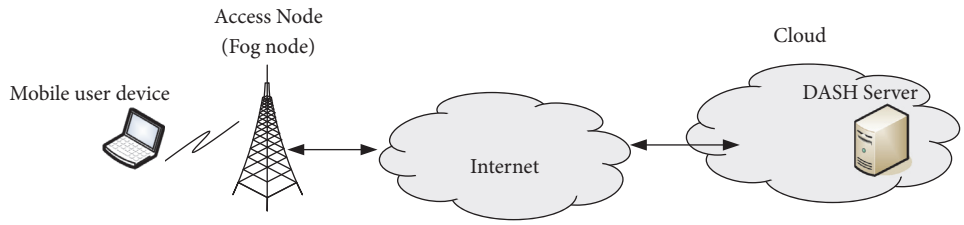


Figure 1: An MFC application scenario for QoE prediction.

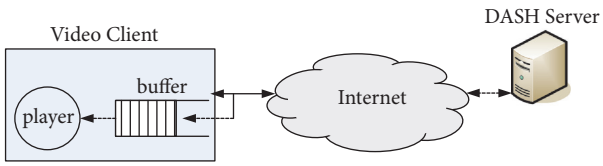


Figure 2: An abstracted model of DASH delivery system.

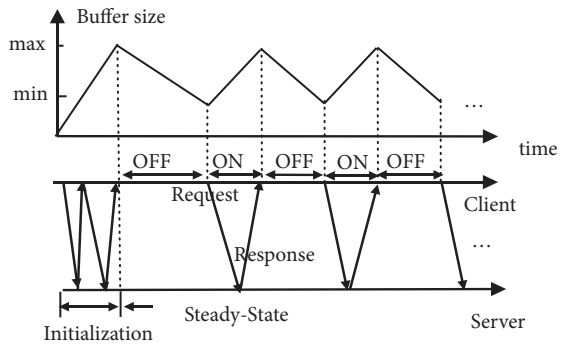


Figure 3: Request-response and buffer size timing sequence in a video viewing session [12].

the amount of bytes fetched from the server in each sampling interval as the download volume.

Figure 4(a) shows the variation of bandwidth observed at user terminal interface during the video playback. The horizontal axis is time, and the vertical axis is the bandwidth. We limit the bandwidth to 20 Mbps in duration 2 (*i.e.*, from 120 to 184 seconds about). Figure 4(b) shows the buffer size variation with time. In this figure, according to the buffer size we divide the playback into three phases: Steady State (SS), Closing Frozen (CF), and frozen phase. In SS phase, the buffer size is between the min and max threshold. In CF phase, the buffer size is smaller than the minimum threshold but larger than zero while it is zero in frozen phase. Figure 4(c) shows the download volume variation with time. We also used circle points to mark those time instants at which the client issues request packets. To ease the visualization, we move them vertically as shown in Figure 4(d). It is worth noting that it takes time for the client to respond to the bandwidth variation. Hence there exists delay between the variation of bandwidth and those of buffer size and bidirection traffic.

There is apparent ON-OFF traffic pattern during smooth video playing. In Figure 4, it can be seen that the player works smoothly in durations 1 and 3, since there are sufficient bandwidths (see Figure 4(a)). Accordingly, the buffer size oscillates regularly. Moreover, the download and request traffic also show apparent ON-OFF pattern, as shown in Figures 4(c) and 4(d), respectively, which is consistent with what we discussed in Section 2.

However, there also exists apparent ON-OFF traffic pattern when rebuffering event occurs, which means that we cannot merely use ON-OFF traffic pattern to predict occurrence of rebuffer event. In time duration 2 (see Figure 4(a)), the client will encounter frozen events because the bandwidth has been limited to 20 Mbps. The buffer is drained out in this duration (see Figure 4(b)). This is why we call this duration as frozen phase. From Figures 4(c) and 4(d), it can be seen that the traffic in both directions in frozen phase is ON-OFF. It means that we cannot simply use ON-OFF traffic pattern to figure out if a rebuffer event will occur or not because there is apparent ON-OFF traffic pattern in both SS and rebuffering cases.

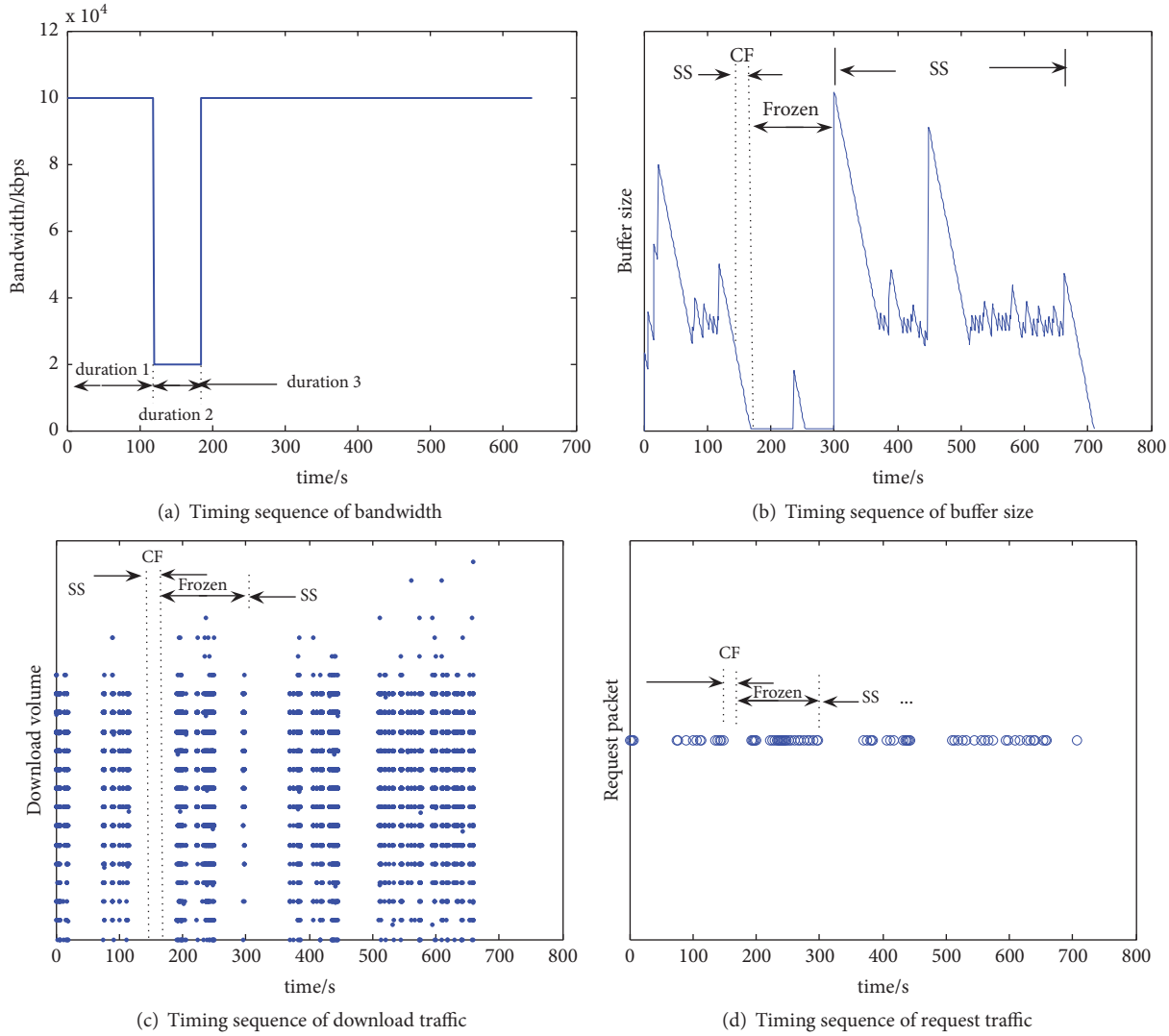
One simple way to tackle this problem is to use the request density to infer the possible occurrence of frozen events as described in [15]. For example, request density during frozen phase in Figure 4(d) is denser than that in SS phase. However, this method would cause false detection of rebuffering event in the following situation: in the duration labelled as CF in Figure 4(b), the video playing is smooth; however, the request sequence is denser (see Figure 4(d)). Thus, in this case, this method will falsely report occurrence of rebuffering event.

To address the above challenge, in this paper, we propose a method to identify rebuffering event by using combination of two-way traffic in an online video viewing session. The details will be discussed in the next section.

4. Prediction Method

In this section, we first explain our definition of the client working phase and establish the relationship between working phases at a user and traffic pattern observed. Then we show detailed method to characterize the traffic pattern. Finally, we show how the scheme proposed in this paper works.

4.1. Redefinition of Client Working State. According to the DASH working mechanism, we divide the client operation into three phases, each of which corresponds to a different level of buffer state. These phases and their definitions are shown in first column of Table 1. In this table, we also give the traffic pattern and its corresponding QoE by observing the experiment results in Figure 4. For example, during the



(a) Timing sequence of bandwidth

(b) Timing sequence of buffer size

(c) Timing sequence of download traffic

(d) Timing sequence of request traffic

Figure 4: The playback of a video. Here, "SS" means steady state, "CF" means imminent-frozen state, and "frozen" means staying in frozen state.

Table 1: Different working phases and their corresponding traffic characteristics.

Working phases	Traffic patterns		QoE labels
	Download traffic	Request density	
SS: buffer size> min threshold	ON-OFF	sparse	no re-buffering
CF: 0 <buffer size≤ minimum threshold	OFF	dense	no re-buffering
Frozen: buffer size=0	ON-OFF	dense	re-buffering

CF phase, we can see the following: first, it is seen that the download traffic is in OFF pattern (see Figure 4(c)); second, the client issues request densely (see Figure 4(d)); and, finally, the video is played smoothly because buffer size is above zero (see Figure 4(b)). Thus, the corresponding rows for the CF phase are "OFF", "dense", and "no rebuffering", respectively.

Existing work only classifies the operation of a client into two states according to the buffer size: buffer size above zero and equals to zero. However, to predict the occurrence of rebuffering events, we find that it is crucial to divide the client operation when buffer size is above zero into two substates, *i.e.*, SS and CF (see Table 1) because of the following reasons: traffic pattern of CF is different from SS and frozen; CF means imminent frozen although the client is not in frozen state yet.

The results in Table 1 motivate us to predict QoE by observing the traffic pattern of two-way traffic between client and video server during the playback of a video. However, the descriptive term of traffic patterns used in Table 1 is

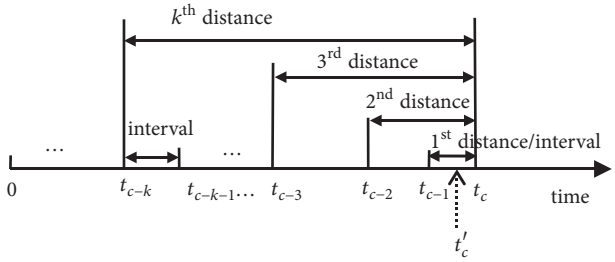


Figure 5: Request interval and request distance.

still qualitative which is infeasible in actual implementation of QoE prediction. During the playback of online videos, what we can observe is the time sequence of two-way traffic between client and server. Thus, we need to establish the relationship between time sequences of two-way traffic and corresponding traffic patterns, which will be explored by classification of machine learning in the following.

4.2. Metrics of Traffic Pattern

4.2.1. Download Throughput. It needs to quantize the download traffic; we use "download throughput" to describe the speed of traffic downloading. Assume an in-network observer can count the traffic volumes in both directions at discrete time instants. Specifically, $t_0, t_1, \ldots, t_{i-1}, t_i, \ldots$ are the sampling time instants. Denote the current time instant, which is the time instant when we predict QoE, as t_c. At time instant t_i, we define x_i^d as the amount of bytes transferred from the server to the client in time duration $[t_{i-1}, t_i]$ and define vector $X_c^d=[x_{c-w}^d, x_{c-w+1}^d \ldots, x_c^d]$, where w is length of observation time duration.

Denote the download throughput during interval $[t_{i-1}, t_i]$ with length $\Delta t_i = t_i - t_{i-1}$ as r_i^d, which is calculated as

$$r_i^d = \frac{x_i^d}{\Delta t_i} \tag{1}$$

where $r_i^d \geq 0$. The value of r_i^d may be zero since it is possible that no video content is downloaded during interval $[t_{i-1}, t_i]$.

We use moving average to smooth the download throughput. For the download throughput r_i^d at time t_i, the corresponding moving average download throughput is calculated as follows.

$$\overline{r_i^d} = \begin{cases} \delta \overline{r_{i-1}^d} + (1-\delta)\, r_i^d & i > 0 \\ r_0^d & i = 0 \end{cases} \tag{2}$$

where δ is a parameter for the moving average and in this paper is fixed to be 0.98. Thus, for time duration $[t_{c-w}, t_{c-w-1}, \ldots, t_c]$, we can get vectors $\mathrm{R}^d = [r_{c-w}^d, r_{c-w-1}^d, \ldots, r_c^d]$ and $\overline{\mathrm{R}^d} = [\overline{r_{c-w}^d}, \overline{r_{c-w-1}^d}, \ldots, \overline{r_c^d}]$.

4.2.2. Request Distance. A simple metric to describe the density of request is to use request intervals. For example, in Figure 5, assume we need to predict QoE at time instant t_c, also known as the current time instant, and $t_{c-1}, t_{c-2}, \ldots, t_{c-k}$ are the time instants at which k previous request packets are sent, where $k = 1, 2, \ldots, w$, w is length of observation time duration. The request interval is the time difference between two adjacent time instants of sending requests.

Table 2: Features extracted.

Features	Description
r_c^d	downloaded throughput at prediction instant t_c
$\overline{r_c^d}$	moving-average download throughput at prediction instant t_c
d_c^k	The k^{th} request distance at prediction instant t_c, $k = 1, 2, \ldots, w$

However, we find that different requests may have different contributions to predict the QoE at a specific time instant. For example, as shown in Figure 5, given request instant history $t_{c-k}, \ldots, t_{c-2}, t_{c-1}$, we find the time interval between request instant and current time instant contributes more to predict QoE. We will show this by the following experiment results.

In the absence of request interval, we introduce a new metric, known as request distance. The concept of request distance is also shown in Figure 5. Specifically, the k^{th} request distance ($k = 1, 2, \ldots, w$) refers to time difference between t_c and time instant at which the previous k^{th} request arrives, which is calculated as

$$d_c^k = t_{c-k} - t_c, \quad k = 1, 2, \ldots, w \tag{3}$$

where t_{c-k} is time instant at which the k^{th} previous request is sent. At current time instant, it is possible that no request arrives, just like time instant t_c'.

Figure 6 shows the CDF curves of request distance and request interval for SS and frozen phase. The horizontal axis is the discrete request distance, which is defined as request distance divided by sample interval. The vertical axis is the CDF. The two thin lines with label "$k = 1$" are CDF of request interval because when $k = 1$ the request distance is just the request interval. The two thick lines with label "$k = 7$" are the CDF when we observe, for example, the 7^{th} request distance. In Figure 6, we can see that the difference between the curve labelled with "SS ($k = 7$)" and its corresponding curve labelled with "frozen ($k = 7$)" is apparent, which means that we can discern state SS and state frozen by observing the request distance when $k = 7$. By contrast, according to the two thin lines, we cannot discern state SS and state frozen by observing the request interval because the two curves in this case are very close. Thus, it is more effective in figuring out if rebuffering event occurs using the request distance than request interval. In the next section we will further prove this conclusion by experiments.

4.3. Features Extraction and Selection. Table 2 summarizes the features extracted from the traffic patterns. We shall use information gain to evaluate the importance of each feature and select those most important features as the classifier inputs. For this purpose, the following operations will be taken.

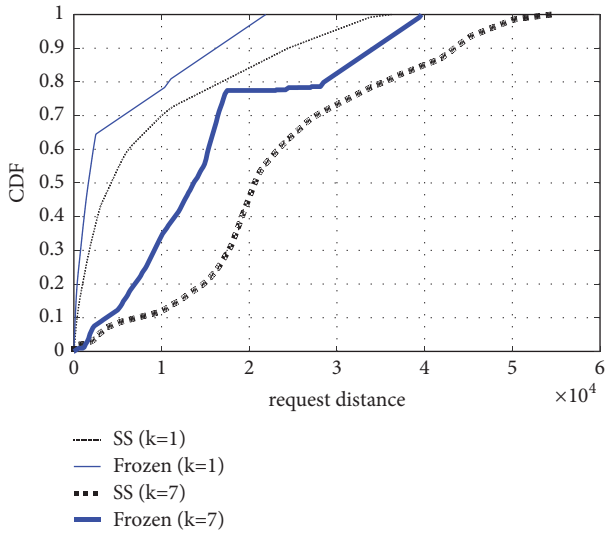


Figure 6: The CDF of request distance. The horizontal axis is the discrete request distance.

Firstly, we construct training set $D = \{r_c^d, \overline{r_c^d}, d_c^1, d_c^2, \ldots, d_c^w, Y\}$. To ease the presentation, we rename the items in D as $D = \{A_1, A_2, \ldots, A_{w+2}, Y\}$ in which A_n is the n^{th} feature ($1 \le n \le w + 2$) and $Y = \{y_1, y_2, \ldots, y_{w+2}\}$ is the class label vector for given features $\{A_1, A_2, \ldots, A_{w+2}\}$, where y_n is the class label of feature A_n and it is a QoE label in Table 1. In order to get the QoE label for each feature, we select some users to work as trainers to report their buffer size information using probe mechanism. Then the information will be translated into QoE labels.

Secondly, we calculate information gain to evaluate the importance of a feature. The information gain for feature n is defined as

$$G(D, A_n) = H(D) - H(D \mid A_n) \tag{4}$$

where entropy is

$$H(D) = -\sum_{y_j} P(y_j) \log_2 P(y_j) \tag{5}$$

where $P(y_j)$ denotes, in the whole training set, the probability that the class label y_j equals "1" or "0", which represent the video being frozen or not, respectively.

The conditional entropy is

$$H(D \mid A_n) = -\sum_{v} P(A_n \in D \mid A_n = v) H(D \mid A_n = v), \tag{6}$$

where v is a specific value of A_n. $P(A_n \in D \mid A_n = v)$ denotes the probability that $A_n = v$. And

$$H(D \mid A_n = v) = -\sum_{y_i} P(v, y_n = l_i) \log_2 P(v, y_n = l_i) \tag{7}$$

where $P(v, y_n = l_i)$ denotes the probability that the class label $y_n = l_i$ (l_i = "1" or "0") when $A_n = v$.

By this way, we obtain the information gain of each feature in D.

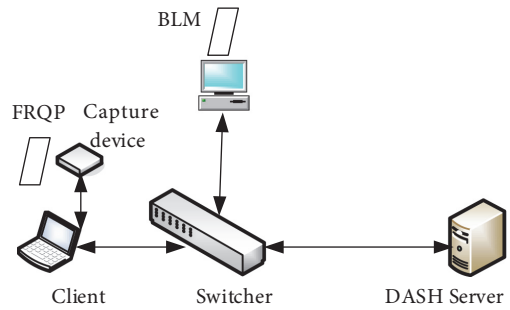


Figure 7: Experiment setup.

4.4. QoE Prediction Algorithm. Based on the features extracted and selected from the bidirection traffic, we propose a Fog-assisted Real-time QoE Prediction (FRQP) approach working as probe mechanism which is implemented at fog node as shown in Figure 1. The FRQP has two working states: training state and predicting state.

At the training state (offline phase), some users are selected to use special devices, each of which will periodically report its buffer size to FRQP, then FRQP translates the report into QoE labels. Then FRQP will train a classifier based on the reported buffer size working as QoE labels and features extracted from the observed bidirection video traffic.

At the predicting state (online phase), FRQP will predict a specific user's QoE by feeding the features extracted from this user's bidirection video traffic.

We will select a subset of features from D to reduce complexity of training state. Denote the number of features fed into classifier as m, which is less than or equal to $w+2$. We will sort the features in the decreasing order of information gain and select top m features as input of the classifier. The value of m will be tuned using experimental method.

5. Experiments and Evaluation

5.1. Experiment Setup. In order to evaluate the performance of FRQP, we set up a testbed in a controlled lab environment. Figure 7 shows the experiment setup. In Figure 7, an HP server running centos Linux 7 and Apache HTPP Sever acts as the DASH server. A mobile computer running windows 7 is used as the client. The mobile computer also acts as a capture device, on which a capture tool is installed to monitor traffic. FRQP, which runs as an application, is installed on the mobile computer as well. Another PC computer is used to install a bandwidth limitation module (BLM) to limit the bandwidth between the video server and client. The bandwidth limitation module is implemented with Iperf software, which sends background traffic at rate of 80 Mbps between the server and the client. The limited bandwidth is in the range of 20 Mbps and 100 Mbps.

At the server side, a video clip, *Big Buck Bunny*, is hosted and available for retrieval by the client. This video file, lasting for about 10 minutes, has twenty different representations, in which encoding bitrates range between 50 Kbps and 500 Kbps. These representations of the video are divided into 6 seconds of chunks. At the client side, a Google Chrome

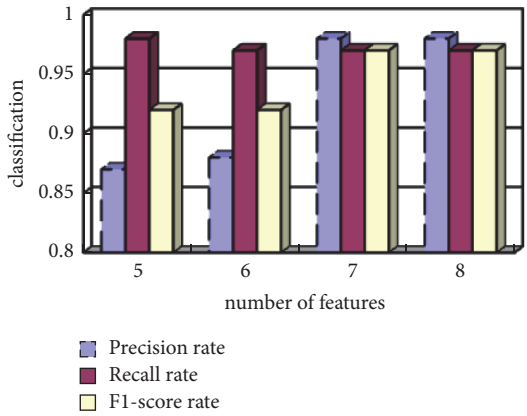


Figure 8: The classification results with different features.

Table 3: Information gain of different features.

Features	Information gain G
r_c^d	0.4988
r_c^d	0.3338
d_c^7	0.2209
d_c^9	0.2208
d_c^8	0.2111
d_c^6	0.1600
d_c^5	0.1254
d_c^4	0.1109
d_c^3	0.1074
d_c^2	0.0972
d_c^1	0.0659

browser runs, which is able to record the status of the playback, such as requested bitrate and buffer filling level. The information of buffer size is translated to QoE labels for training and evaluation.

5.2. Feature Evaluation. In order to select appropriate features for classification, we calculate the information gains of all features as shown in Table 2. Table 3 lists the eleven features with top highest information gain G. We get some insights from the result. First, the download throughput is dominant. Second, the k^{th} request distances are noticeable. Based on the ranking of the features, we select those features with top information gains for classification. Here the difficulty in decision is how many features listed in Table 3 should be selected for classification. We make a decision through experiments which will be described in the next subsection.

5.3. Classification Results. We study the impact of the number of features on the classification result. The results of decision tree classifier are shown in Figure 8. It can be seen that both of the precision and recall rate are high enough when the top eight features (see Table 3) are used as input; the performance improvement is insignificant when more features are used. We therefore use the top eight features in the following tests.

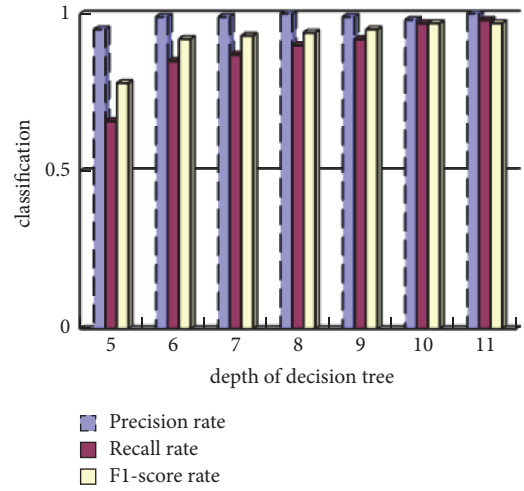


Figure 9: The classification results with different depths.

Using the selected features, we evaluate the performance and accuracy of different classification algorithms based on the machine learning tool Sklearn. Specifically, we compare the performance of five different machine learning algorithms: binary decision trees, random forest, support vector machine, naive Bayes, and classification based on linear regression. The results are based on 5-fold cross validation. In general we observe that random forest and decision trees perform better than the other three with satisfied classification rates while having less consumption of CPU time. From the perspectives of accuracy, simplicity, explainability, and execution speed, we finally adopt decision trees as our classifier.

We also study the impact of depth of decision tree on the performance of the classification. The classification result is depicted in Figure 9. The results show that depth of 11 is enough.

The prediction result with 8 features and 11-depth is shown in Table 4. The results show that rebuffering events can be recognized with a precision of 98% while there are 3% rebuffering events missed and just 1% of no rebuffering events are identified wrongly. Comparatively the results using request interval as features are also shown in Table 4. Here the number of request intervals is the same as that of request distance. It can be seen that, using request interval as the metric, there would be 21% of false detection of rebuffering events, although the detection of those no rebuffering events is accurate.

6. Related Work

In the literature, there have been many approaches for estimating or measuring QoE of online video streaming services. According to how the data collection works and where the approaches are implemented, existing work can be classified into the following three categories: approaches assessing QoE at the client/server side, approaches assessing QoE in the network, and hybrid approaches.

The first type of approaches estimates QoE based on measurement tools that run at the client/server to collect QoE

Table 4: Experiment results ("1"=rebuffering event, "0"=no-rebuffering event).

		Precision rate	Recall rate	F1-score rate
Request distance	1	0.98	0.97	0.97
	0	0.99	1.00	1.00
Request interval	1	0.79	0.90	0.84
	0	0.98	0.96	0.97

statistics [16]. However, the information collected by these tools is not accessible for the network providers, which make it difficult for them to guide network resource allocation according to up-to-date user QoE.

The second type of approaches can be further divided into two subcategories. The first subcategory relies on deep packet inspection [8] or logs obtained from a network node [15, 17, 18] to infer the QoE. In [8], the manifest files are parsed to obtain information for traffic prediction. Reference [19] gets the video information using packet traces. These existing work assessed QoE based on abundant information of video packets, e.g., complete manifest file and timestamps with respect to requests (e.g., HTTP requests, redirected HTTP requests, and the HTTP response) for each video chunk. The costs are efforts made in collecting and extracting such video information. A comprehensive overview on this topic was presented in [4, 20]. Meanwhile, as more and more video traffic is encrypted, it will impact the ability of operators to assess the user's QoE via this type of approaches. As for the second subcategory approaches, QoE is estimated by measuring the network-related QoS parameters such as throughput, loss rate, delay, and accordingly build model to map these network-related QoS parameters into user QoE [21]. Most of these works leverage machine learning (ML) technique to estimate QoE [21–25]. The study in [26] used network performance metrics, such as delay and packet losses, while [24] mapped application QoS (such as video bitrate, frame rate) to assess the QoE. The authors in [13] proposed a model to predict user engagement in terms of viewing time and number of visits using video application QoS as input. The video application related QoS include average bitrate, join time, buffering ratio, and rate of buffering, which are fed back from the client software at video viewers. An overview of QoE prediction based on QoS using machine learning techniques and more in-depth discussions were presented in [27, 28].

The third category combines the measurement at both client/server and network to estimate the QoE. The client/server report certain video playing setting to the network and the network accordingly infers the QoE based on such information [7, 29]. However, this type of approaches needs to modify video delivery protocols, which made the measurement hard to be deployed.

Unlike the above work, our method in this paper is more simple and practicable since it can predict QoE from network traffic even when it is encrypted. Moreover, all the above existing approaches work offline. However, in practice, it is desired for network providers to detect QoE in a real-time fashion and then allocate network resource to provide better services. Our proposed method in this paper meets this demand.

The authors in [10] measured QoE from encrypted traffic. The main difference between our work in this paper and that in [10] is that we use less information, *i.e.*, just bidirectional traffic quantity to make traffic prediction.

Currently, some studies about integrating edge computing in multimedia applications have appeared. The work in [30] highlights the potentials of using edge computing in multimedia services, interactive media applications, and video streaming. In [31], Mobile Edge Computing (MEC) server is used as a controlling component to implement the video caching strategy and also to adjust the video bitrates flexibly. The work in [32] designs and implements a video streaming service exploiting MEC functionalities. The study in [33] proposes an architecture for adaptive HTTP video streaming tailored to an MEC environment. In the proposed architecture, the adaptation algorithm runs as an MEC service, with an aim to relax network congestion while improving the quality of user experience. In [34], the authors discussed the network service migration from the cloud to fog nodes for video distribution with QoE support. The aforementioned works present a generic discussion on system architecture, cache, adaptive bitrate, and service migration for edge technologies. However, there is no video quality assessment.

7. Conclusions and Future Work

In this paper, we presented a novel method to predict buffering events in real time at network edge, *i.e.*, fog node. Our solution is based on the monitoring of network traffic, which means that it works without client and server's participations. In addition, our proposed solution does not need deep packet parsing. Experimental results show that our solution can accurately detect buffering events with about 98% accuracy.

In the future, we will explore how to extend our solution to work in multihop network environment and further how to accelerate the classification calculation. We will also study the communication between the fog node and cloud.

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 work was supported by NSFC (61271199) and the Fundamental Research Funds in Beijing Jiaotong University (2011JBZ003).

References

[1] Cisco, *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021 White Paper*, 2017.

[2] Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats, ISO/IEC Standard 23 009-1:2014, May 2014.

[3] 3GPP, Transparent end-to-end Packet-switched Streaming Service (PSS); Progressive Download and Dynamic Adaptive Streaming over HTTP (3GP-DASH)," 3GPP, International standard 3GPP TS 26.247, Dec 2015.

[4] P. Juluri, V. Tamarapalli, and D. Medhi, "Measurement of quality of experience of video-on-demand services: A survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 401–418, 2016.

[5] L. Plissonneau, E. Biersack, and P. Juluri, "Analyzing the impact of YouTube delivery policies on user experience," in *Proceedings of the 2012 24th International Teletraffic Congress, ITC 2012*, pp. 89–96, September 2012.

[6] R. Schatz, T. Hoßfeld, and P. Casas, "Passive YouTube QoE monitoring for ISPs," in *Proceedings of the 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2012*, pp. 358–364, July 2012.

[7] S. Tang, X. Qin, and G. Wei, "Analysis on the state of mobile HTTP video streaming at the client-side," in *Proceedings of the International Conference on Wireless Communications and Signal Processing, WCSP 2015*, October 2015.

[8] A. Farshad, P. Georgopoulos, M. Broadbent, M. Mu, and N. Race, "Leveraging SDN to provide an in-network QoE measurement framework," in *Proceedings of the IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS 2015*, pp. 239–244, May 2015.

[9] A. Finamore, M. Mellia, Z. Gilani, K. Papagiannaki, V. Erramilli, and Y. Grunenberger, "Is there a case for mobile phone content pre-staging?" in *Proceedings of the 2013 9th ACM International Conference on Emerging Networking Experiments and Technologies, CoNEXT 2013*, pp. 321–326, December 2013.

[10] G. Dimopoulos, I. Leontiadis, P. Barlet-Ros, and K. Papagiannaki, "Measuring video QoE from encrypted traffic," in *Proceedings of the 2016 ACM Internet Measurement Conference, IMC 2016*, pp. 513–526, November 2016.

[11] 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*, pp. 13–15, August 2012.

[12] S. Akhshabi, S. Narayanaswamy, A. C. Begen, and C. Dovrolis, "An experimental evaluation of rate-adaptive video players over HTTP," *Signal Processing: Image Communication*, vol. 27, no. 4, pp. 271–287, 2012.

[13] F. Dobrian, V. Sekar, A. Awan et al., "Understanding the impact of video quality on user engagement," *Computer Communication Review*, vol. 41, no. 4, p. 362, 2011.

[14] Conviva, "2014 viewer experience report," https://www.conviva.com/press-releases/conviva-releases-2014-viewer-experience-report/.

[15] T. Wu, R. Huysegems, and T. Bostoen, "Scalable network-based video-freeze detection for HTTP adaptive streaming," in *Proceedings of the 23rd IEEE International Symposium on Quality of Service, IWQoS 2015*, pp. 95–104, June 2015.

[16] P. Juluri, L. Plissonneau, and D. Medhi, "Pytomo: A tool for analyzing playback quality of YouTube videos," in *Proceedings of the 2011 23rd International Teletraffic Congress, ITC 2011*, pp. 304-305, September 2011.

[17] R. Huysegems, B. De Vleeschauwer, K. De Schepper et al., "Session reconstruction for HTTP adaptive streaming: Laying the foundation for network-based QoE monitoring," in *Proceedings of the 2012 IEEE 20th International Workshop on Quality of Service, IWQoS 2012*, June 2012.

[18] R. Schatz, T. Hossfeld, and P. Casas, "Passive YouTube QoE monitoring for ISPs," in *Proceedings of the 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2012*, pp. 358–364, July 2012.

[19] P. M. A. Kumar and S. Chandramathi, "Intelligent video QoE prediction model for error-prone networks," *Indian Journal of Science and Technology*, vol. 8, no. 16, pp. 1–9, 2015.

[20] V. Menkovski, A. Oredope, and A. Liotta, "Optimized online learning for QoE prediction," *International Journal of Advanced Media and Communication*, 2009.

[21] V. Menkovski, G. Exarchakos, and A. Liotta, "Online QoE prediction," in *Proceedings of the 2010 2nd International Workshop on Quality of Multimedia Experience, QoMEX 2010*, pp. 118–123, Norway, June 2010.

[22] M. Alreshoodi and J. Woods, "Survey on Qoe/Qos Correlation Models Formultimedia Services," *International Journal of Distributed and Parallel Systems*, vol. 4, no. 3, pp. 53–72, 2013.

[23] V. Menkovski, G. Exarchakos, A. Liotta, and A. C. Sánchez, "Quality of experience models for multimedia streaming," *International Journal of Mobile Computing and Multimedia Communications*, vol. 2, no. 4, pp. 1–20, 2010.

[24] P. Gastaldo, R. Zunino, and J. Redi, "Supporting visual quality assessment with machine learning," *EURASIP Journal on Advances in Signal Processing*, vol. 2013, no. 1, 2013.

[25] R. K. P. Mok, E. W. W. Chan, and R. K. C. Chang, "Measuring the quality of experience of HTTP video streaming," in *Proceedings of the 12th IFIP/IEEE International Symposium on Integrated Network Management, IM 2011*, pp. 485–492, Ireland, May 2011.

[26] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, and H. Zhang, "Developing a predictive model of quality of experience for internet video," in *Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication, ACM SIGCOMM 2013*, pp. 339–350, August 2013.

[27] S. Aroussi and A. Mellouk, "Survey on machine learning-based QoE-QoS correlation models," in *Proceedings of the 2014 IEEE International Conference on Computing, Management and Telecommunications, ComManTel 2014*, pp. 200–204, April 2014.

[28] Y. Chen, K. Wu, and Q. Zhang, "From QoS to QoE: A tutorial on video quality assessment," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 1126–1165, 2015.

[29] M. Mu, M. Broadbent, A. Farshad et al., "A scalable user fairness model for adaptive video streaming over SDN-assisted future networks," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 8, pp. 2168–2184, 2016.

[30] K. Bilal and A. Erbad, "Edge computing for interactive media and video streaming," in *Proceedings of the 2nd International*

Conference on Fog and Mobile Edge Computing, FMEC 2017, pp. 68–73, Spain, May 2017.

[31] X. Xu, J. Liu, and X. Tao, "Mobile edge computing enhanced adaptive bitrate video delivery with joint cache and radio resource allocation," *IEEE Access*, vol. 5, pp. 16406–16415, 2017.

[32] S. Salsano, L. Chiaraviglio, N. Blefari-Melazzi et al., "Toward Superfluid Deployment of Virtual Functions: Exploiting Mobile Edge Computing for Video Streaming," in *Proceedings of the 29th International Teletraffic Congress, ITC 2017*, pp. 48–53, September 2017.

[33] Y. Li, P. A. Frangoudis, Y. Hadjadj-Aoul, and P. Bertin, "A Mobile Edge Computing-based architecture for improved adaptive HTTP video delivery," in *Proceedings of the 2016 IEEE Conference on Standards for Communications and Networking, CSCN 2016*, Germany, November 2016.

[34] R. Denis, S. Matias, R. Juergen et al., "Service migration from cloud to multi-tier fog nodes for multimedia dissemination with QoE support," *Sensors*, vol. 18, no. 2, 2018.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 1818690, 8 pages
https://doi.org/10.1155/2018/1818690

Research Article

Transcoding Based Video Caching Systems: Model and Algorithm

Hongna Zhao,[1] Chunxi Li,[1] Yongxiang Zhao,[1] Baoxian Zhang,[2] and Cheng Li[3,4]

[1] *Beijing Jiaotong University, No. 3 Shangyuancun, Beijing 100044, China*
[2] *University of Chinese Academy of Sciences, No. 19A Yuquan Road, Beijing 100049, China*
[3] *The School of Computer and Information Engineering, Tianjin Chengjian University, China*
[4] *Memorial University of Newfoundland, St. John's Campus, Newfoundland, Canada A1B 3X5*

Correspondence should be addressed to Yongxiang Zhao; yxzhao@bjtu.edu.cn

Received 3 May 2018; Accepted 17 July 2018; Published 1 August 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Hongna Zhao 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 explosive demand of online video watching brings huge bandwidth pressure to cellular networks. Efficient video caching is critical for providing high-quality streaming Video-on-Demand (VoD) services to satisfy the rapid increasing demands of online video watching from mobile users. Traditional caching algorithms typically treat individual video files separately and they tend to keep the most popular video files in cache. However, in reality, one video typically corresponds to multiple different files (versions) with different sizes and also different video resolutions. Thus, caching of such files for one video leads to a lot of redundancy since one version of a video can be utilized to produce other versions of the video by using certain video coding techniques. Recently, fog computing pushes computing power to edge of network to reduce distance between service provider and users. In this paper, we take advantage of fog computing and deploy cache system at network edge. Specifically, we study transcoding based video caching in cellular networks where cache servers are deployed at the edge of cellular network for providing improved quality of online VoD services to mobile users. By using transcoding, a cached video can be used to convert to different low-quality versions of the video as needed by different users in real time. We first formulate the transcoding based caching problem as integer linear programming problem. Then we propose a Transcoding based Caching Algorithm (TCA), which iteratively finds the placement leading to the maximal delay gain among all possible choices. We deduce the computational complexity of TCA. Simulation results demonstrate that TCA significantly outperforms traditional greedy caching algorithm with a decrease of up to 40% in terms of average delivery delay.

1. Introduction

The explosive demand of online video watching from mobile users brings huge bandwidth pressure to cellular networks. A common way to relieve such pressure is to deploy cache servers close to the end users to help video diffusion [1, 2]. Caching algorithms can be categorized into two types: online algorithms and offline algorithms, which typically operate at small and large time scales, respectively. Typical online algorithms include Least Recently Used (LRU) [3] and Least Frequently Used (LFU) [4], both of which make caching decisions based on dynamically arrived user requests. In contrast, offline caching algorithms make caching decisions for all videos: caching each of them or not based on their historical data of user fetching without consideration of users' real-time requests. In this paper, we will focus on design of offline caching algorithm. Unless otherwise specified, the term caching algorithm in this paper means offline caching algorithm.

Traditional offline caching algorithms (e.g., [5, 6]) assume that all cached video files are independent and treat them separately. They compute each file's popularity by counting the time the file has been accessed in the past and cache the most popular files. However, different users may request different versions (also with different resolutions) of a video, and the contents of different versions of a video are not independent but relevant. Thus, caching of files in such cases may contain a lot of redundancy since one version of a video

can be utilized to produce some other versions of the video by using certain video encoding techniques such as scalable video coding (SVC) [7] and transcoding [8].

SVC has been used to improve caching performance [9]. SVC encodes a video into different layers. With SVC, one version of a video contains one or more video layers, and different quality versions of a video share certain low video layers. Based on this observation, the popularity can be tuned from a per-file perspective to a per-video-layer perspective. With such an understanding, [9] proposed a caching algorithm for providing video services layered by SVC so that caching decision is made on a per-layer level instead of a per-video level. However, SVC-based caching has the following disadvantages. First, it requires SVC-encoded video files stored on cache servers and also SVC-decoding capability at mobile terminals; however, due to the high decoding complexity and excessive overhead [7], SVC is not widely deployed in online VoD. This largely limits the wide application of SVC-based caching algorithm in reality. Second, the number of the quality versions supported by a SVC-encoded video exactly equals the number of layers in the video. This largely restricts the granularity of services that a SVC-encoded service platform can provide to users, who may desire various quality versions of a video.

Transcoding has also been used to improve the caching performance of an online VoD system. Different from SVC encoding/decoding, transcoding has been a mature technology for converting a video from high-quality to any low-qualities [8]. The conversion can be easily done at cache servers without involvement of mobile clients. Compared with SVC-based caching, transcoding based caching has two advantages. First, video quality transforming can be conducted on much finer granularity to meet diverse user requirements. Second, transcoding is compatible with existing video formats, without need to upgrade the software at mobile terminals (client side) and video source (source side). In [10], Shen et al. proposed a transcoding based online caching algorithm, with which a caching system transcodes the cached videos according to real-time user requests and makes cache replacement with LRU algorithm. However, this algorithm is just a simple combination of LRU and transcoding, which largely restricts its caching performance. Moreover, the use of LRU in [10] cannot ensure each video has at most one quality version in a cache server, and thus much redundancy still remains among cached files.

Recently, fog computing pushes computing power to edge of network to reduce distance between service provider and users. In this paper, we take advantage of fog computing and deploy cache at networks edge. Specifically, we focus on studying transcoding enabled caching. The objective is to enable cache servers to keep most valuable video versions so as to minimize the video delivery delay for video requests from all users subject to constraints of cache sizes and limited bandwidth on links between different base stations (for cooperative caching). We first formulate the transcoding based caching problem as integer linear programming problem. Then we propose a Transcoding based Caching Algorithm (TCA), which iteratively finds the placement leading to the maximal delay gain for video fetching among all possible choices by considering request arrival rates of videos, file sizes of different video versions, and also delivery delays between different cache servers. To reduce content redundancy, TCA restricts each cache server to keep at most one quality version of a video. We deduce the computational complexity of TCA. Simulation results demonstrate that our TCA algorithm can significantly reduce the video transmission delay by up to 40% compared with traditional offline greedy algorithm in [5].

The rest of the paper is organized as follows. Section 2 gives a brief review of related work. Section 3 describes the caching system model and formulates optimal transcoding based independent caching and cooperative caching problems, respectively. In Section 4, we propose the TCA caching algorithm. In Section 5, we perform simulations to evaluate the performance of TCA. We conclude this paper in Section 6.

2. Related Work

Traditional offline caching algorithms usually treat individual video files separately and tend to keep the most popular video files in cache [5, 6]. Reference [5] proposed a cooperative caching algorithm for a distributed cache system. Staring with an all-zero caching vector, this algorithm then iteratively updates the caching vector by placing a file to a cache server such that this placement leads to the maximum performance improvement, which is computed based on the popularity of each video. Reference [6] proposed a caching algorithm which makes use of both user interests and video popularities, in order to achieve a good balance between cache efficiency and user preference.

However, in the traditional caching algorithms, several files kept at a cache server may belong to the same video but they have different qualities and thus lead to a lot of redundancy since one version of a video can actually be used to produce some other versions of the video by using certain video encoding/decoding technique, such as SVC [7] and transcoding [8].

SVC encodes a video into different layers, and different quality versions of a video share certain low layers. Thus, for SVC-encoded videos, the popularity calculation is changed from a per-video perspective to a per-layer perspective. Following this direction, in [9], a SVC-based caching algorithm was proposed to improve the caching performance. However, due to the high decoding complexity and the additional overhead [7, 11], SVC is not widely deployed in online VoD in particular for resource-limited mobile terminals. Moreover, the amount of quality versions that a SVC-based video can be transformed is quite limited and it equals the number of its encoded layers. This largely restricts the service granularity that a SVC-based encoding system can provide.

Transcoding has been a mature technology, which can convert a high-quality video to a low-quality video in real time [12]. By applying transcoding at cache servers, video quality transforming can be conducted on much finer granularity to meet diverse user requirements, and it is compatible with existing video formats without need to upgrade the software at mobile clients or video sources.

In this aspect, [10] designed an online caching system, which uses LRU to make cache replacement and uses transcoding to covert cached video files. Reference [13] designed an augmented radio access network model to evaluate the performance of such transcoding based online caching systems and showed that the caching performance can be further improved. Reference [14] studied a multiple-parameter optimal model subject to the cache storage and transcoding capacity constraints and proposed a cooperative LRU-based online video caching algorithm (online JCCP), which uses the collaboration among the cache servers to further improve cache performance. However, in these studies, LRU is just simply combined with transcoding, and the caching or replacing of a file is still based on LRU itself, which cannot ensure that each video has at most one quality version cached in individual cache server. For example, when a high version needs to be cached, removing the low version of the file from the cache (if any) should be a better choice. However, the use of LRU in [10, 13, 14] fails to incorporate such operation. Thus, redundancy still remains in caches.

Transcoding also causes certain extra cost. Reference [15] pointed out this problem and proposed an online partial transcoding caching scheme used in cloud computing network with an aim to minimize the total extra cost caused by the transcoding and storage. Reference [16] in 2018 proposed a cloud-based architecture to allocate transcoding tasks among virtual machines in a cache system to decrease the cost of streaming service provider and pointed out transcoding based video caching is worth being further studied. Moreover, due to the significant improvement of capability of mobile edge computing [17, 18] and the appearance of new transcoding technologies [19, 20], it is now feasible to conduct full transcoding in caching systems at mobile edge.

In this paper, we focus on studying transcoding based offline caching algorithm. Our algorithm in this paper differs from the above work in the following ways. First, it restricts one cache server to keep at most one quality version of a video in order to remove unnecessary redundancy. Second, it made caching decision on the quality version of a video to be cached based on all the requests for different versions of the video.

3. System Model and Problem Formulation

In this section, we first give an overview of transcoding based video caching system for providing streaming services at mobile edges. Then, we illustrate the key idea for performing transcoding based caching. Finally, we formulate the optimal transcoding based caching problem.

3.1. System Model. We first introduce the caching system under study. As Figure 1 shows, there are N cache servers in the system, which are geographically distributed at the edge of the cellular network to provide online VoD services to mobile users covered by the base stations. We assume each cache server is attached to a base station via a short link with unlimited bandwidth, and for simplicity we assign the same sequence number to a cache server and its attached base station. Each cache server can transcode a cached video from a high-quality version to a low-quality version in real time.

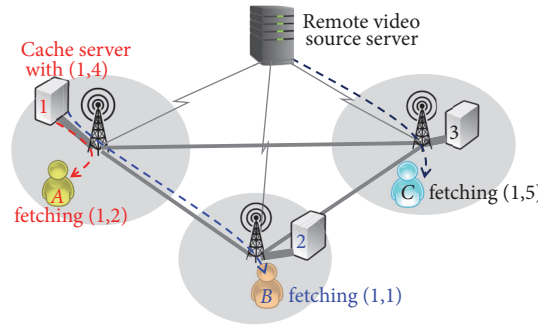


Figure 1: Example illustrating the application scenario with three cellular cells. In this figure, combination of two figures, in the form of (x, y), represents video x's y version.

These cache servers are connected to a remote video source server, which has all the quality versions of all the videos that users may request, via the backhaul network. In this paper, each user is assumed to be covered by one base station and can fetch video data from its local cache server, the video source server, or another local cache server (in the case of cooperative caching).

We next introduce a typical application scenario as shown in Figure 1 to illustrate the operation of vide fetching in such a system. As the figure shows, a user (say A) sends a request for a video quality version labeled by $(1, 2)$, i.e., a quality version 2 of video 1, to its local cache server 1, which just has a video quality version $(1, 4)$. Therefore, cache server 1 can satisfy the request by transcoding $(1, 4)$ to $(1, 2)$ since the locally cached version is higher than the requested version. We assume the transmission rate between a mobile user and its local base station is high enough such that the delivery delay for fetching a video from local cache server to its users is negligible, and we further assume transcoding can be performed in real time on a cache server such that we do not consider the transcoding delay.

Furthermore, assume the $N = 3$ cache servers in the system work cooperatively. A user, say B in Figure 1, who requests $(1, 1)$ but cannot get it directly from its local cache server 2, can fetch the video also from server 1, which transcodes $(1, 4)$ to $(1, 1)$ and send $(1, 1)$ to user B via base station 2. Along the path from server 1 to user B, we assume the link between base station 1 and base station 2 will allocate a certain bandwidth for each session for such video transmission and also cause certain delivery delay. Finally, if the request for $(1, 5)$ of a user C cannot be satisfied by any local cache servers, he/she needs to fetch it from the remote source server. Along the path from the remote video source server to user C, we assume the backhaul link will allocate a certain bandwidth for each session of such video transmission and also cause certain delivery delay.

Our objective here is to find a caching vector to place video files on different cache servers, while minimizing the average delivery delay of video transmissions for all users. Such optimization has great challenges on computation complexity. For example, considering there are five levels of qualities with either of two given videos, assume each one of

Table 1: Symbol definition.

Symbols	Definition
$\mathcal{N}$	A set of all cache servers.
$\mathcal{V}$	A set of all videos.
$\mathcal{Q}$	A set of all quality versions of a video.
C_n	Cache space of server n.
S_{vq}	Size of file (v, q), i.e., quality version q of video v.
λ_{nvq}	Number of requests for file (v, q) from the users under base station n.
d_n	Unit data delivery delay from remote source server to a user under base station n.
$d_{nn'}$	Unit data delivery delay from server n to a user under base station n'
D_n	Delivery delay of server n for serving all its local requests in independent caching.
D_{nv}	Delivery delay of server n for serving its local requests for all versions of video v in independent caching.
D^c	Delivery delay for serving all requests from all users in cooperative caching.
D_v^c	Delivery delay for serving the requests from all users for all versions of video v in cooperative caching.
X	Caching vector, where each element x_{nv} equals the quality version of video v to be cached at server n while $x_{nv} = 0$ means no file related to video v is cached at server n.

the three cache servers in Figure 1 has a limited space equal to the maximum size of the video versions with the highest quality. Then, for the 2 videos each with 5 different qualities, each cache server needs to select from the $6 * 6$ version choices one video or two videos to store, where each digital number of 6 corresponds to the 5 quality versions of either file plus a zero quality meaning the video is not to be stored. Moreover, if the three cache servers cooperate, the video version choices for them are $6^{2\times3}$; let us further generalize this computation as follows: in a real cache system, assume V is the total number of the videos each having Q levels of qualities and N is the total number of cache servers, the video version choices for these cache servers are $(Q + 1)^{V\times N}$, a huge amount, and thus such optimization problem is usually too complex to be solved in polynomial-time.

In the above system model, we assume transcoding can be performed in real time on a cache server without delay. In fact, the transcoding delay is related to the capacity of the transcoding server as well as the load of the transcoding processes. We will leave this problem in our future work.

Next, we will first define variables and parameters used and then formulate the problem in the following subsection; we will then propose a heuristic algorithm to solve this problem in the next section.

3.2. Symbols and Parameters. Before formulating the problem under study, we list the symbols and parameters used hereafter in Table 1.

We denote the set of cache servers by $\mathcal{N} = \{n \mid n = 1, 2, \ldots, N\}$, where N is the total number of cache servers. We assume a cache server $n, n \in \mathcal{N}$ has a space of C_n bytes for caching video files.

Assume there are in total V videos in the system, each having Q different versions to be requested, and we use $\mathcal{V} = \{1, 2, \cdots, V\}$ and $\mathcal{Q} = \{1, 2, \ldots, Q\}$ to denote the set of all the videos and the set of all the video versions, respectively. Each quality version of a video, denoted by (v, q), $v \in \mathcal{V}$, $q \in \mathcal{Q}$, has a given size of S_{vq} bytes, and we assume $S_{v1} < S_{v2} < \cdots < S_{vQ}$, $\forall v$. The number of requests for each video version (v, q) from the users under base station n is denoted by λ_{nvq}, which is assumed to be known in advance like in [5, 9].

Delivering a file may cause certain delivery delay. Specifically, delivering a video with a size of S_{vq} from the remote video source server to a user in the cell of base station n will cause a delay of $S_{vq}d_n$, where d_n is the delay for delivering one unit of video data from the source server to the user via base station n, due to the limited bandwidth assigned on backhaul link for the transmission. Furthermore, in the situation where these cache servers work cooperatively, delivering a video with a size of S_{vq} from a server n' to a user associated with base station n will cause a delay of $S_{vq}d_{nn'}$, where $d_{nn'}$ is the delay for delivering one unit of video data from cache server n' to the user via base station n, due to the limited bandwidth assigned on the link between base station n' and base station n for the transmission. We denote a caching vector by an set of integers $X = \{x_{nv} \mid n \in \mathcal{N}, v \in \mathcal{V}, x_{nv} \in \mathcal{Q} + \{0\}\}$, where each element x_{nv} represents the quality version of video v cached at server n, and specially, $x_{nv} = 0$ means server n does not have any version of video v. We use $X_n = \{x_{nv} \mid v \in \mathcal{V}\}$ to represent the caching vector for server n.

3.3. Formulation of Caching with Transcoding

3.3.1. Independent Caching with Transcoding. We first consider the situation that cache servers operate independently.

In this situation, a user can only fetch a video file either from its local cache server (with high priority) or from the remote video source in the cloud (with low priority). Our goal is to find an optimal caching vector X_n for each individual cache server n ($n \in \mathcal{N}$) while minimizing the average delivery delay of all the video fetching of users covered by the base station n. Since the average delivery delay equals the total delivery delay of all the video fetching divided by the total number of requests, and the total number of requests is assumed to be a constant, the objective of minimizing the average delivery delay is equivalent to minimizing the total delivery delay (denoted by D_n). The total delivery delay D_n can be computed as follows:

$$D_n = \sum_{v \in \mathcal{V}} \sum_{q \in \mathcal{Q}} \lambda_{nvq} S_{vq} d_n I^{x_{nv} < q}, \tag{1}$$

where $I^{\text{condition}}$ is an indicator function which equals 0 and 1, respectively, when the "condition" equals false and true. We explain (1) as follows. First, let us consider a given video quality version (v, q) requested by users under base station n. If the cached version for video v at server n can satisfy these requests, i.e., $x_{nv} \geq q$, we have $I^{x_{nv} < q} = 0$, and these users can fetch (v, q) from the local cache server directly without delay; otherwise, we have $I^{x_{nv} < q} = 1$, and these users have to fetch it from the remote video source server with a delivery delay $\lambda_{nvq} S_{vq} d_n$, where λ_{nvq} is the total number of the requests for file (v, q), and $S_{vq} d_n$ is the video delivery delay for each such request. Considering all the versions of video v to be requested, we have the following total delivery delay (denoted by D_{nv}) for fetching all these versions.

$$D_{nv} = \sum_{q \in \mathcal{Q}} \lambda_{nvq} S_{vq} d_n I^{x_{nv} < q}. \tag{2}$$

Considering all the videos and all their quality versions to be requested, we have formula (1).

Finally, we formulate the optimal transcoding based independent caching at a cache server n as follows:

$$\begin{aligned} \underset{X_n}{\text{minimize}} \quad & D_n \\ \text{subject to} \quad & \sum_{v \in \mathcal{V}} \sum_{q \in \mathcal{Q}} S_{vq} I^{x_{nv} = q} \leq C_n \\ & x_{nv} \in \mathcal{Q} + \{0\}, \quad \forall v \in \mathcal{V}. \end{aligned} \tag{3}$$

Formulation (3) has two constraints. First, the size of the total cached files at a server n must not exceed the cache space C_n. Second, the value of x_{nv} must be an integer, and it indicates this is an integer programming problem. As having been mentioned in Section 3.1, the computation complexity for (3) is $(Q+1)^V$ for each cache server; such computation is too complex to be solved in polynomial-time for large V.

3.3.2. Cooperative Caching with Transcoding. We then consider the situation where the cache servers work cooperatively with each other for providing high-quality online VoD services. In this situation, among all the servers including the local cache servers and the remote video source server, a user can fetch a video file from a best server which can satisfy the request while leading to the minimal delivery delay. Our goal is to find a caching vector X for the local caching system while minimizing the average delivery delay of all users. Again, this objective is equivalent to minimize the total delivery delay (denoted by D^c) of all video fetching requests.

The total delivery delay D^c can be computed as follows:

$$D^c = \sum_{\substack{n \in \mathcal{N} \\ v \in \mathcal{V} \\ q \in \mathcal{Q}}} \left\{ \lambda_{nvq} S_{vq} d_n \prod_{n' \in \mathcal{N}} I^{x_{n'v} < q} + \left(1 - \prod_{n' \in \mathcal{N}} I^{x_{n'v} < q}\right) \lambda_{nvq} S_{vq} \min_{n' \in \mathcal{N}'} d_{nn'} \right\}, \tag{4}$$

where $\mathcal{N}' = \{n' \mid n' \in N, I^{x_{n'v} < q} = 0\}$ is the set of the cache severs which can satisfy the requests for a given video version (v, q).

We explain (4) as follows. Let us first compute the delivery delay for a given combination of n, v, and q, i.e., transmitting video (v, q) to satisfy all user requests under base station n. On one hand, if none of the cache servers can satisfy the user requests, we have $I^{x_{n'v} < q} = 1, \forall n' \in \mathcal{N}$, and thus have $\prod_{n' \in \mathcal{N}} I^{x_{n'v} < q} = 1$. Thus, the users have to fetch (v, q) from the remote video source server, which leads to a delivery delay of $\lambda_{nvq} S_{vq} d_n$. On the other hand, if at least one cache server n' can satisfy the user requests, we have $I^{x_{n'v} < q} = 0$ for each such server n' and thus have $(1 - \prod_{n' \in \mathcal{N}} I^{x_{n'v} < q}) = 1$. Then, among all such cache severs $\mathcal{N}'$, we choose the cache server with the minimal delay to base station n to satisfy the requests of video (v, q), and we then have the total delivery delay as $\lambda_{nvq} S_{vq} \min_{n' \in \mathcal{N}'} d_{nn'}$. Here, we assume the delay between a pair of cache servers is smaller than that between the remote source server and a local cache server. Considering all the versions of video v to be requested, we have the following total delivery delay (denoted by D_v^c) for fetching all these versions:

$$D_v^c = \sum_{\substack{n \in \mathcal{N} \\ q \in \mathcal{Q}}} \left\{ \lambda_{nvq} S_{vq} d_n \prod_{n' \in \mathcal{N}} I^{x_{n'v} < q} + \left(1 - \prod_{n' \in \mathcal{N}} I^{x_{n'v} < q}\right) \lambda_{nvq} S_{vq} \min_{n' \in \mathcal{N}'} d_{nn'} \right\}. \tag{5}$$

Considering all the videos and all their quality versions to be requested, we have formula (4).

Finally, we formulate the optimal transcoding based cooperative caching problem as follows:

$$\begin{aligned} \underset{X}{\text{minimize}} \quad & D^c \\ \text{subject to} \quad & \sum_{v \in \mathcal{V}} \sum_{q \in \mathcal{Q}} S_{vq} I^{x_{nv} = q} \leq C_n, \quad \forall n \in \mathcal{N} \\ & X_{nv} \in \mathcal{Q} + \{0\}, \quad \forall n \in \mathcal{N}, \ \forall v \in \mathcal{V}. \end{aligned} \tag{6}$$

The two constrains in (6) are similar to those in (3). As having been mentioned in Section 3.1, the computation complexity for (6) is $(Q+1)^{V \times N}$; such computation is usually too complex to be solved in polynomial-time.

4. Transcoding Based Cache Algorithm

In this section, we design a transcoding based caching algorithm (TCA).

(1) $E = \{e_{nvq} \mid \forall n \in \mathcal{N}, \forall v \in \mathcal{V}, \forall q \in \mathcal{Q}\}$
(2) $E_n = \{e_{nvq} \mid \forall v \in \mathcal{V}, \forall q \in \mathcal{Q}\}, n \in \mathcal{N}$
(3) $X \longleftarrow \{0, 0, \cdots, 0\}$
(4) $X_n \longleftarrow \{0, 0, \cdots, 0\}, \forall n \in \mathcal{N}$
(5) **for** $i = 1, 2, ..., N \times V \times Q$ **do**
(6) $\quad e_{\alpha\beta\gamma} = \text{argmax}_{g \in E} J_X(g)$
(7) $\quad$ **if** $J_X(e_{\alpha\beta\gamma}) = 0$ **then**
(8) $\quad\quad$ **break**
(9) $\quad$ **end if**
(10) $\quad C_\alpha \longleftarrow C_\alpha - S_{\beta\gamma} + S_{\beta x_{\alpha\beta}}$
$\quad\quad$ /*note: $S_{\beta x_{\alpha\beta}} = 0$ if $x_{\alpha\beta} = 0$*/
(11) $\quad x_{\alpha\beta}$ of $X \longleftarrow \gamma$
(12) $\quad x_{\alpha\beta}$ of $X_\alpha \longleftarrow \gamma$
(13) $\quad E \longleftarrow E \setminus e_{\alpha\beta q}, \forall q \leq \gamma$
(14) $\quad E_\alpha \longleftarrow E_\alpha \setminus e_{\alpha\beta q}, \forall q \leq \gamma$
(15) $\quad$ **for** $e_{\alpha vq} \in E_\alpha$ **do**
(16) $\quad\quad$ **if** $C_\alpha + S_{vx_{\alpha v}} < S_{vq}$ **then**
(17) $\quad\quad\quad E \longleftarrow E \setminus e_{\alpha vq}$
(18) $\quad\quad\quad E_\alpha \longleftarrow E_\alpha \setminus e_{\alpha vq}$
(19) $\quad\quad$ **end if**
(20) $\quad$ **end for**
(21) $\quad$ **if** $|E| = 0$ **then**
(22) $\quad\quad$ **break**
(23) $\quad$ **end if**
(24) **end for**
(25) Ouput X

Algorithm 1: Procedure of TCA.

4.1. Algorithm Overview. TCA works for generating a caching vector X in a greedy manner such that it iteratively assigns a video quality version to an available cache server, which leads to the maximal performance gain among all choices, until no file can be placed.

TCA works as follows. First, it initializes an all-zero caching vector X with $x_{nv} = 0, \forall n, v$. Then, it iteratively updates X by placing a video version on a cache server, one video version placed each time. Given a set of video versions to be considered for caching at a set of cache servers, there typically exist multiple placement choices in each iteration. For each possible placement (n, v, q) (i.e., placing video (v, q) on server n), we compute the delivery delay for video v (called after-placement delay) by using (2) or (5), based on whether independent caching or cooperative caching is used, under the current X. Moreover, we have a delivery delay for video v before the placement (called base delay). The reduced delay (i.e., the base delay minus the after-placement delay) is called the delay gain associated with the placement. Among all possible placement choices, TCA chooses the choice leading to the maximal delay gain and accordingly updates the caching vector. This process is repeated until no more file can be cached at any of the cache servers or no more video version needs to be considered for caching.

4.2. Algorithm Design. The procedure of TCA is shown in Algorithm 1, which is described partially based on the code framework in [5].

First, a series of variables are initialized in lines 1–4, including (i) $E = \{e_{nvq} \mid n \in \mathcal{N}, v \in \mathcal{V}, q \in \mathcal{Q}\}$ being initialized as a set containing all possible placement choices, where each element $e_{nvq} \in E$ represents a possible placement, i.e., it is still applicable to place (v, q) at server n, (ii) the possible placements $E_n \subseteq E$ for each cache server n, (iii) an all-zero caching vector X, and (iv) an all-zero caching vector X_n for each cache server n.

Then, it iteratively updates X (see line 5 to line 24 in Algorithm 1). In each iteration, line 6 finds the best placement $e_{\alpha\beta\gamma}$ (i.e., placing the γ version of video β at server α) among all possible placements in E, where function $J_X(g)$ computes the delay gain of a possible placement g based on the current caching vector X. Following that, if the maximal delay gain equals zero, TCA will break the loop; otherwise, it will continue the following operations. Line 10 adjusts cache size C_α, by removing the space to be occupied by the new video version (β, γ) and also retrieving the space occupied by the previously cached video version $(\beta, x_{\alpha\beta})$ (if any). Lines 11 and 12 update $x_{\alpha\beta} = \gamma$ in X and X_α, respectively, according to the new choice of $e_{\alpha\beta\gamma}$; lines 13 and 14 removes placements $e_{nvq}, \forall n = \alpha, v = \beta, q \leq r$, from E and E_α, since all the user requests for fetching $(\beta, q), q \leq r$ from server α can be satisfied by this new placement and thus it no longer needs to cache any lower version of the video α. Next, lines 15–20 remove those placements, which are impossible to be realized due to limited remaining cache space, from E and E_n. Following that, if E becomes empty (see line 21), TCA will break the loop. Finally, TCA outputs the final caching vector X.

The computational complexity of TCA can be easily deduced as follows. The initialization in lines 1–4 obviously takes $O(\mathcal{N}\mathcal{V}\mathcal{Q})$ time. In the "for" loop between line 5 and line 24, line 6 has the highest complexity of $O(\mathcal{N}\mathcal{V}\mathcal{Q})$ and the "for" loop takes at most $O(\mathcal{N}\mathcal{V}\mathcal{Q})$ time. Thus, the computational complexity of TCA is $O(\mathcal{N}^2\mathcal{V}^2\mathcal{Q}^2)$.

5. Performance Evaluation

In this section, we perform simulations to evaluate the performance of TCA. We compare three instances of two caching algorithms, including two instances of TCA, which adopt independent caching (TCA-I) and cooperative caching (TCA-C), respectively, and the traditional greedy algorithm (Greedy) for cooperative caching [5].

5.1. Parameter Settings. We set $N = 3$ cache servers for simulations, each having the same cache size. The default rate of transmitting videos between each of the servers and the remote source server is 2 Mbps, and the default rate (i.e., cooperative transmit rate) among the cache servers for video delivery in the mode of cooperative caching was set to 5 Mbps. The default cache size at a cache server is 400 GBytes. We assume there are 1000 videos for caching, all of which have one-hour playback duration. According to the Youtube recommended resolutions and bitrates, each video supports 5 different quality levels of video resolutions, including 240p, 360p, 480p, 720p, and 1080p, which correspond to video bitrates of 400, 750, 1000, 2500, and 4500kbps, respectively

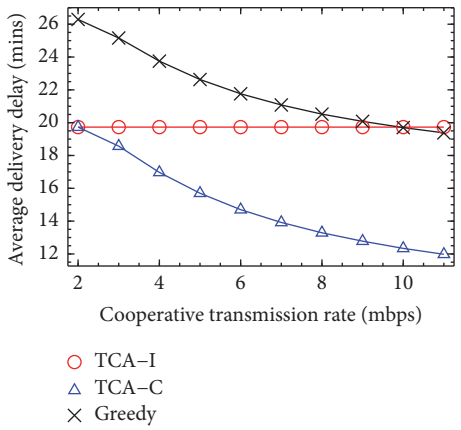


Figure 2: Impact of cooperative transmit rate.

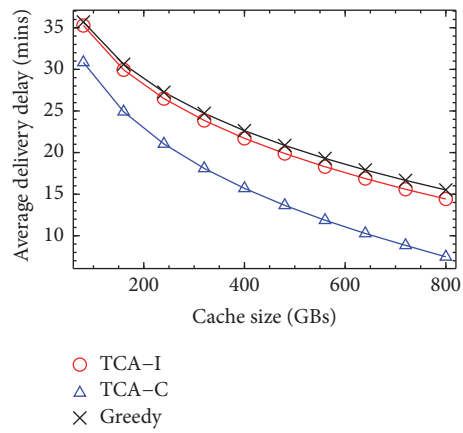


Figure 3: Impact of cache size.

[21]. Thus, the total size of the 5000 video versions is close to 4 TB.

Following typical settings used for empirical studies in VoD system [9, 22], the popularity of the 1000 videos is assumed to follow a Zipf distribution; i.e., the i^{th} most popular video will be requested at a rate in proportion to i^{-z}, where z is a shape parameter and was set to 0.8 unless otherwise specified. Furthermore, different versions of a video are assumed to be equally requested by users.

5.2. Simulation Results. Impact of cooperative transmit rate on delivery delay. For this test, we conducted a series of simulations with varying cooperative transmit rate ranging from 2 to 11 Mbps at 1 Mbps granularity. The results are shown in Figure 2, where the x-axis is the cooperative transmit rate and the y-axis is the average video transmission delay. The three curves correspond to TCA-C, Greedy, and TCA-I, respectively. The former two algorithms adopt cooperative caching and can reduce the delivery delay by increasing the cooperative transmit rate. The third algorithm uses independent local caching and thus has a constant performance irrelevant to cooperative transmit rate. The results in this figure clearly demonstrate that TCA-C has the best performance, since it makes full use of both transcoding and cooperative caching at different cache servers. The gains in terms of delivery delay are up to 40% and 38% as compared to TCA-I and Greedy, respectively.

Impact of cache size on delivery delay. In this test, we conduct a series of simulations with varying cache size ranging from 50 GB to 800 GB at 100 GB granularity. Figure 3 shows the results. In Figure 3, all the three curves show similar decreasing trends on average delivery delay with cache space increasing. Moreover, for each given cache size, the results show that TCA-C significantly outperforms TCA-I and Greedy, with a decrease of average delivery delay by up to 50% and 53%, respectively. The results indicate that TCA-C can efficiently utilize available cache space, so as to improve user experience of video fetching.

Impact of video popularity distribution on delivery delay. In this test, we conduct a series of simulations with varying shape parameter Zipf ranging from 0.4 to 1.2 at 0.1 granularity.

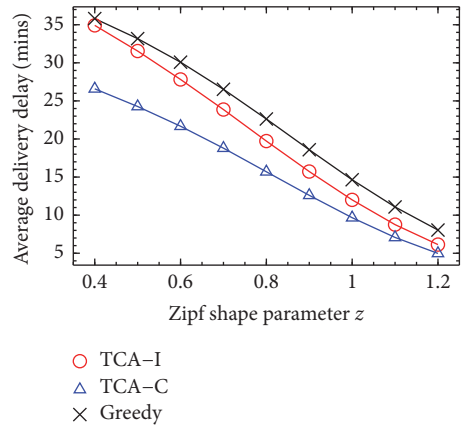


Figure 4: Impact of Zipf shape parameter.

Figure 4 shows the results. Not surprisingly, all the simulated algorithms lead to decrease on delivery delay with increasing of the shape parameter z. This is because a larger z indicates fewer videos are more popular such that the caching of them can satisfy a large proportion of all users' requests. Figure 4 shows that TCA-C always has the best performance.

6. Conclusion

In this paper, we studied transcoding based caching for improving the performance of a distributed video caching system. We formulated optimal transcoding based video caching problem under two different modes (i.e., independent caching and cooperative caching) as integer linear programming problem. We then proposed a transcoding based caching algorithm TCA, which iteratively places a video version to a cache server, which leads to the maximal delay gain among all possible choices. Simulation results demonstrate that TCA (when working in cooperative caching mode) can significantly reduce the delivery delay compared with traditional greedy 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 work is supported in part by the National Science Foundation of China under Grants nos. 61572071, U1534201, 61531006, and 61471339, the Natural Sciences and Engineering Research Council (NSERC) of Canada (Discovery Grant RGPIN-2018-03792), and the InnovateNL SensorTECH Grant 5404-2061-101.

References

[1] Q. Zhang, Z. Xiang, W. Zhu, and L. Gao, "Cost-based cache replacement and server selection for multimedia proxy across wireless Internet," *IEEE Transactions on Multimedia*, vol. 6, no. 4, pp. 587–598, 2004.

[2] H. Chen and Y. Xiao, "Cache access and replacement for future wireless internet," *IEEE Communications Magazine*, vol. 44, no. 5, pp. 113–123, 2006.

[3] R. L. Mattson, J. Gecsei, D. R. Slutz, and I. L. Traiger, "Evaluation techniques for storage hierarchies," *IBM Systems Journal*, vol. 9, no. 2, pp. 78–117, 1970.

[4] A. V. Aho, P. J. Denning, and J. D. Ullman, "Principles of optimal page replacement," *Journal of the ACM*, vol. 18, pp. 80–93, 1971.

[5] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "FemtoCaching: wireless video content delivery through distributed caching helpers," in *Proceedings of the IEEE INFOCOM 2012*, vol. 59, pp. 1107–1115, Orlando, FL, USA, 2012.

[6] L. E. Chatzieleftheriou, M. Karaliopoulos, and I. Koutsopoulos, "Caching-aware recommendations: Nudging user preferences towards better caching performance," in *Proceedings of the 2017 IEEE Conference on Computer Communications, INFOCOM 2017*, pp. 1–9, Atlanta, GA, USA, May 2017.

[7] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, 2007.

[8] A. Vetro, C. Christopoulos, and H. Sun, "Video transcoding architectures and techniques: An overview," *IEEE Signal Processing Magazine*, vol. 20, no. 2, pp. 18–29, 2003.

[9] K. Poularakis, G. Iosifidis, A. Argyriou, I. Koutsopoulos, and L. Tassiulas, "Caching and operator cooperation policies for layered video content delivery," in *Proceedings of the 35th Annual IEEE International Conference on Computer Communications, IEEE INFOCOM 2016*, pp. 1–9, San Francisco, CA, USA, April 2016.

[10] B. Shen, S.-J. Lee, and S. Basu, "Caching Strategies in Transcoding-Enabled Proxy Systems for Streaming Media Distribution Networks," *IEEE Transactions on Multimedia*, vol. 6, no. 2, pp. 375–386, 2009.

[11] G. Zhang, *Computational complexity optimization on H.264 scalable/multiview video coding [Ph.D. thesis]*, University of Central Lancashire, 2014.

[12] B. Shen and S. Roy, "A very fast video spatial resolution reduction transcoder," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 1989–1992, Orlando, FL, USA, 2012.

[13] S. E. Elayoubi and J. Roberts, "Performance Evaluation of Video Transcoding and Caching Solutions in Mobile Networks," in *Proceedings of the 27th International Teletraffic Congress, ITC 2015*, pp. 55–63, Ghent, Belgium, September 2015.

[14] T. X. Tran, P. Pandey, A. Hajisami, and D. Pompili, "Collaborative multi-bitrate video caching and processing in Mobile-Edge Computing networks," in *Proceedings of the 2017 13th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*, pp. 165–172, Jackson Hole, WY, USA, Feburary 2017.

[15] G. Gao, W. Zhang, Y. Wen, Z. Wang, and W. Zhu, "Towards Cost-Efficient Video Transcoding in Media Cloud: Insights Learned from User Viewing Patterns," *IEEE Transactions on Multimedia*, vol. 17, no. 8, pp. 1286–1296, 2015.

[16] X. Li, M. A. Salehi, M. Bayoumi, N. Tzeng, and R. Buyya, "Cost-Efficient and Robust On-Demand Video Transcoding Using Heterogeneous Cloud Services," *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 3, pp. 556–571, 2018.

[17] A. Albanese, P. S. Crosta, C. Meani, and P. Paglierani, "GPU-accelerated video transcoding unit for multi-access edge computing scenario," in *Proceedings of the ICN 2017*, pp. 143–147, Venice, Italy, 2017.

[18] S. Dutta, T. Taleb, P. A. Frangoudis, and A. Ksentini, "On-the-fly QoE-aware transcoding in the mobile edge," in *Proceedings of the 59th IEEE Global Communications Conference, GLOBECOM 2016*, pp. 1–6, Washington, DC, USA, 2016.

[19] K. Fung and W. Siu, "Low complexity H.263 to H.264 video transcoding using motion vector decomposition," in *Proceedings of the 2005 IEEE International Symposium on Circuits and Systems*, pp. 908–911, Kobe, Japan, 2005.

[20] A. J. Daz-Honrubia, G. Cebrin-Mrquez, J. L. Martnez, P. Cuenca, and J. M. Puerta, "Low-complexity heterogeneous architecture for H.264/HEVC video transcoding," *Journal of Real-Time Image Processing*, vol. 12, no. 2, pp. 311–327, 2016.

[21] Live encoder settings, bitrates, and resolutions, https://support.google.com/youtube/answer/2853702?hl=en, 2018.

[22] M. Hefeeda and O. Saleh, "Traffic modeling and proportional partial caching for peer-to-peer systems," *IEEE/ACM Transactions on Networking*, vol. 16, no. 6, pp. 1447–1460, 2008.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 2102348, 11 pages
https://doi.org/10.1155/2018/2102348

Research Article

A Task Scheduling Algorithm Based on Classification Mining in Fog Computing Environment

Lindong Liu,[1,2] **Deyu Qi**,[1] **Naqin Zhou**,[3] **and Yilin Wu**[2]

[1]*Research Institute of Computer Systems, South China University of Technology, Guangzhou, China*
[2]*Department of Computer Science, Guangdong University of Education, Guangzhou, China*
[3]*Cyberspace Institute of Advanced technology, Guangzhou University, Guangzhou, China*

Correspondence should be addressed to Lindong Liu; hongox@163.com

Received 28 April 2018; Revised 18 June 2018; Accepted 5 July 2018; Published 1 August 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Lindong Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Fog computing (FC) is an emerging paradigm that extends computation, communication, and storage facilities towards the edge of a network. In this heterogeneous and distributed environment, resource allocation is very important. Hence, scheduling will be a challenge to increase productivity and allocate resources appropriately to the tasks. We schedule tasks in fog computing devices based on classification data mining technique. A key contribution is that a novel classification mining algorithm I-Apriori is proposed based on the Apriori algorithm. Another contribution is that we propose a novel task scheduling model and a TSFC (Task Scheduling in Fog Computing) algorithm based on the I-Apriori algorithm. Association rules generated by the I-Apriori algorithm are combined with the minimum completion time of every task in the task set. Furthermore, the task with the minimum completion time is selected to be executed at the fog node with the minimum completion time. We finally evaluate the performance of I-Apriori and TSFC algorithm through experimental simulations. The experimental results show that TSFC algorithm has better performance on reducing the total execution time of tasks and average waiting time.

1. Introduction

Many applications, such as health monitoring application or intelligent traffic control application may need to receive feedback in a short amount of time, and the latency due to sending data to the cloud and then returning the response from the cloud to the operator of these programs has bad effects [1]. So, in 2012, Bonomi presented a novel concept called the fog computing [2]. Fog computing consists of a large number of geographically distributed fog servers which can be cellular base stations, access points, gateways, switches, and routers with limited capabilities, as compared to specialized computing facilities such as data centers [3–5]. In fog computing, the massive data generated by different kinds of Internet of Things (IoT) [6, 7] devices can be processed at the network edge instead of transmitting it to the centralized cloud infrastructure due to bandwidth and energy consumption concerns [8]. Fog computing has become a new computing model in providing local computing resources and storage for end-users rather than cloud computing.

The contradiction [9, 10] between computation intensive applications and resource limited devices becomes the bottleneck for providing satisfactory quality of experience. This contradiction needs to be solved by task scheduling in fog computing environment. Task scheduling is widely applied in distributed computing systems and the cloud computing environment [11, 12]. Task scheduling in fog computing is to allocate appropriate resources for application tasks. How to select appropriate resources for the application task to meet the minimum completion time, to satisfy the users' quality of service (QoS) requirements, to improve the fog computing throughput, and to achieve the load balancing scheduling can be defined as task scheduling problem in fog computing environment. Therefore, it is of great practical significance to achieve efficient resource utilization and higher performance in the fog computing environment.

In fog computing environment, task scheduling depends on whether there are dependencies between the tasks that are scheduled. It can be divided into independent task scheduling

and related task scheduling. Related task scheduling is often referred to as dependent task scheduling [13]. There is no dependency relationship and data communication among tasks in independent task scheduling [14, 15]. Dependent task scheduling has some dependence and there is data communication among tasks. A typical task scheduling model is built on the basis of graphs, usually called task graphs. The most common task graph is Directed Acyclic Graph (DAG), so the dependent task scheduling is also called DAG scheduling.

Before tasks are scheduled, tasks have two ways to arrive. One is the batch mode. When all tasks arrive, they are allocated to the corresponding fog nodes through a scheduling algorithm. Another is the online mode. The arrival time of each task is random and a task is scheduled to a fog node as soon as it arrives at the RMS (resource management system). Task scheduling of fog nodes has been proved to be a NP-complete problem [16]. The research work of task scheduling is a very important aspect and has been widely and deeply studied by researchers [17]. At present, although many research achievements have been obtained for task scheduling, researchers are still continuing to explore and study [18]. Research of scheduling tasks in fog computing environment has not been well-established yet due to the lack of fog architecture that manages and allocates resources efficiently. Our research also has a positive influence on some optimization problems [19–22].

The rest of the paper is organized as follows. In Section 2 we describe the related work of the research. In Section 3, we introduce the classification mining algorithm and an improved I-Apriori algorithm. In Section 4, we introduce a task scheduling model, the scheduling algorithm, and the scheduling process in fog computing. The analysis of the experimental process and experimental results of task scheduling algorithm are given in Section 5, followed by our conclusion made in Section 6.

2. Related Work

2.1. Related Work of Classification Mining. Classification mining algorithms are widely used in text, image, video, traffic, medical, big data, and other application scenarios. A pipelined architecture for the implementation of axis parallel binary DTC was proposed in [23] that dramatically improves the execution time of the algorithm while consuming minimal resources in terms of area. Reference [24] proposed a fast and accurate data classification approach which can learn classification rules from a possibly small set of records that are already classified. The proposed approach is based on the framework of the so-called Logical Analysis of Data (LAD). The accuracy and stability of the proposed algorithm are better than that of the standard LAD algorithm. Sequence classification was introduced in [25] using rules composed of interesting patterns found in a dataset of labelled sequences and accompanying class labels. They measure the interestingness of a pattern in a given class of sequences by combining the cohesion and the support of the pattern. They use the discovered patterns to generate confident classification rules and present two different ways of building a classifier. The patterns that the algorithm discovers represent the sequences well and are proved to be more effective for the classification tasks than other machine learning algorithms. A Bayesian classification approach for automatic text categorization using class-specific features was proposed in [26]. Unlike conventional text categorization approaches, the method selects a specific feature subset for each class. One noticeable significance of the algorithm is that most feature selection criteria such as Information Gain (IG) and Maximum Discrimination (MD) can be easily incorporated into the algorithm. Compared with other algorithms, it demonstrates that the algorithm is effective and further indicates its wide potential applications in data mining. Furthermore, we will apply this algorithm to other areas, such as oblivious RAM [27, 28], string mapping [29], and match problem [30].

2.2. Related Work of Independent Task Scheduling. For a large scale environment, e.g., cloud computing system, there had been also numerous scheduling approaches proposed with the goal of achieving the better task execution time for cloud resources [31]. Independent task scheduling algorithms mainly include MCT algorithm [32], MET algorithm [32], MIN-MIN algorithm [33], MAX-MIN algorithm [33], PMM algorithm, and genetic algorithm. The MCT (Minimum Completion Time) algorithm assigns each task in any order to the processor core that causes the task to be finished at the earliest time. It makes some tasks unable to be allocated to the fastest processor core. The MET (Minimum Execution Time) algorithm assigns each task to a processor core in any order that minimizes the execution time of the task. Contrary to the MCT algorithm, the MET algorithm does not consider the processor core's ready time, which may lead to serious load imbalance across processor cores. The MIN-MIN algorithm calculates the minimum completion time of all unscheduled tasks firstly, and then selects the task with the minimum completion time and assigns the task to the processor core that can minimize its completion time, repeating the process many times until all tasks are scheduled. The same as the MCT algorithm, the MIN-MIN algorithm is also based on the minimum completion time. The MIN-MIN algorithm considers all tasks that are not scheduled, but the MCT algorithm considers only one task at a time. The MAX-MIN algorithm is similar to the MIN-MIN algorithm, which also calculates minimum completion time without scheduled tasks firstly and then selects the task with the largest minimum completion time and assigns the task to the processor core with the minimum completion time. The PMM (Priority MIN-MIN) algorithm is an improvement of the MIN-MIN algorithm. It does not choose the smallest task with the earliest complete time, but it selects k tasks with smaller earliest completion time and schedules the task with highest priority in the k tasks. The PMM algorithm takes the standard deviation of the task on each processor core as the priority of the task. The higher the standard deviation, the higher the task priority.

On one hand, literature of existing classification algorithms applies decision tree algorithm and Bayes classification algorithm to various application scenarios. On the other hand, combined with cloud computing, distributed computing, big data, grammatical evolution [34, 35], and other

technologies, researchers are focused on how to optimize and improve the performance of classification algorithms. In task scheduling, few researchers apply the classification mining algorithm to schedule tasks.

3. Classification Data Mining

3.1. Overview. Classification mining algorithm [36] is the key technology of data mining. As a supervised learning algorithm, it is based on existing training data sets to set up a model to predict the categories of new data sets. It can find classification rules and predict new data types through analysis of the training data set. A classification mining algorithm consists of two stages which are building the model phase and using the model phase. In the first stage, it analyzes the existing training data set and builds a corresponding model and then generates some classification rules. In the second stage, it classifies new data sets based on the constructed classification model.

Major classification mining algorithms include random decision forests [37], decision tree algorithm, Bayes algorithm, genetic algorithm, artificial neural network algorithm [34], and classification algorithm based on association rules. Classification algorithm is widely used in wireless sensor networks, network intrusion detection, call logs, and risk assessment in banks. In this paper, the classification algorithm based on association rules is introduced and the Apriori algorithm is improved and evaluated.

3.2. Mining Model. Apriori [35, 38, 39] is a classical classification algorithm based on association rules (CBA). It generates frequent itemsets through an iterative process. The Apriori algorithm includes two steps. First of all, it finds frequent itemsets from a known transaction in which the frequency is greater than or equal to minimum support threshold through pruning and connection operation of frequent itemsets. Then, it generates association rules based on the frequent itemsets and minimum confidence degree.

The improved association rule mining model is implemented in two steps. (1) Firstly, the transaction database D is scanned to store the transaction identification *TID* for each itemset, and the candidate 1-itemset C_1 is generated. Delete the itemsets from C_1 which are less than the minimum support threshold, and get the frequent 1-itemsets of L_1. (2) Loop execution of the process is done until $L_{k\text{-}1}$ is empty. Firstly, let $L_{k\text{-}1}$ and $L_{k\text{-}1}$ be joined to generate candidate itemset C_k. Secondly, a new transaction identifier list can be obtained through the intersection of the transaction identifier list, and the count of the itemsets can be obtained directly through C_k. Thirdly, comparing the count of C_k with the minimum support threshold *min_sup*, reserve itemsets which are more than or equal to minimum support threshold *min_sup*, and delete the rest of itemsets; then the final frequent itemset L is generated.

3.3. Improved Association Rule Mining Algorithm. In the process of producing frequent itemsets in the Apriori algorithm, there are two factors that affect the performance of the algorithm. Firstly, it needs to scan the original transaction database every time to generate the frequent k-itemsets, so the number of scanned transaction databases is too much, which can result in the decline of algorithm performance. Secondly, in the process of tree cutting, the algorithm needs to scan candidate k-1 sets to get candidate itemset. Therefore, the algorithm scans itemsets many times; it also leads to the decline of algorithm performance. In view of the above problems, we improve the process of frequent itemsets in the algorithm, and an improved I-Apriori algorithm is proposed based on the Apriori algorithm. The I-Apriori algorithm is described as follows in Algorithm 1.

```
1 Input: transaction database D; min_sup
2 Output: frequent itemsets L
3 C1=find_candidate_1-itemsets(D);
4 int count=the number of TID in D;
5 for each itemset s of C1{
6    s.item-set=s;
7    s.count=count of s in C1;
8 s.tid-list=the set of all TID includes s;
9 if s.count<min_sup*count
10    delete s in C1;
11 }
12 L1=C1;
13 for (k=2; Lk-1 ≠ ∅; k++){
14    for each itemset l1 in Lk-1{
15       for each itemset l2 in Lk-1{
16          c=l1⋈l2;
17          c.tid-list= l1.tid-list∧l2.tid-list;
18          c.count=count TID in c.tid-list;
19       }
20    }
21 if c.count>=min_sup*count
22    add c to Ck;
23    Lk=Ck;
24 }
```

Algorithm 1: I-Apriori algorithm.

In the I-Apriori algorithm, during the process of generating the candidate itemset C_k every time, except for storing the itemset and the count of support degree, it is more important to store the transaction identifier list attribute *Tid-list*. After completing the connection operation between itemsets, the algorithm can get the list of transaction identifiers and the count of itemsets directly through the attribute *Tid-list* and does not need to scan the transaction database again. Based on the above reasons, I-Apriori algorithm can improve the performance effectively.

3.4. Algorithm Evaluation. The efficiency of Apriori algorithm and I-Apriori algorithm is evaluated based on time complexity and algorithm execution time.

3.4.1. Time Complexity. Suppose the number of transactions and items in the transaction database D is n and m, and the iteration times of frequent itemsets in the algorithm is k. The time complexity of classical Apriori algorithm is composed of three layers nested for loops, *apriori_gen* subroutine and

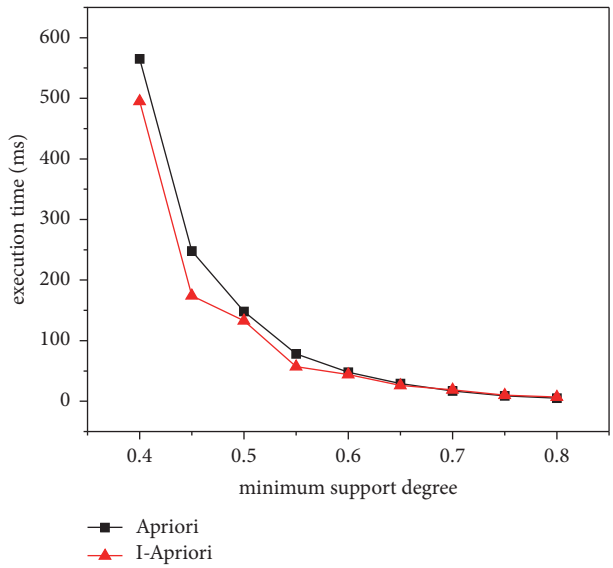


Figure 1: Comparison of execution time.

has_infrequent_subset subroutine called *apriori_gen* subroutine in the main algorithm. It is easy to find that the time complexity of Apriori algorithm is $O(k^4 * m*n)$. According to the I-Apriori algorithm shown in Algorithm 1, because only one time is needed to scan the transaction database D, the time complexity of I-Apriori algorithm is $O(m+n+k^3)$. Obviously, $O(m+n+k^3)$ is better than $O(k^4 * m*n)$. The greater the transaction database D, the more the number of items, the more iterations, and the higher efficiency of the I-Apriori algorithm.

3.4.2. Experimental Analysis. The Java language is used to realize the classic Apriori algorithm and the I-Apriori algorithm, respectively. The hardware environment is Intel 2.5 GHz CPU, 4 GB memory, and the operation system is Windows 7. We generated corresponding frequent itemsets for the transaction database.

When the number of transactions in the transaction database is 200 and the number of items is 20, the execution time needed for the two algorithms to generate frequent itemsets under different minimum support degree (0.4~0.8) is shown in Figure 1. When the number of items in the transaction database is 20 and the minimum support degree is 0.4 and 0.6 (several experiments show that the execution time of the algorithms is longer when the minimum support degree is 0.4, while the algorithm has a shorter execution time when the minimum support degree is 0.6; therefore, 0.4 and 0.6 are chosen to compare the execution time of the two algorithms under different transaction numbers), the execution time needed for the two algorithms to generate frequent itemsets under different number of transactions (50~400) is shown in Figure 2.

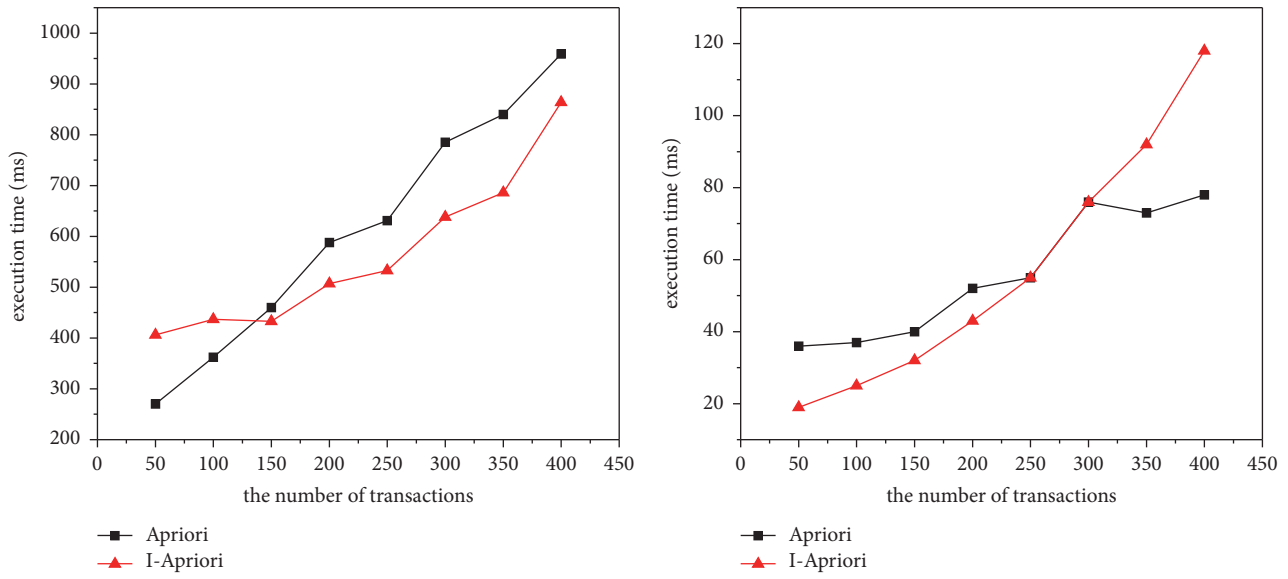
From Figure 1, when the minimum support degree of Apriori algorithm and I-Apriori algorithm is small, the execution time of generating frequent itemsets of I-Apriori algorithm is smaller than that of Apriori algorithm. With the increase of minimum support degree, there is little difference in execution time of the two algorithms. When the minimum support degree is large, the execution time of generating frequent itemsets of I-Apriori algorithm is larger than that of Apriori algorithm. When the minimum support degree is small and the number of iterations is greater, the efficiency of the I-Apriori algorithm is higher. When the minimum support degree is large and the number of iterations is smaller, the efficiency of the Apriori algorithm is higher. Therefore, the I-Apriori algorithm is suitable for smaller minimum support degree and more iterations in classification mining. When the minimum support degree is small, the number of iterations of classification mining will increase. The I-Apriori algorithm will reduce the times of scanning the transaction database significantly, and the execution time of the algorithm is shorter. On the contrary, when the minimum support degree is large, the number of iterations will be decreased. Although the I-Apriori algorithm also can reduce the times of scanning the transaction database, I-Apriori algorithm has no advantage over Apriori algorithm.

In the case of smaller minimum support degree in Figure 3, when the number of transactions is smaller, the execution time of generating frequent itemsets of the Apriori algorithm is smaller than that of the I-Apriori algorithm. With the increase of the number of transactions, the efficiency of I-Apriori algorithm is obviously higher than that of Apriori algorithm. In the case of larger minimum support degree in Figure 4, the execution time of generating frequent itemsets of the Apriori algorithm is larger than that of the I-Apriori algorithm when the number of transactions is small. With the increase of the number of transactions, the Apriori algorithm is more efficient than the I-Apriori algorithm. Generally speaking, the I-Apriori algorithm is suitable for small minimum support degree and large number of transactions when generating frequent itemsets.

4. Task Scheduling of Fog Computing

Task scheduling of fog computing is to schedule tasks to fog nodes with different computing powers, and arrange their execution order reasonably, so that the total execution time is shortest. All notations utilized in the paper are listed in Table 1.

4.1. Fog Computing System Architecture. Fog computing system [40] has three tiers in a hierarchy network, as represented in Figure 3. The front-end tier consists of IoT devices, which serve as user interfaces that send requests from users via WiFi access points or Internet. IoT devices are always subject to strict constraints on their resource such as CPU, memory, and, when run, a very complex application. The fog tier, which is formed by a set of near-end fog nodes, receives and processes part of a workload of users' request. The fog tier is generally deployed near IOT terminals, which provides limited computing resources for users. Users can access the computing resources in the fog tier directly, so it can avoid additional communication delays. The cloud tier consists of multiple servers or cloud nodes. The remote cloud can provide abundant computing resources, but it is located



(a) Execution time under the minimum support degree 0.4

(b) Execution time under the minimum support degree 0.6

FIGURE 2: (a) Comparison of execution time under the minimum support degree 0.4. (b) Comparison of execution time under the minimum support degree 0.6.

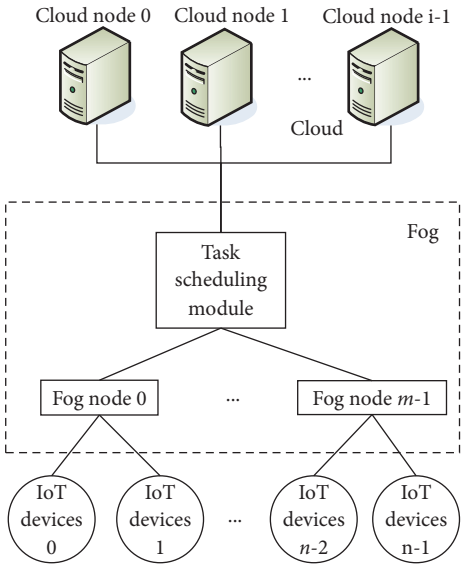


FIGURE 3: Fog computing system architecture.

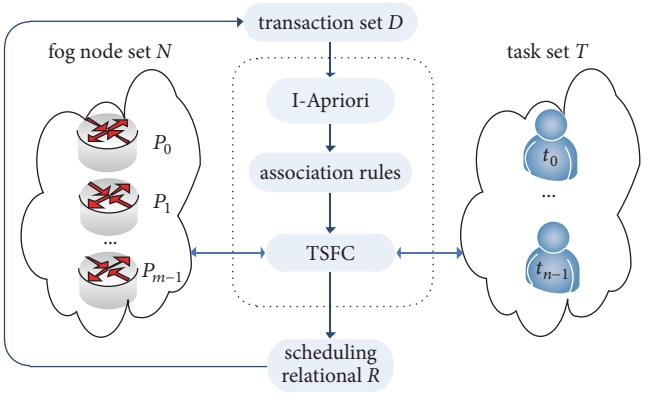


FIGURE 4: Task scheduling model of fog computing.

physically far from the users and the transmission delay is large.

4.2. Task Scheduling Model. In order to implement the task scheduling of fog computing effectively, the classification algorithm is integrated into the task scheduling process of fog computing. Figure 4 presents the task scheduling model of fog computing. In order to realize an effective scheduling process between the fog node set N and the task set T, the scheduling module consists of two algorithms, i.e., I-Apriori algorithm and TSFC (Task Scheduling in Fog Computing) algorithm. Firstly, based on the scheduling transaction set D, association rules of the node set and the task set are generated by the I-Apriori algorithm. Secondly, the association rules are used as the input of TSFC algorithm to get the task scheduling relationship between the fog node set and the task set. Finally, the task scheduling relationship R is inserted into the scheduling transaction set D to provide input data for the next task scheduling.

4.3. TSFC Scheduling Algorithm. Based on the I-Apriori algorithm, TSFC algorithm is designed and is shown in Algorithm 2. The basic idea of the algorithm is to schedule tasks in the task scheduling relational table with higher priority. Set the completion time of these tasks in the table to a larger value, and then select the fog node with the minimum completion time. Execute a loop from the rest of

Table 1: Summary of the notations.

var	definition
$TS(k)$	TS contains k task sets that needs to be scheduled
T	$T=\{t_1,t_2,\cdots,t_n\}$ is a set of n tasks. In this set, all of the tasks in T are independent tasks
N	$N=\{P_0, P_1, \ldots, P_{m-1}\}$ denotes the set of processors
D	An edge $d_{ij} \in D$ denotes a link between processor P_i and P_j
bw_{ij}	the bandwidth between processor P_i and P_j
P	$P=(N,D)$ denotes the topology of a fog computing network
$Time[n,m]$	$Time[i,j]$ descripts the estimated running time of the task t_i on fog node P_j
$D[z]$	Scheduling transaction set contains z transactions
$R[t,c]$	Task scheduling relationship contains the scheduling relationship between the task set T and the fog node N
ST_i	The start execution time of task t_i
AT_i	Actual arrival time of task t_i
FT_i	the completion time of the task t_i
TST	Total task scheduling execution time is the maximum value of all tasks' completion time
AWT	Average waiting time is TST divided by n, $AWT=\sum_{j=1}^{n}(ST_j - AT_j)/n$

```
1 Input: Time[n, m], R[t, c], TS
2 Output: TST, AWT
3 double ST[i], FT[i], min_FT, Total_Time, Time, TST, AWT, WT;
4 int Total_node, Total_Task, Task, Total_TaskSet, best_node_ID;
5 read every taskset from TS;
6 for (i=0; i<Total_TaskSet; i++){
7    read a task from taskset;
8    for (j=0; j<Task; j++){
9       for (k=0; k<Total_node; k++){
10           if (R[j,k]=-1)
11             FT[j]=Time[j,k];
12         else
13             FT[j]=ST[j]-R[j,k];
14       }
15    find minimal min_FT and corresponding k in FT[j] with task j;
16    ST[k]=ST[k]+min_FT;
17    sort all tasks in taskset i from small to large according to FT[j];
18    ST[ ]=0; FT[ ]=0;
19    }
20   while (taskset i is not empty) {
21    select task t_Task with the largest FT[j] in taskset;
22    if t_Task=null
23       break;
24    select the node with smallest FT[j] of t_Task and return best_node_ID;
25    WT=WT+ST[best_node_ID];
26    ST[best_node_ID]=FT[best_nodee_ID];
27    delete t_Task;
28    Task=Task-1;
29    recalculate every FT[j] of the rest of tasks;
30    sort all tasks in taskset i from small to large according to FT[j];
31    }
32   write the largest ST[j] to Time of all tasks;
33   Total_Time=Total_Time+Time;
34   Total_Task=Total_Task+Task;
35   TST=Total_Time;
36   AWT=WT/Total_Task;
37 }
```

Algorithm 2: TSFC algorithm.

Table 2: Execution time matrix Time$[n,m]$ of task set T and fog node set N.

task	P_0	P_1	P_2	P_3
t_0	200	211	180	223
t_1	102	122	91	130
t_2	81	92	88	95
t_3	55	59	57	61
t_4	32	33	29	36
t_5	155	160	149	173
t_6	287	291	267	305
t_7	135	142	122	160
t_8	228	237	204	251
t_9	178	183	161	195

the tasks to select the task with minimum completion time to schedule and assign the selected task to the fog node with minimum completion time until all of the tasks are scheduled. Supposing the number of task sets, tasks, and fog nodes is k, n, and m, respectively, the time complexity of TSFC algorithm is $O(k*n^2+k*n*m)$.

4.4. Analysis of Scheduling Process. In order to understand and analyze the TSFC algorithm, a complete case is used to analyze the scheduling process of the TSFC algorithm. We analyze the whole process of task scheduling algorithm of fog computing. Suppose that the task set T contains 10 tasks and the node set C includes 4 fog nodes; that is, n=10 and m=4. The execution time matrix $Time[n,m]$ of task set T and node set C is shown in Table 2.

(1) Transaction database. Transaction set $D[z]$ is shown in Table 3. Each scheduling information between the task set T and the fog node set N is stored as a transaction information. A Boolean value is used to describe whether the task or node is scheduled or not. The Boolean true value representing the task or node is scheduled. On the contrary, the Boolean false value representing the task or node is not scheduled. In addition, it is assumed that the transaction set D contains 10 transactions; that is, z=10.

(2) Classification mining. The transaction database D is used as the input of I-Apriori algorithm, and the minimum support degree min_sup=0.5. Frequent itemsets $\{P_1,P_0,P_3,t_7\}$ and $\{P_2,P_0,P_3,t_7\}$ are generated by the I-Apriori algorithm. Association rules $t_7 => P_1 \vee P_0 \vee P_3$ (minimum confidence degree is equal to 0.833) and $t_7 => P_2 \vee P_0 \vee P_3$ (confidence degree is equal to 1.0) are generated with the minimum confidence degree min_conf=0.8.

(3) Task scheduling relational table. According to the association rules generated by the I-Apriori algorithm, the scheduling relationship between the task set and fog node set is shown in Table 4. In the task scheduling relational table $R[t,N]$, there are three kinds of values of task t_i corresponding to fog node P_j. In the first case, if the task t_i and the fog node P_j do not appear in the association rules, every value of the row corresponding to task t_i is equal to −1. In the second case, if task t_i and fog node P_j appear in the association rules, then calculate the confidence degree of task t_i on the fog node P_j. Let the confidence degree of tsk t_i corresponding to each fog node P_j be $tP_k(k\in[1,m])$, and the value of task t_i and fog node P_j is equal to $tP_k/\sum_{k=0}^{m-1} tP_k$ in the task scheduling relational table $R[t,N]$. For example, the scheduling relationship value between t_7 and P_1 is 0.833/(0.833 + 0.833 + 0.833 + 1.0 + 1.0 + 1.0)∗100=11.15. In the last case, the corresponding scheduling relationship value is equal to 0 when the fog node does not appear in the association rules.

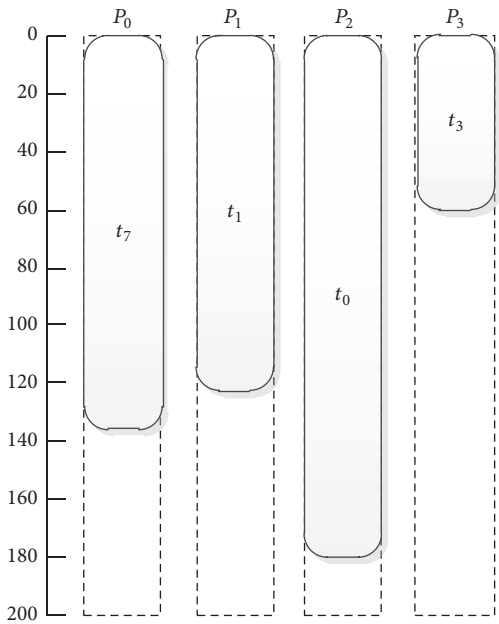


Figure 5: Scheduling diagram between tasks and fog node.

(4) Scheduled tasks TS. Scheduled tasks TS is a task list that needs to be scheduled in an experiment. Suppose the arrival time (AT_i) of all tasks is equal to 0. The task set to be scheduled is shown in Table 5.

(5) Task scheduling. Because all of the tasks are independent, the communication cost among tasks is not considered in TSFC algorithm. The value of every element of the communication matrix is equal to 0. The task set is scheduled based on the TSFC algorithm with Tables 2, 4, and 5 as input. Then, output the execution time of (TST) and the average waiting time (AWT) of the scheduled tasks.

Take the first task set$\{t_0,t_1,t_3,t_7\}$ in the scheduled tasks TS as an example. Task t_7 is scheduled to fog node firstly because the task t_7 appears in the association rules, and task t_7 is scheduled on fog node P_0 or P_3. Recalculate the minimum completion time of the three tasks $\{t_0,t_1,t_3\}$ in the task set 1, and select task t_0 with the largest minimum completion time to be scheduled on fog node P_2. Next, recalculate the minimum completion time of the remaining two tasks $\{t_1,t_3\}$ again, and task t_1 is scheduled on fog node P_1. Finally, task t_3 is scheduled on fog node P_3. The scheduling relationship between the task and fog node in the task set 1 is shown in Figure 5.

Table 3: Transaction database.

transaction	P_0	P_1	P_2	P_3	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9
T_1	T	T	T	T	T	F	T	F	F	T	F	T	F	F
T_2	F	T	F	T	F	F	F	T	F	F	F	F	T	F
T_3	T	T	T	T	F	T	T	F	T	F	F	T	F	F
T_4	T	F	T	T	F	F	F	F	F	T	F	T	F	T
T_5	F	T	F	T	F	T	T	F	F	F	F	F	F	F
T_6	F	F	F	T	F	F	F	T	F	F	F	F	F	F
T_7	T	T	F	T	F	F	F	T	F	T	F	T	F	F
T_8	F	T	T	F	T	F	F	F	F	F	F	F	T	F
T_9	T	T	T	T	F	F	F	T	T	F	F	T	F	T
T_{10}	T	T	T	T	F	T	T	F	F	F	T	F	F	T

Table 4: Relationship between task and fog node $R[t, N]$.

task	P_0	P_1	P_2	P_3
t_0	-1	-1	-1	-1
t_1	-1	-1	-1	-1
t_2	-1	-1	-1	-1
t_3	-1	-1	-1	-1
t_4	-1	-1	-1	-1
t_5	-1	-1	-1	-1
t_6	-1	-1	-1	-1
t_7	35.33	11.15	18.19	35.33
t_8	-1	-1	-1	-1
t_9	-1	-1	-1	-1

Table 5: Scheduled tasks *TS*.

No.	Task set
1	t_0,t_1,t_3,t_7
2	t_1,t_2,t_6,t_7
3	t_3,t_4,t_5,t_8,t_9
4	t_1,t_2,t_3,t_5,t_7,t_8
5	t_0,t_1,t_6,t_7,t_8,t_9

5. Simulation Experiment and Result Discussion

5.1. Experimental Purpose. In order to verify the TSFC algorithm proposed in this paper, we compare the performance of TSFC algorithm under the same experimental conditions with other three independent task scheduling algorithms, MCT, MET, and MIN-MIN.

5.2. Simulation Environment. Based on the simulator toolkit provided by SimGrid [41–43], the simulation environment for heterogeneous multiprocessors is built as follows:

(1) Internodes are interconnected through high speed networks.

(2) Each fog node can perform task execution at the same time and communicate with other fog nodes without competition.

(3) Every task is not preempted on the fog node.

(4) The fog nodes are heterogeneous.

The computer used in the experiment is configured as follows: Intel Core i5-3210M@2.5 GHz dual core processor, 8 GB memory. The number of the fog nodes in the experiment is 4 and 6, respectively.

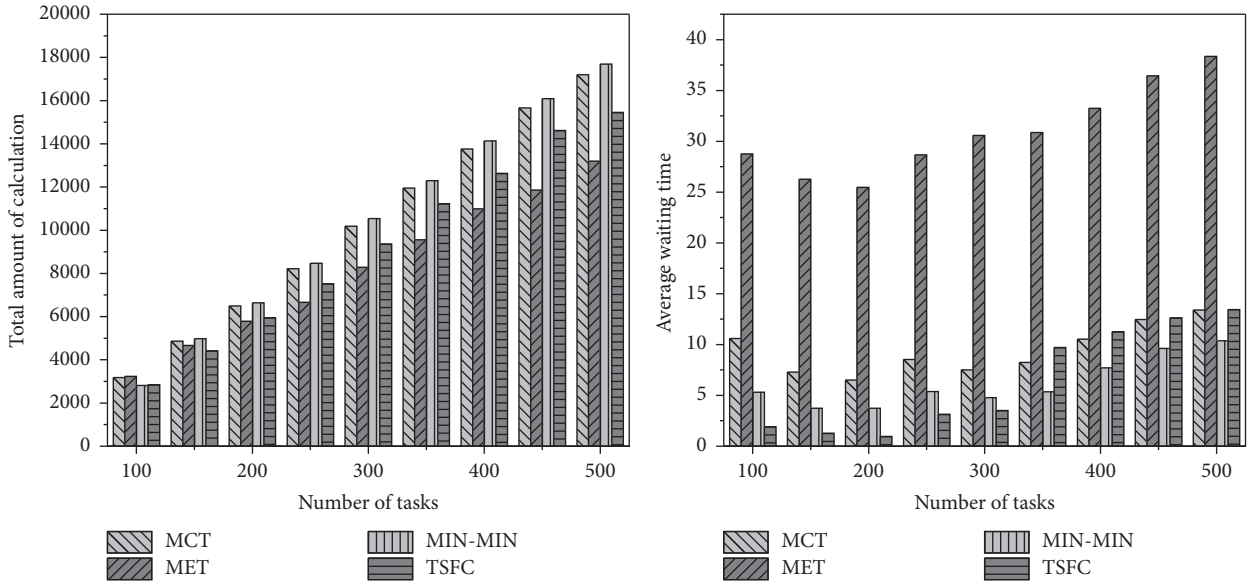
5.3. Test Data Set. The input data of TSFC algorithm include the task execution time matrix, the task scheduling relational table, and the task set. The task execution time matrix includes execution time of 10 tasks and 4 fog nodes as well as 10 tasks and 6 fog nodes. The execution time of each node is generated by a random program. The task scheduling relational table is based on the task scheduling model of fog nodes with the I-Apriori algorithm. The number of tasks in the experiment starts from 100, increasing 50 tasks each time, until the number of tasks reaches 500 tasks.

5.4. Discussion of Experimental Results

5.4.1. Result Analysis under 4 Fog Nodes. The TSFC, MCT, MET, and MIN-MIN algorithms are used to schedule the task set under 4 fog nodes, respectively. *TST* and *AWT* under different number of tasks in the four algorithms are shown in Figure 6.

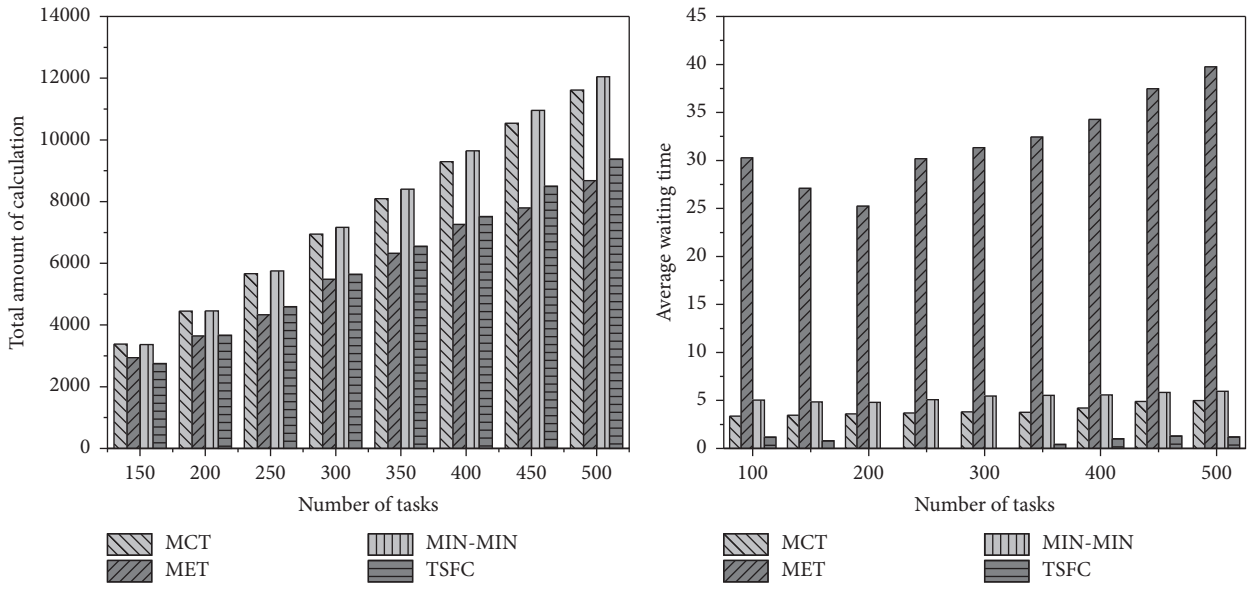
5.4.2. Result Analysis under 6 Fog Nodes. The TSFC, MCT, MET, and MIN-MIN algorithms are used to schedule the task set under 6 fog nodes, respectively. After scheduling, *TST* and *AWT* under different number of tasks in the four algorithms are shown in Figure 7.

We can see from Figures 6(a) and 7(a) that, with the number of tasks increases, the value of *TST* generated by TSFC, MCT, MET, and MIN-MIN algorithms is increasing. However, the value of *TST* generated by the TSFC algorithm is smaller than those by MCT and MIN-MIN algorithms. As the number of tasks increases, the efficiency of TSFC algorithm is higher than MCT and MIN-MIN algorithms. When the number of tasks is small, the value of *TST* generated by the TSFC algorithm is lower than that by the MET algorithm. As the number of tasks increases, the value of *TST* generated by the TSFC algorithm is larger than that by the MET algorithm. Because TSFC algorithm takes task completion time as a main parameter, as the number of tasks increases, the total completion time of scheduled tasks will be closer to the optimal solution.



(a) Comparison of execution time under 4 fog nodes (b) Comparison of average waiting time under 4 fog nodes

Figure 6: (a) Comparison of execution time under 4 fog nodes. (b) Comparison of average waiting time under 4 fog nodes.



(a) Comparison of execution time under 6 fog nodes (b) Comparison of average waiting time under 6 fog nodes

Figure 7: (a) Comparison of execution time under 6 nodes. (b) Comparison of average waiting time under 6 fog nodes.

The values of *AWT* generated by TSFC, MCT, MET, and MIN-MIN algorithms are stable from Figures 6(b) and 7(b). The value of *AWT* generated by the TSFC algorithm is less than that by MCT, MET, and MIN-MIN algorithms (the value of *AWT* of MIN-MIN algorithm in Figure 6(b) is better when the number of tasks is larger). The minimum value of *AWT* generated by TSFC algorithm in Figure 6(b) is only 3.7% of MET's, and the maximum value of *AWT* is only 35.1% of MET's. The minimum value of *AWT* generated by TSFC algorithm in Figure 7(b) is equal to 0. Because the MET algorithm takes the shortest execution time of tasks as the main scheduling parameter, the execution time of different tasks on the same fog nodes is proportional, so it will cause most of the tasks to be scheduled on the same fog node and resulting in a much higher *AWT* value. The TSFC algorithm schedules tasks which have minimum value in minimum completion time, and it shortens the value of task waiting time as much as possible, so the value of *AWT* is smaller.

In summary, the value of *TST* and *AWT* generated by TSFC algorithm is better than MCT, MET, and MIN-MIN

algorithms. The TSFC algorithm is superior to MCT, MET, and MIN-MIN algorithms in the experiments.

6. Conclusion

The fog computing is a new paradigm which attracts lots of attention. Providing satisfactory computation performance is a great challenge in the fog computing environment. In this paper, we proposed an I-Apriori algorithm by improving the Apriori algorithm. Experimental results show that the I-Apriori algorithm can improve the efficiency of generating frequent itemsets effectively. A novel task scheduling model and a novel TSFC algorithm of fog computing environment are proposed based on the I-Apriori algorithm. Association rules are generated by the I-Apriori algorithm which act as an important parameter of TSFC task scheduling algorithm. Experimental results show that TSFC algorithm has better performance than other similar algorithms in terms of task total execution time and average waiting time.

In this article, there are some other issues that do not involve, for example, bandwidth between processors, multilayer task scheduling in fog computing, and others. In future work, we will explore these areas. Furthermore, we will apply TSFC algorithm to other areas, such as oblivious RAM, string mapping, and match problems.

Data Availability

All data generated or analyzed during this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. 61772205), Guangdong Province Natural Science Foundation Team Project (Grant no. 10351806001000000), Guangdong Provincial Scientific and Technological Projects (Grants nos. 2016A010101018, 2016A010119171, 2016A010106007, and 2016B090927010), Guangdong Province Advanced and Key Technology Creative Research Project (Grant no. 2014B010110004), Nansha Science and Technology Projects (Grant no. 2017GJ001), and the Industry-University-Academy Collaborative Innovation Key Project of Guangzhou (Grant no. 201604016074).

References

[1] D. Rahbari, S. Kabirzadeh, and M. Nickray, "A security aware scheduling in fog computing by hyper heuristic algorithm," in *Proceedings of the 2017 3rd Iranian Conference on Intelligent Systems and Signal Processing (ICSPIS)*, pp. 87–92, Shahrood, December 2017.

[2] C. Puliafito, E. Mingozzi, and G. Anastasi, "Fog Computing for the Internet of Mobile Things: Issues and Challenges," in *Proceedings of the 2017 IEEE International Conference on Smart Computing, (SMARTCOMP '17)*, China, May 2017.

[3] X.-Q. Pham and E.-N. Huh, "Towards task scheduling in a cloud-fog computing system," in *Proceedings of the 18th Asia-Pacific Network Operations and Management Symposium, (APNOMS '16)*, Japan, October 2016.

[4] S. Yi, Z. Hao, Z. Qin, and Q. Li, "Fog computing: Platform and applications," in *Proceedings of the 3rd Workshop on Hot Topics in Web Systems and Technologies, (HotWeb '15)*, pp. 73–78, USA, November 2015.

[5] X. Lyu, C. Ren, W. Ni, H. Tian, and R. P. Liu, "Distributed Optimization of Collaborative Regions in Large-Scale Inhomogeneous Fog Computing," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 574–586, 2018.

[6] M. Huang, Y. Liu, N. Zhang et al., "A Services Routing Based Caching Scheme for Cloud Assisted CRNs," *IEEE Access*, vol. 6, pp. 15787–15805, 2018.

[7] X. Liu, M. Dong, Y. Liu, A. Liu, and N. Xiong, "Construction Low Complexity and Low Delay CDS for Big Data Code Dissemination," *Complexity*, vol. 2018, Article ID 5429546, 19 pages, 2018.

[8] L. Ni, J. Zhang, C. Jiang, C. Yan, and K. Yu, "Resource Allocation Strategy in Fog Computing Based on Priced Timed Petri Nets," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1216–1228, 2017.

[9] Q. Zhu, B. Si, F. Yang, and Y. Ma, "Task offloading decision in fog computing system," *China Communications*, vol. 14, no. 11, pp. 59–68, 2017.

[10] M. Mukherjee, L. Shu, and D. Wang, "Survey of Fog Computing: Fundamental, Network Applications, and Research Challenges," *Communications Surveys & Tutorials*, 2018.

[11] S. Gu, Q. Zhuge, J. Yi, J. Hu, and E. H.-M. Sha, "Optimizing Task and Data Assignment on Multi-Core Systems with Multi-Port SPMs," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 9, pp. 2549–2560, 2015.

[12] W. Lin, S. Xu, L. He, and J. Li, "Multi-resource scheduling and power simulation for cloud computing," *Information Sciences*, vol. 397-398, pp. 168–186, 2017.

[13] K. Chronaki, A. Rico, M. Casas et al., "Task scheduling techniques for asymmetric multi-core systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 7, pp. 2074–2087, 2017.

[14] G. Lucarelli, F. Mendonca, and D. Trystram, "A new on-line method for scheduling independent tasks," in *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, (CCGRID '17)*, pp. 140–149, Spain, May 2017.

[15] J. Wu and X.-J. Hong, "Energy-Efficient Task Scheduling and Synchronization for Multicore Real-Time Systems," in *Proceedings of the IEEE 3rd international conference on big data security on cloud*, pp. 179–184, China, May 2017.

[16] C. Tang, X. Wei, S. Xiao et al., "A Mobile Cloud Based Scheduling Strategy for Industrial Internet of Things," *IEEE Access*, vol. 6, pp. 7262–7275, 2018.

[17] T. Li, Y. Liu, L. Gao, and A. Liu, "A cooperative-based model for smart-sensing tasks in fog computing," *IEEE Access*, vol. 5, pp. 21296–21311, 2017.

[18] Y. Liu, J. E. Fieldsend, and G. Min, "A Framework of Fog Computing: Architecture, Challenges, and Optimization," *IEEE Access*, vol. 5, pp. 25445–25454, 2017.

[19] H. Wang, W. Wang, Z. Cui, X. Zhou, J. Zhao, and Y. Li, "A new dynamic firefly algorithm for demand estimation of water resources," *Information Sciences*, vol. 438, pp. 95–106, 2018.

[20] W. Lin, S. Xu, J. Li, L. Xu, and Z. Peng, "Design and theoretical analysis of virtual machine placement algorithm based on peak workload characteristics," *Soft Computing*, vol. 21, no. 5, pp. 1301–1314, 2017.

[21] W. Chen, L. Peng, J. Wang et al., "Inapproximability results for the minimum integral solution problem with preprocessing over infinity norm," *Theoretical Computer Science*, vol. 478, pp. 127–131, 2013.

[22] Y. Wang, K. Li, and K. Li, "Partition Scheduling on Heterogeneous Multicore Processors for Multi-dimensional Loops Applications," *International Journal of Parallel Programming*, vol. 45, no. 4, pp. 827–852, 2017.

[23] F. Saqib, A. Dutta, J. Plusquellic, P. Ortiz, and M. S. Pattichis, "Pipelined decision tree classification accelerator implementation in FPGA (DT-CAIF)," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 64, no. 1, pp. 280–285, 2015.

[24] R. Bruni and G. Bianchi, "Effective Classification Using a Small raining Set Based on iscretization and Statistical Analysis," *IEEE Transactions On knowledge and data engineering*, vol. 27, no. 9, pp. 2349–2361, 2015.

[25] C. Zhou, B. Cule, and B. Goethals, "Pattern Based Sequence Classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 5, pp. 1285–1298, 2016.

[26] B. Tang, H. He, P. M. Baggenstoss, and S. Kay, "A Bayesian Classification Approach Using Class-Specific Features for Text Categorization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 6, pp. 1602–1606, 2016.

[27] Z. Liu, Y. Huang, J. Li, X. Cheng, and C. Shen, "DivORAM: Towards a practical oblivious RAM with variable block size," *Information Sciences*, vol. 447, pp. 1–11, 2018.

[28] B. Li, Y. Huang, Z. Liu, J. Li, Z. Tian, and S. Yiu, "HybridORAM: Practical oblivious cloud storage with constant bandwidth," *Information Sciences*, 2018.

[29] W. Chen, Z. Chen, N. F. Samatova, L. Peng, J. Wang, and M. Tang, "Solving the maximum duo-preservation string mapping problem with linear programming," *Theoretical Computer Science*, vol. 530, pp. 1–11, 2014.

[30] Y. Huang, W. Li, Z. Liang, Y. Xue, and X. Wang, "Efficient business process consolidation: combining topic features with structure matching," *Soft Computing*, vol. 22, no. 2, pp. 645–657, 2018.

[31] W. Lin, C. Zhu, J. Li, B. Liu, and H. Lian, "Novel algorithms and equivalence optimisation for resource allocation in cloud computing," *International Journal of Web and Grid Services*, vol. 11, no. 2, pp. 69–78, 2015.

[32] M. Maheswaran, S. Ali, H. J. Siegel, D. Hensgen, and R. F. Freund, "Dynamic mapping of a class of independent tasks onto heterogeneous computing systems," *Journal of Parallel and Distributed Computing*, vol. 59, no. 2, pp. 107–131, 1999.

[33] T. D. Brauny, H. Siegely, N. Becky et al., "A Comparison Study of Static Mapping Heuristics for a Class of Meta-tasks on Heterogeneous Computing Systems," *parallel & distributed computing*, vol. 61, no. 6, pp. 810–837, 2001.

[34] Y. Li, G. Wang, L. Nie, Q. Wang, and W. Tan, "Distance metric optimization driven convolutional neural network for age invariant face recognition," *Pattern Recognition*, vol. 75, pp. 51–62, 2018.

[35] M. Xiao, Y. Yin, Y. Zhou, and S. Pan, "Research on improvement of apriori algorithm based on marked transaction compression," in *Proceedings of the 2nd IEEE Advanced Information Technology, Electronic and Automation Control Conference, (IAEAC '17)*, pp. 1067–1071, China, March 2017.

[36] V. S. Tseng, C.-W. Wu, P. Fournier-Viger, and P. S. Yu, "Efficient algorithms for mining the concise and lossless representation of high utility itemsets," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 3, pp. 726–739, 2015.

[37] W. Lin, Z. Wu, L. Lin, A. Wen, and J. Li, "An ensemble random forest algorithm for insurance big data analysis," *IEEE Access*, vol. 5, pp. 16568–16575, 2017.

[38] J. Yang, H. Huang, and X. Jin, "Mining web access sequence with improved apriori algorithm," in *Proceedings of the 20th IEEE International Conference on Computational Science and Engineering and 15th IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, CSE and EUC 2017*, pp. 780–784, China, July 2017.

[39] S. Zhang, Z. Du, and J. T. L. Wang, "New techniques for mining frequent patterns in unordered trees," *IEEE Transactions on Cybernetics*, vol. 45, no. 6, pp. 1113–1125, 2015.

[40] D. Hoang and T. D. Dang, "FBRC:Optimization of task scheduling in Fog-based Region and Cloud," in *Proceedings of the 16th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, 11th IEEE International Conference on Big Data Science and Engineering and 14th IEEE International Conference on Embedded Software and Systems, Trustcom/BigDataSE/ICESS 2017*, pp. 1109–1114, Australia, August 2017.

[41] C. A. Brennand, J. M. Duarte, and A. P. Silva, "SimGrid: A simulator of network monitoring topologies for peer-to-peer based computational grids," in *Proceedings of the 2016 8th IEEE Latin-American Conference on Communications (LATINCOM)*, pp. 1–6, Medellin, Colombia, November 2016.

[42] A. Degomme, A. Legrand, G. S. Markomanolis, M. Quinson, M. Stillwell, and F. Suter, "Simulating MPI Applications: The SMPI Approach," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 8, pp. 2387–2400, 2017.

[43] A. Mohammed, A. Eleliemy, and F. M. Ciorba, "Towards the Reproduction of Selected Dynamic Loop Scheduling Experiments Using SimGrid-SimDag," in *Proceedings of the 19th international conference on high performance computing and communications; IEEE 15th international conference on smart city; IEEE 3rd international conference on data science and systems*, pp. 623–626, Bangkok, December 2017.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 5296406, 8 pages
https://doi.org/10.1155/2018/5296406

Research Article

Fog Computing-Assisted Energy-Efficient Resource Allocation for High-Mobility MIMO-OFDMA Networks

Lingyun Lu,[1] Tian Wang,[1] Wei Ni,[2] Kai Li,[3] and Bo Gao[1]

[1]*College of Computer Science, Beijing Jiaotong University, Beijing, China*
[2]*Cyber-Physical System, Data61, CSIRO, Australia*
[3]*Real-Time and Embedded Computing Systems Research Centre (CISTER), 4249015 Porto, Portugal*

Correspondence should be addressed to Tian Wang; 17120423@bjtu.edu.cn

Received 3 May 2018; Accepted 27 June 2018; Published 11 July 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Lingyun Lu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a suboptimal approach for resource allocation of massive MIMO-OFDMA systems for high-speed train (HST) applications. An optimization problem is formulated to alleviate the severe Doppler effect and maximize the energy efficiency (EE) of the system. We propose to decouple the problem between the allocations of antennas, subcarriers, and transmit powers and solve the problem by carrying out the allocations separately and iteratively in an alternating manner. Fast convergence can be achieved for the proposed approach within only several iterations. Simulation results show that the proposed algorithm is superior to existing techniques in terms of system EE and throughput in different system configurations of HST applications.

1. Introduction

Recent development and deployment of high-speed trains (HSTs) have dramatically improved the efficiency and user experience in interstate transportations. However, providing high data rates and good quality of service (QoS) to passengers in the presence of rapidly varying channel conditions and scarce bandwidth availability is a challenging task [1]. Critical challenges have arisen from real-time communications between HSTs and fixed base stations (BS). Existing narrow-band railway communication systems, such as GSM-R, are not suitable for HSTs due to typically low capacity. 5G technology is currently adopting a so-called network densification approach, which involves the deployment of a large number of base stations (BSs), to increase the network coverage and provide higher throughput to the users [2]. Orthogonal-Frequency Division-Multiple-Access (OFDMA) has been extensively adopted for wideband communications, but severe Doppler shift exists in the communication process because of high mobility, resulting in the difficulties in channel estimation [3] and subsequently destructive inter-carrier interference (ICI) [4]. On the other hand, increasing the number of antennas at both transmitters and receivers, also known as Multiple-Input Multiple-Output (MIMO), can improve robustness against ICI. Particularly, MIMO with a large number of antennas has been increasingly studied for enhancing quality and reliability of wideband wireless communications. Unfortunately, the benefits do not come for free. Energy consumption would grow substantially, as the number of antennas increases. An energy-efficient resource allocation of MIMO-OFDMA is expected to balance spectral efficiency and energy efficiency (EE) [5].

There has been a lot of work on wireless resource allocation in static and low-speed mobile system. In [6], it was revealed that network energy can be saved by assigning nonoverlapping frequency bands to different cells. In [7], a power loading algorithm was proposed to maximize the EE of MIMO. In [8], the authors investigated the energy-efficient bandwidth allocation in downlink flat fading OFDMA channels and maximized the numbers of bits transmitted per joule, by using the Lagrangian and time-sharing techniques. In [9], the authors proposed a hybrid structure of resource allocation in OFDMA cellular systems, which maximized both the EE and the downlink system capacity. The proposed structure, combined with resource allocation, was shown to improve the EE and the system capacity of OFDMA. In [10],

the resource allocation for energy-efficient OFDMA systems was formulated as a mixed nonconvex and combinatorial optimization problem and solved by exploiting fractional programming. In [11], the energy-efficient configuration of spatial and frequency resources was studied to maximize the EE for downlink MIMO-OFDMA systems in the absence of channel state information (CSI) at the BS. However, none of the existing works have taken into account the destructive ICI. For HSTs at a speed of over 500km/h, the fast time-varying channel and the severe Doppler shift have yet to be addressed, and high-mobility communication shall be one of the most important and extreme use scenarios in future 5th generation (5G) mobile communication networks [12–16]. The OFDMA resource allocation strategy was designed for fast-changing mobile environments in [17], where a suboptimal allocation policy was developed at a significant cost of computational complexity.

Fog computing, also known as fogging, is an architecture that uses edge devices to carry out a substantial amount of local computation, storage, and communication [18–20]. We use a fog server at the BS to concentrate data, data processing, and applications. The fog server can increase overall computing capability, which helps in efficient resource allocation and utilization.

Fog computing emphasizes proximity to end-users and client objectives, dense geographical distribution and local resource pooling, latency reduction, and backbone bandwidth savings. Therefore, we use this technology to provide practical value for real-time implementation of HST communications.

This paper aims to design an efficient resource allocation strategy to improve the communication performance of HSTs. After analyzing the multiuser MIMO-OFDMA downlink system, the influence of mobility on the system is quantified. A mathematical model is put forth to maximize the EE of the system. To tackle the problem, an iterative algorithm with fast convergence is proposed. Specifically, we propose to decouple the problem between the allocations of antennas, subcarriers, and transmit powers and solve the problem by carrying out the allocations separately and iteratively in an alternating manner. Fast convergence can be achieved for the proposed approach within only several iterations. Simulation results demonstrate the gain of the proposed approach in terms of EE and throughput, as compared with existing schemes.

The rest of the paper is organized as follows. We present the system model in Section 2 and formulate and solve the problem of interest in Section 3. In Section 4, the simulation results are provided, followed by conclusions in Section 5.

2. System Model

The system of interest is a multiuser MIMO-OFDMA system, as illustrated in Figure 1, where there is a fixed BS equipped with M transmit antennas ($M \gg 1$) and K user terminals located in a HST. A fog server is employed at the BS to help the resource allocation computation. Each of the user equipment has a single receive antenna. The users share radio resources for down services. Each of the user equipment has a single

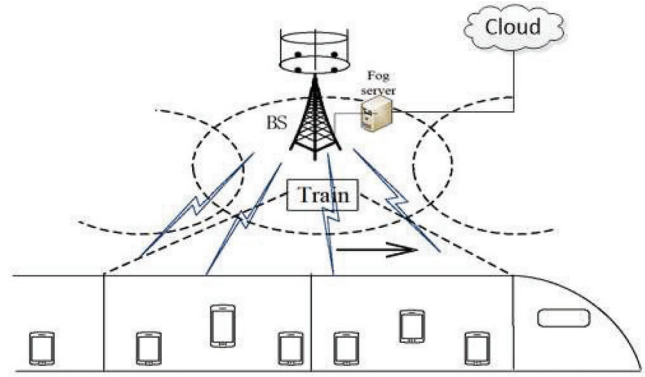


Figure 1: Networks architecture for multiuser MIMO system.

receive antenna. The users share radio resources for down services. Different users are assigned with different OFDM subcarriers and different antennas, given the large number of transmit antennas. Coherent beamforming is carried out at the BS to produce physical beams towards the users.

The speed of HST can lead to severe Doppler shifts. Let $h_{k,n}$ denote the complex channel gain between the BS and user k on subcarrier n. The total number of subcarriers is N. The knowledge on $h_{k,n}$ can be inaccurate at the BS, because of the fast-changing HST environment and hence estimation errors. We assume

$$h_{k,n} = \widehat{h}_{k,n} + \Delta h_{k,n}, \tag{1}$$

where $\widehat{h}_{k,n}$ is the estimate of $h_{k,n}$ at the BS and $\Delta h_{k,n}$ is an independent and identically distributed (i.i.d.) measurement error. $\Delta h_{k,n}$ yields a complex Gaussian distribution due to the use of the Minimum Mean Square Error (MMSE) estimators. $\Delta h_{k,n} \sim N(\mu, \sigma_e^2)$ and

$$\sigma_e^2 = \frac{1}{1 + (\Delta f / f_d)(p_{k,n}/n_0)} \tag{2}$$

where Δf is the subcarrier interval and f_d is the maximum Doppler shift which can be written as $f_d = V \cdot f_c / c$. c is the speed of light. f_c is the carrier frequency. $p_{k,n}$ is the transmit power allocated to user k on subcarrier n. n_0 is the noise power spectral density [10].

We assume that each subcarrier has an equal bandwidth of B. Therefore, the total bandwidth of the system is $B_{tot} = NB$. We also assume that each subcarrier is assigned an equal transmit power; i.e., $p_{k,n} = P_k / b_k = p_k$, where b_k and P_k are the number of subcarriers and the transmit power of the BS allocated to user k, respectively.

The Doppler shift can compromise the orthogonality between OFDM subcarriers, resulting in ICI [22]. At a speed of V, the power of ICI on a subcarrier can be written as [23]

$$ICI(V) = \sum_{n=1}^{N} \frac{(T_s f_d)^2}{2} \sum_{j=1, j\neq n}^{N} \frac{1}{(j-n)^2} \tag{3}$$

where T_S denotes the duration of an OFDM symbol.

In the case that $M \longrightarrow \infty$, the receive signal-to-noise ratio (SNR) can be approximated to [17]

$$\rho_{k,n} \approx \frac{p_{k,n} l_k M_k \left(1 - \sigma_e^2\right)}{n_0 B + p_k ICI\,(V)} \tag{4}$$

where M_k is the number of antennas of the BS assigned to user k.

The asymptotic rate of the MIMO can be achieved based on the random matrix theory [17]. Specifically, the rate asymptotically converges to the average rate in mean square. The asymptotic rate can be replaced with the average data rate. The total data rate of user k converges to

$$r_k = b_k B \log_2 \left(1 + \frac{p_k l_k M_k \left(1 - \sigma_e^2\right)}{n_0 B + p_k ICI\,(V)}\right). \tag{5}$$

We also consider nonideal circuit power at the BS. We can adopt a linear model [24] at the BS to characterize the circuit power consumption, as given by

$$\phi = P_c \max_k \{M_k\} + \sum_{k=1}^{K} b_k p_k + P_0 \tag{6}$$

where P_c is the power consumption per active antenna, consisting of the power consumption of filtering, mixing, power amplification, and digital-to-analog conversion. P_0 is the constant part of the power consumption at the BS and is independent of the number of active antennas.

3. Optimization Problem Formulation

The goal of this paper is to maximize the EE of the BS, which can be formulated as

$$\begin{aligned} \max_{MBP} \quad & \left\{Q\left(\boldsymbol{M}, \boldsymbol{B}, \boldsymbol{P}\right) = \frac{R\left(\boldsymbol{M}, \boldsymbol{B}, \boldsymbol{P}\right)}{\phi\left(\boldsymbol{M}, \boldsymbol{B}, \boldsymbol{P}\right)}\right\} \\ \text{s.t.} \quad & C1: \sum_{k=1}^{K} P_k \le P_T, \\ & C2: r_k \ge R_{\min}, \\ & C3: \sum_{k}^{K} b_k \le N, \\ & C4: M_k \le M \end{aligned} \tag{7}$$

where, given (5) and (6), the EE of the BS can be written as

$$\begin{aligned} Q &= \frac{R}{\phi} \\ &= \frac{\sum_{k=1}^{K} b_k B \log_2 \left(1 + p_k l_k M_k \left(1 - \sigma_e^2\right) / \left(n_0 B + p_k ICI_n\,(V)\right)\right)}{P_c \max_k \{M_k\} + \sum_{k=1}^{K} b_k p_k + P_0}; \end{aligned} \tag{8}$$

the vector $\boldsymbol{B} = [b_1, b_2, \ldots, b_K]^{\mathrm{T}}$ collects the subcarrier allocation of all K users; $\boldsymbol{M} = [M_1, M_2, \ldots, M_K]^{\mathrm{T}}$ collects the antenna allocations for the users; and $\boldsymbol{P} = [P_1, P_2, \ldots, P_K]^{\mathrm{T}}$ collects the power allocations of the users. The constraint C1 specifies the total transmit power constraint P_T. C2 specifies the minimum data rate per user. C3 and C4 restrict the total numbers of subcarriers and antennas, respectively.

Clearly, problem (7) is a combinatorial mixed integer programming problem. The objective of (7) also has a fractional form with variables in the denominator of the objective. All this makes (7) a NP-hard nonconvex problem with poor tractability. In order to solve the problem efficiently, we develop a suboptimal solution, where the subcarriers allocation, antennas, and transmit powers are optimized separately and sequentially in an alternating manner.

3.1. Subcarrier Allocation. Given M and P, we first propose to allocate subcarriers to maximize the EE while satisfying the minimum data rates of the users. According to the objective of (8), the subcarrier allocation can be expressed as

$$b_k = \arg\max_{\boldsymbol{B}} Q\left(\boldsymbol{M}, \boldsymbol{B}, \boldsymbol{P}\right). \tag{9}$$

We propose to allocate subcarriers based on the criterion of EE. First, we calculate the number of subcarriers allocated to each user according to the minimum data rate of the user. Then, we choose the user with the highest EE and allocate a subcarrier to the user, one user after another, and this repeats until all users are allocated or all subcarriers are assigned. The proposed allocation of subcarriers can be summarized in Algorithm 1.

3.2. Transmit Power and Antenna Allocation. Given the subcarrier allocation developed in Section 3.1, problem (7) can be reformulated to a fractional programing problem with respect to M and P, as given by [25]

$$\begin{aligned} F\left(q\right) = \max \quad & \left\{R\left(\boldsymbol{M}, \boldsymbol{P}\right) - q\phi\left(\boldsymbol{M}, \boldsymbol{P}\right)\right\} \\ \text{s.t.} \quad & \text{C1, C2, C4.} \end{aligned} \tag{10}$$

This is mixed integer programming. We proceed to relax the integer constraint C4, i.e., M_k to $\widetilde{M}_k \in [M_{\min}, M]$. $M_{\min}$ is the minimum number of antennas to meet the requirements of uninterrupted transmission for all users [10]. As a result, (10) can be further reformulated as

$$\begin{aligned} \max \quad & \left\{R\left(\boldsymbol{M}, \boldsymbol{P}\right) - q\widetilde{\phi}\left(\boldsymbol{M}, \boldsymbol{P}\right)\right\} \\ \text{s.t.} \quad & C1: \sum_{k=1}^{K} P_k \le P_T, \\ & C2: r_k \ge R_{\min}, \\ & C4: \widetilde{M}_k \in \left[M_{\min}, M\right] \end{aligned} \tag{11}$$

where q is the optimal solution for problem (10).

Initialization Initialize transmit power allocation vector $\boldsymbol{P}^0 = [P_1{}^0, P_2{}^0, \cdots, P_K{}^0]^{\mathrm{T}}$ and antenna allocation vector $\boldsymbol{M}^0 = [M_1{}^0, M_2{}^0, \cdots, M_K{}^0]^{\mathrm{T}}$. Then, we calculate each user's initial data rate $\boldsymbol{R}^0 = [r_1{}^0, r_2{}^0, \cdots, r_K{}^0]^{\mathrm{T}}$.
for *user* $k = 0$ *to* K **do**
  Calculate $b_k = \lceil R_{\min}/r_k^0 \rceil$
end
while $\sum_k^K b_k > N$ **do**
  $\widetilde{k} \longleftarrow \arg\max_{k\in[1,2,\cdots,K]}\{b_k\}$, $b_{\widetilde{k}} \longleftarrow 0$
end
while $\sum_k^K b_k > N$ **do**

$$Q_k = \frac{(b_k+1)B\log_2(1+p_k l_k M_k(1-\sigma_e^2)/(n_0B+p_k ICI_n(V)))}{P_c\max_k\{M_k\}+(b_k+1)p_k+P_0} - \frac{b_k B\log_2(1+p_k l_k M_k(1-\sigma_e^2)/(n_0B+p_k ICI_n(V)))}{P_c\max_k\{M_k\}+b_k p_k+P_0}$$

  $i \longleftarrow \max_k\{Q_k\}$, $b_i = b_i + 1$
end
Output: Subcarrier allocation policy $\boldsymbol{B}^*$.

Algorithm 1

We can prove that (11) is a concave function by evaluating the Hessian matrix of $-F(q)$, as given by

$$\begin{bmatrix} \dfrac{a^2bP_k{}^2}{\ln 2\left(n_0b+cP_k+aP_kM_k\right)^2} & -\dfrac{ab^2n_0}{\ln 2\left(n_0b+cP_k+aP_kM_k\right)^2} \\ -\dfrac{ab^2n_0}{\ln 2\left(n_0b+cP_k+aP_kM_k\right)^2} & \dfrac{ab^2n_0M_k\left(2bcn_0+abn_0M_k+2c^2P_k+2acP_kM_k\right)}{\ln 2\left(n_0b+cP_k+aP_kM_k\right)^2\left(n_0b+cP_k\right)^2} \end{bmatrix}, \tag{12}$$

where $a = l_k(1-\sigma_e^2)$, $b = Bb_k$, and $c = ICI(V)$. Both the determinant of the Hessian matrix and its kth order principal matrix are nonnegative. Thus the Hessian matrix is positive semidefinite. Hence, $-F(q)$ is strictly convex. As a result, the objective function of problem (11) is jointly concave over $(\boldsymbol{M}, \boldsymbol{P})$ while all the constraints are linear. In addition, (11) yields the Slater conditions [26] and therefore holds strong duality. The dual problem of (11) and the primary problem (11) have zero duality gap.

Given q, the Lagrangian function can be written as

$$\begin{aligned} &L(\boldsymbol{M},\boldsymbol{P},\boldsymbol{\lambda},\mu) \\ &= \mu\left(P_T - \sum_{k=1}^K P_k\right) \\ &\quad + \sum_{k=1}^K (1+\lambda_k) b_k B\log_2\left(1+\frac{p_k l_k M_k\left(1-\sigma_e^2\right)}{n_0B+p_k ICI_n(V)}\right) \\ &\quad - \sum_{k=1}^K \lambda_k R_{\min} - q\left[P_c\max_k\{M_k\} + \sum_{k=1}^K b_k p_k + P_0\right], \end{aligned} \tag{13}$$

where $\boldsymbol{\lambda}$ collects the Lagrange multipliers associated with constraint C2 and $\lambda_k \geq 0$; $\mu \geq 0$ is the Lagrange multiplier associated with constraint C1. The dual problem of (11) is given by

$$\min_{\boldsymbol{\lambda},\mu\geq 0}\left\{\max_{\boldsymbol{M},\boldsymbol{P}}\{L(\boldsymbol{M},\boldsymbol{P},\boldsymbol{\lambda},\mu)\}\right\}. \tag{14}$$

Given $(\boldsymbol{\lambda}, \mu)$, according to the KKT conditions, the optimal power allocation, denoted by $\boldsymbol{P}^*$, and antenna allocation, denoted by $\boldsymbol{M}^*$, can be obtained as

$$P_k^* = \frac{b\sqrt{\left(a^2n_0d+4a^2c+4ac^2\right)/n_0d} - a - 2c}{2ac+2ac^2}; \tag{15}$$

$$M_k^* = \left\lceil \frac{b_k B(1+\lambda_k)}{qP_c\ln(2)} - \frac{1}{\alpha_M(V)} \right\rceil, \tag{16}$$

where $a = l_kM_k(1-\sigma_e^2)$, $b = n_0Bb_k$, $c = ICI(V)$, $d = (q+\mu)\ln 2/(1+\lambda_k)$, and $\alpha_M(V) = P_kl_k(1-\sigma_e^2)/(n_0B+P_kICI(V))$.

The subgradient method can be employed to obtain $(\boldsymbol{\lambda}, \mu)$ in an interactive manner, as given by

$$\begin{aligned} \mu(t+1) &= \left[\mu(t) - \delta_1(t)L'(\mu(t))\right]^+; \\ \lambda_k(t+1) &= \left[\lambda_k(t) - \delta_2(t)L'(\lambda_k(t))\right]^+, \end{aligned} \tag{17}$$

Initialization Set $\boldsymbol{\lambda} = \mathbf{0}$, $\mu = 0$.
repeat
  Initialize $\boldsymbol{P}^* = \boldsymbol{P}^0$, $\boldsymbol{M}^* = \boldsymbol{M}^0$.
  repeat
    Update $\boldsymbol{P}^*$, $\boldsymbol{M}^*$ according to (15) and (16) by using antennas and power distribution strategies
  until $L(\boldsymbol{M}, \boldsymbol{P}, \boldsymbol{\lambda}, \mu)$ *converges*.;
  Update $\boldsymbol{\lambda}$ and μ according to (17).
until *(14) converges*.;
Output: $\boldsymbol{P}^*$ and $\boldsymbol{M}^*$.

Algorithm 2

Initialization Initialize $q = 0$ and the maximum tolerance $\varepsilon = 0.01$.
Solve (14) according to CA algorithm and obtain resource allocation policies $\boldsymbol{P}'$, $\boldsymbol{M}'$.
if $R(\boldsymbol{M}', \boldsymbol{P}') - q\phi(\boldsymbol{M}', \boldsymbol{P}') < \varepsilon$ **then**
  return $(\boldsymbol{M}^*, \boldsymbol{P}^*) = (\boldsymbol{M}', \boldsymbol{P}')$ the current combination is the optimal combination
else
  Set $q = R(\boldsymbol{M}', \boldsymbol{P}')/\phi(\boldsymbol{M}', \boldsymbol{P}')$.
end
Output: Resource allocation policies $\boldsymbol{P}^*$, $\boldsymbol{M}^*$ and energy efficiency q^*.

Algorithm 3

where $[x]^+ = \max\{0, x\}$; $t \geq 0$ is the index for the iterations. $\delta_1(t) > 0$ and $\delta_2(t) > 0$ are the step sizes to adjust $\mu(t)$ and $\lambda k(t)$, respectively; and $L'(\mu(t))$ and $L'(\lambda_k(t))$ are the subgradients of the Lagrangian function at $\mu(t)$ and $\lambda_k(t)$, respectively.

The resource allocation policy can be developed based on (15)–(17). Since (λ, μ) and (M, P) can be decoupled in (15), (16), and (17), we can use an improved coordinate ascent (CA) method, where, during each iteration, we first optimize (M, P), given (λ, μ) and q, and then optimize (λ, μ), given (M, P), in an alternating fashion until convergence. Given q, the proposed allocation of transmit antennas and subcarriers is summarized in Algorithm 2.

Finally, we can use the Dinkelbach method [25] to update q. The solution for problem (11) can be summarized in Algorithm 3.

4. Simulation Results

In this section, we simulate the proposed algorithm to verify its effectiveness, where block Rayleigh fading channels are considered. Other simulation parameters are listed in Table 1. We note that the proposed algorithm can be applied under any channel conditions, such as Rician fading channels.

For comparison purpose, the following two resource allocation schemes are also stimulated.

(1) Band allocation based on SNR (BABS) algorithm [27]: it is used for subcarrier allocation. The transmit powers and antennas are allocated in the same way as in the proposed algorithm.

(2) EMMPA algorithm [28]: this algorithm first allocates subcarriers evenly and then allocates the rest of subcarriers to the users with the best channel condition. The scheme developed in [26] is used for the transmit power and antenna allocation.

Figure 2 shows the convergence of the proposed algorithm with different transmit powers, where $K = 20$, $N = 64$, and $V = 500$km/h. It is seen that the EE of the proposed algorithm increases and quickly stabilizes with the growth of iterations. The maximum of the EE can be attained after around only six iterations.

Figure 3 plots the system EE versus the maximum transmit power, where $K = 20$ and $V = 500$km/h. We can see that the system EE increases with maximum transmit power. When the transmit power is large enough, the system EE stabilizes. This is because the BS does not need to activate extra antennas or consume extra power when the

Table 1: Simulation parameters.

Parameter Notation	Value
Speed of electromagnetic wave c	3×108 m/s
Noise power spectral density n_0	2×10^{-7} W/Hz
Minimum data rate	3.0×10^7 bit/s
Center carrier frequency f_c	2.6GHz
Subcarriers number N	64
Total Bandwidth B_{tot}	5MHz
Power consumption per antenna P_c	30dBm [21]
Static power consumption P_0	40dBm [21]
Minimum antenna M_{min}	24 [8]
Maximum antenna Mmax	100

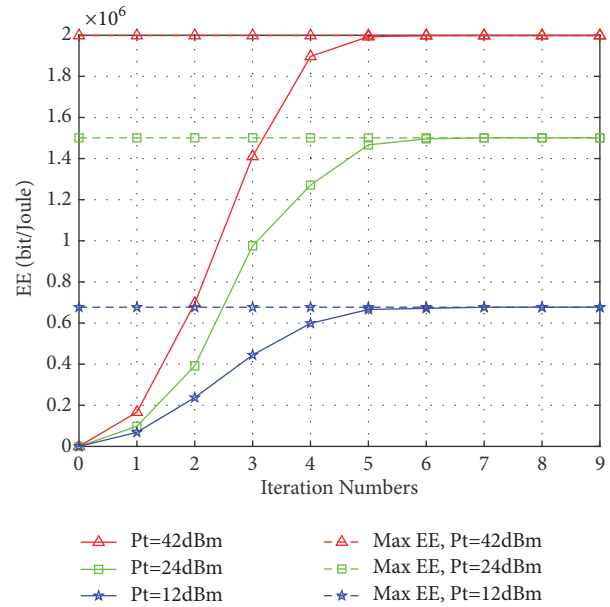


Figure 2: System EE versus iteration numbers for different transmit power with $K = 20$, $N = 64$, $V = 500$km/h.

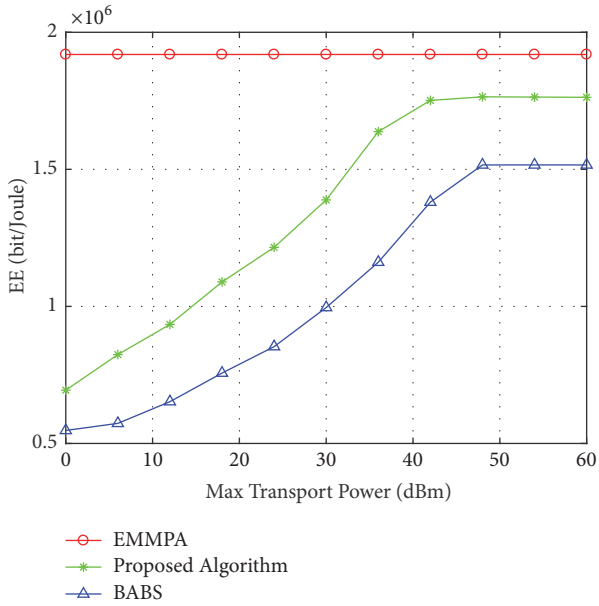


FIGURE 3: System EE versus maximum transmit power with $K = 20$, $V = 500$km/h.

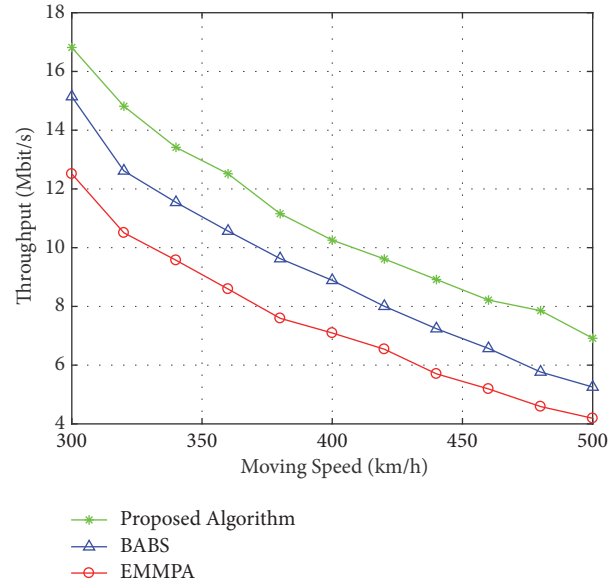


FIGURE 4: System throughput versus the moving speed with $P_T = 40$dBm, $K = 20$.

system maximum EE is reached. The figure also shows that our proposed algorithm performs between the BABS and EMMPA algorithms. The system EE of EMMPA is higher than our proposed algorithm since EMMPA does not have the constraint of P_T and, thus, has a fixed EE value. Additionally, EMMPA is an unconstrained problem to maximize the system EE. The system EE of our proposed algorithm is higher than that of the BABS algorithm because our approach is based on the maximization of EE, while the BABS is based on the minimization of SNR.

Figure 4 presents the system throughput versus the moving speed V, where $P_T = 40$dBm and $K = 20$. We can see that, as V increases, the system throughput significantly decreases. This is because ICI power and channel estimation error are increasingly severe and thus increasingly detrimental to communication quality. The system throughput is about 20.7% higher under our proposed algorithm than under BABS algorithm. EMMPA provides the lowest throughput because of its nature of an unconstrained optimization of maximizing EE without constraints. BABS is to minimize the transmit power while allocating subcarriers. The conclusion drawn is that our proposed algorithm can significantly improve throughput.

Figure 5 shows the system EE versus the number of users K, where $P_T = 40$dBm and $V = 500$km/h. It can be seen that the system EE decreases with the number of users. This is because when the number of subcarriers is fixed, each user can be allocated with a less number of subcarriers, resulting in an increase of the transmit powers to satisfy the users data rate requirement. EMMPA has no demand for the data rate, but the subcarriers assigned to the users who have better channel conditions decrease, and thus system EE also decreases.

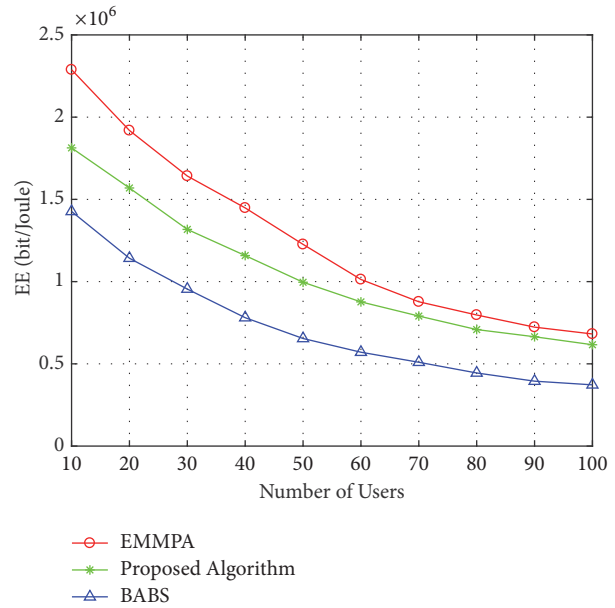


FIGURE 5: System EE versus number of users with $P_T = 40$dBm, $V = 500$km/h.

5. Conclusion

This paper models the resource allocation strategy for multiuser MIMO-OFDMA downlink system for HSTs, where subcarriers, transmit power, and antennas are jointly optimized. Specifically, we propose an iterative suboptimal algorithm to optimize the system EE with fast convergence. In terms of the system performance, simulation results show

that both EE and throughput are improved. Furthermore, the proposed approach is able to fast stabilize within only several iterations and therefore provides practical value for real-time implementation of HST communications.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request. No additional data are available.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This project is supported by the National Natural Science Foundation of China (Grant no. 61771002).

References

[1] A. O. Laiyemo, P. Luoto, P. Pirinen, and M. Latva-Aho, "Feasibility studies on the use of higher frequency bands and beamforming selection scheme for high speed train communication," *Wireless Communications and Mobile Computing*, vol. 2017, 2017.

[2] D. Thembelihle, M. Rossi, and D. Munaretto, "Softwarization of Mobile Network Functions towards Agile and Energy Efficient 5G Architectures: A Survey," *Wireless Communications and Mobile Computing*, vol. 2017, 2017.

[3] C. Zhang, P. Fan, Y. Dong, and K. Xiong, "Service-based high-speed railway base station arrangement," *Wireless Communications and Mobile Computing*, vol. 15, no. 13, pp. 1681–1694, 2015.

[4] D. Gesbert, S. Hanly, H. Huang, S. Shamai Shitz, O. Simeone, and W. Yu, "Multi-cell MIMO cooperative networks: a new look at interference," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 9, pp. 1380–1408, 2010.

[5] I. Chih-Lin, C. Rowell, S. Han, Z. Xu, G. Li, and Z. Pan, "Toward green and soft: a 5G perspective," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 66–73, 2014.

[6] O. Holland, V. Friderikos, and A. H. Aghvami, "Green spectrum management for mobile operators," in *Proceedings of the 2010 Ieee Globecom Workshops*, pp. 1458–1463, Miami, FL, USA, December 2010.

[7] R. S. Prabhu and B. Daneshrad, "Energy-efficient power loading for a MIMO-SVD system and its performance in flat fading," in *Proceedings of the 53rd IEEE Global Communications Conference, GLOBECOM 2010*, IEEE, Miami, FL, USA, December 2010.

[8] A. Akbari, R. Hoshyar, and R. Tafazolli, "Energy-efficient resource allocation in wireless OFDMA systems," in *Proceedings of the IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC '10)*, pp. 1731–1735, IEEE, Instanbul, Turkey, September 2010.

[9] X. Xiao, X. Tao, Y. Jia, and J. Lu, "An energy-efficient hybrid structure with resource allocation in OFDMA networks," in *Proceedings of the 2011 IEEE Wireless Communications and Networking Conference, WCNC 2011*, pp. 1466–1470, Mexico, March 2011.

[10] D. W. K. Ng, E. S. Lo, and R. Schober, "Energy-efficient resource allocation in OFDMA systems with large numbers of base station antennas," *IEEE Transactions on Wireless Communications*, vol. 11, no. 9, pp. 3292–3304, 2012.

[11] G. Y. Li, Z. Xu, C. Xiong et al., "Energy-efficient wireless communications: tutorial, survey, and open issues," *IEEE Wireless Communications Magazine*, vol. 18, no. 6, pp. 28–34, 2011.

[12] K. Xiong, P. Fan, Y. Zhang, and K. Ben Letaief, "Towards 5G High Mobility: A Fairness-Adjustable Time-Domain Power Allocation Approach," *IEEE Access*, vol. 5, pp. 11817–11831, 2017.

[13] Y. Lu, K. Xiong, P. Fan, and Z. Zhong, "Optimal Multicell Coordinated Beamforming for Downlink High-Speed Railway Communications," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 10, pp. 9603–9608, 2017.

[14] K. Xiong, Y. Zhang, P. Fan, H.-C. Yang, and X. Zhou, "Mobile Service Amount Based Link Scheduling for High-Mobility Cooperative Vehicular Networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 10, pp. 9521–9533, 2017.

[15] T. Li, K. Xiong, P. Fan, and K. B. Letaief, "Service-Oriented Power Allocation for High-Speed Railway Wireless Communications," *IEEE Access*, vol. 5, pp. 8343–8356, 2017.

[16] K. Xiong, B. Wang, C. Jiang, and K. J. R. Liu, "A Broad Beamforming Approach for High-Mobility Communications," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 11, pp. 10546–10550, 2017.

[17] R. Couillet and M. Debbah, *Random Matrix Methods for Wireless Communications*, Cambridge University Press, Cambridge, UK, 2011.

[18] X. Lyu, H. Tian, W. Ni, Y. Zhang, P. Zhang, and R. P. Liu, "Energy-Efficient Admission of Delay-Sensitive Tasks for Mobile Edge Computing," *IEEE Transactions on Communications*, vol. 66, no. 6, pp. 2603–2616, 2018.

[19] X. Lyu, W. Ni, H. Tian et al., "Optimal schedule of mobile edge computing for internet of things using partial information," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2606–2615, 2017.

[20] X. Lyu, C. Ren, W. Ni, H. Tian, and R. P. Liu, "Distributed Optimization of Collaborative Regions in Large-Scale Inhomogeneous Fog Computing," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 574–586, 2018.

[21] T. M. Nguyen, V. N. Ha, and L. Bao Le, "Resource allocation optimization in multi-user multi-cell massive MIMO networks considering pilot contamination," *IEEE Access*, vol. 3, pp. 1272–1287, 2015.

[22] Y. Zhao, X. Li, Y. Li, and H. Ji, "Resource allocation for high-speed railway downlink MIMO-OFDM system using quantum-behaved particle swarm optimization," in *Proceedings of the 2013 IEEE International Conference on Communications, ICC 2013*, pp. 2343–2347, Hungary, June 2013.

[23] T. Wang, J. G. Proakis, E. Masry, and J. R. Zeidler, "Performance degradation of OFDM systems due to doppler spreading," *IEEE Transactions on Wireless Communications*, vol. 5, no. 6, pp. 1422–1432, 2006.

[24] T. M. Nguyen and L. B. Le, "Joint pilot assignment and resource allocation in multicell massive MIMO network: Throughput and energy efficiency maximization," in *Proceedings of the 2015 IEEE Wireless Communications and Networking Conference, WCNC 2015*, pp. 393–398, IEEE, New Orleans, LA, USA, March 2015.

[25] W. Dinkelbach, "On nonlinear fractional programming," *Management Science*, vol. 13, no. 7, pp. 492–498, 1967.

[26] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[27] D. Kivanc, G. Li, and H. Liu, "Computationally efficient bandwidth allocation and power control for OFDMA," *IEEE Transactions on Wireless Communications*, vol. 2, no. 6, pp. 1150–1158, 2003.

[28] G. Miao, "Energy-efficient uplink multi-user MIMO," *IEEE Transactions on Wireless Communications*, vol. 12, no. 5, pp. 2302–2313, 2013.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 6285719, 13 pages
https://doi.org/10.1155/2018/6285719

Research Article

Re-ADP: Real-Time Data Aggregation with Adaptive ω-Event Differential Privacy for Fog Computing

Yan Huo, Chengtao Yong, and Yanfei Lu

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

Correspondence should be addressed to Yan Huo; yhuo@bjtu.edu.cn

Received 28 April 2018; Accepted 24 June 2018; Published 8 July 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Yan Huo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In the Internet of Things (IoT), aggregation and release of real-time data can often be used for mining more useful information so as to make humans lives more convenient and efficient. However, privacy disclosure is one of the most concerning issues because sensitive information usually comes with users in aggregated data. Thus, various data encryption technologies have emerged to achieve privacy preserving. These technologies may not only introduce complicated computing and high communication overhead but also do not work on the protection of endless data streams. Considering these challenges, we propose a real-time stream data aggregation framework with adaptive ω-event differential privacy (Re-ADP). Based on adaptive ω-event differential privacy, the framework can protect any data collected by sensors over any dynamic ω time stamp successively over infinite stream. It is designed for the fog computing architecture that dramatically extends the cloud computing to the edge of networks. In our proposed framework, fog servers will only send aggregated secure data to cloud servers, which can relieve the computing overhead of cloud servers, improve communication efficiency, and protect data privacy. Finally, experimental results demonstrate that our framework outperforms the existing methods and improves data availability with stronger privacy preserving.

1. Introduction

Driven by the development of cyberphysical networks, cloud computing, mobile Internet, context-aware smart devices, and the corresponding data experience explosive growth [1]. Cloud computing provides a good solution to deal with the explosive data growth and realize resource sharing [2]. However, cloud-based services may face many challenges, such as high latency and high overhead at cloud servers, due to the centralized structure and the limitation of network bandwidth. Some researches present a distributed service computing paradigm, called fog networking [3–5]. It allocates the capabilities of data gathering, data processing, computing, and applications to devices located at the edge of the network, so as to provide intelligent services for nearby users.

Although fog computing provides great benefits, sensitive and private information mined from raw data (e.g., social relationships and financial transactions) is also exposed to the risk of disclosure. Even more, due to the complexity and diversity of fog nodes, user privacy in a fog network can easily be disclosure. For example, more than 400,000 electronic eyes in Beijing may lead to privacy leakage (e.g., vehicle location information) by data sharing in vehicular ad hoc networks (VANETs) [6–8]. Similarly, we can also gain illegal access to personal health datasets gathered from various sensors of physical sign in body sensor networks (BSNs) and publish these private data without permission [9–11]. As a result, how to protect user privacy is one of the important research issues in fog computing.

Currently, the protection of aggregated data privacy is mainly divided into two types. The first one is designed based on various encryption technologies, such as homomorphic encryption [6, 8–10]. In this type, the encryption technology may cause huge computational overhead as well as lots of computing resources of cloud services [12]. In addition, the cryptography-based schemes may lower communication system efficiency, especially when the system contains many sensors with high reporting frequency. The reason is that a great number of communication resources may be wasted on transmission of encryption information and the corresponding keys. As a result, this is not suitable for energy-limited sensor networks.

The other type of aggregated data privacy preserving is explored by using differential privacy [13]. Compared with the traditional cryptography-based schemes, differential privacy can protect individuals privacy while improving data accuracy as much as possible. For example, the authors of [14] protect privacy of aggregated data with differential privacy by using machine learning. Although there exist many studies based on differential privacy, some challenges cannot be addressed. These studies do not consider the high correlation of time series so as not to generate real-time aggregated data with high accuracy. However, a practical framework should be able to satisfy batch queries in continuous time by exchanging information only once.

To address these challenges, we propose a real-time privacy-preserving stream data aggregation framework based on adaptive ω-event differential privacy under fog computing architecture. In fog computing, data storage, processing, and applications are concentrated in devices on the edge of the network rather than all in the cloud. This type of architecture reduces the amount of data transmitted to the cloud, increases efficiency, and significantly lowers overhead on the server itself. In addition, a fog center is considered as a data aggregator in our framework. It only reports the aggregation secure results to cloud server. In this way, the efficiency of communication can be greatly improved. Moreover, sensors only report raw data instead of encrypted data because our framework does not utilize complex encryption technology. Finally, many techniques for processing time-series data is exploited in our framework to improve the accuracy of aggregation data, such as adaptive sampling, time-series prediction, and filter.

In a nutshell, the main contributions of the paper are summarized as follows.

(i) We propose a real-time privacy-preserving stream data aggregation framework based on adaptive ω-event differential privacy under fog computing architecture. The framework releases the overhead of cloud servers and generates aggregation data with differential privacy preserving.

(ii) In order to promote ω-event differential privacy, we pioneer a novel metric, i.e., quality of privacy (QoP). The QoP design takes into account both the window size ω and errors of published statistics. Using the metric, we adjust the size of window ω adaptively by dint of the design of QoP-based adaptive ω-event mechanism.

(iii) We exploit the long short-term memory (LSTM) to predict time-series data and design the adaptive sampling scheme to improve the accuracy of aggregation data.

(iv) We theoretically analyze the privacy of the proposed Re-ADP framework and demonstrate the high accuracy of aggregated data through numerical simulation results.

The rest of the paper is organized as follows. In Section 2, we introduce preliminaries of differential privacy and ω-event privacy. Then, we provide the system model, the adversary model, and the whole Re-ADP framework to illustrate our problem. In Section 3, we present a QoP-based adaptive ω-event privacy algorithm that includes a dynamical adjustment method of the window size ω. Section 4 presents a smart grouping-based perturbation algorithm, which can reduce the noise added to data significantly. In Section 5, we analyze whether the Re-ADP framework satisfies differential privacy and provides a series of simulation results to discuss the performance of each mechanism in our framework. We then review previous works related to the privacy preserving of aggregated data and differential privacy in Section 6. Finally, Section 7 concludes our paper and explains promising research directions for future work.

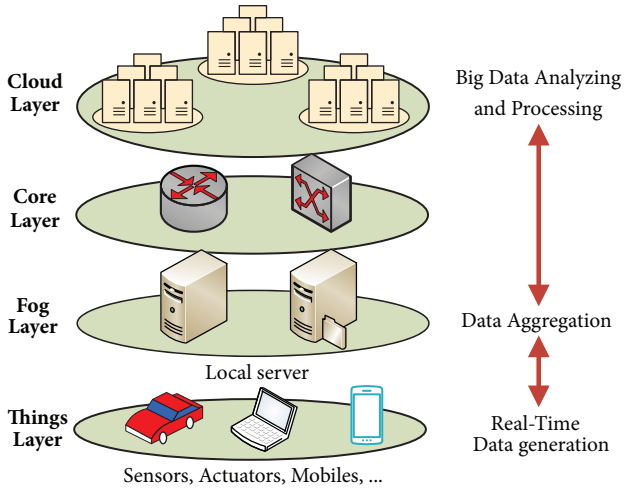


Figure 1: System model.

2. Problem Statement and Preliminaries

2.1. System Model. The system model, shown as Figure 1, is composed of four layers: the things layer, the fog layer, the core layer, and the cloud layer. The function of each layer is described as follows.

(i) **Things layer**, consisting of various smart devices, e.g., sensors, mobiles, and actuators, generates and reports raw data to fog layer.

(ii) **Fog layer**, typically located between IoT devices and core networks, is composed of lots of fog devices. The fog devices can be considered as traditional network devices, such as routers, switches, gateways, or local servers that are specially deployed. In this paper, the devices are mainly composed of local servers and are responsible for (i) gathering and storing data reported from things layer, (ii) computing and aggregating data to satisfy differential privacy, and (iii) responding to query requests from the cloud layer.

(iii) **Core layer** is in charge of transferring and exchanging data between the fog layer and the cloud layer through network protocols such as IP and MPLS.

(iv) **Cloud layer** deploys many cloud servers that can analyze massive aggregated data. Using the analyzing results, cloud services can provide a wide range of services.

2.2. Adversary Model. In this paper, we assume that both the cloud layer and the core layer are untrustworthy. They will try to acquire actual values of gathered data or maliciously tamper data. And the fog layer is considered trusted, which means it can acquire raw data but do not disclose data to the third party.

2.3. Differential Privacy Basics. Differential privacy is one of the most popular notions of privacy in the current research field of privacy preservation. The basic idea is that the record of an individual, regardless of whether or not he is in the dataset, has little impact on the final output, thus protecting the privacy of the individual.

Definition 1. (ϵ-differential privacy [13]): a randomized algorithm $\mathscr{A}$ over datasets can provide ϵ-differential privacy guarantee, if any neighboring datasets D and D' are different on at most one record, and any output $\mathcal{O} \in Range(\mathscr{A})$ satisfies

$$\log \frac{\Pr\left[\mathscr{A}(D) \in \mathcal{O}\right]}{\Pr\left[\mathscr{A}\left(D'\right) \in \mathcal{O}\right]} \leq \epsilon, \tag{1}$$

where $Range(\mathscr{A})$ denotes the range of the randomized algorithm $\mathscr{A}$.

Note that ϵ, called privacy budget, is an important parameter in differential privacy. It represents the privacy level of the randomized algorithm $\mathscr{A}$. More specifically, the level of privacy is inversely proportional to ϵ. Then, a mostly used method to achieve ϵ-differential privacy is the Laplacian mechanism as shown below.

Theorem 2 (the Laplacian mechanism [15]). *Let $\mathscr{D}$ denote a set of datasets. Considering a function $f : \mathscr{D} \longrightarrow \mathbb{A}$, the Laplacian mechanism $\mathscr{A}$ for any dataset $D \in \mathscr{D}$ is*

$$\mathscr{A}(D) = f(D) + Lap\left(\frac{\Delta(f)}{\epsilon}\right), \tag{2}$$

where the noise follows a Laplacian distribution with mean zero and scale $\Delta(f)/\epsilon$. Here, $\Delta(f)$ denotes sensitivity of f, which is defined as the maximum L_1 norm for any neighboring datasets D_1 and D_2.

Theorem 3 (sequential composition [16]). *Let $\mathscr{A}_i$ provide ϵ_i-differential privacy. Then the sequence of $\mathscr{A}_i(D)$ provides $(\Sigma_i \epsilon_i)$-differential privacy.*

Obviously, **Theorem 3** shows that the secrecy level of a combination of several differential privacy-preserving algorithms is the sum of all budgets.

2.4. ω-Event Privacy. ω-event privacy, the abbreviation of ω-event ϵ-differential privacy, is a new privacy model proposed by Kellaris et al. [17]. It can protect privacy for any event sequence occurring at any window of ω time stamp.

We define two neighboring datasets at the ith time stamp as D_i and D_i' and a stream prefix of an infinite series $S = (D_1, D_2, \ldots)$ at the tth time stamp as $S_t = (D_1, D_2, \ldots, D_t)$.

Definition 4 (ω-neighboring [17]). Two stream prefixes S_t, S_t' are ω-neighboring if one of the following two conditions holds:

(i) for each $S_t[i], S_t'[i]$ such that $i \in [t]$ and $S_t[i] \neq S_t'[i]$,

(ii) for each $S_t[i_1], S_t[i_2], S_t'[i_1]$, and $S_t'[i_2]$ with $i_1 < i_2$, $S_t[i_1] \neq S_t'[i_1]$ and $S_t[i_2] \neq S_t'[i_2]$, it holds that $i_2 - i_1 + 1 \leq \omega$,

where ω, a positive integer, denotes the length of a sequence that can be protected at the same time.

Definition 5 (ω-event privacy [17])). A mechanism $\mathscr{A}$ satisfies ω-event differential privacy, if, for all $S \subseteq \mathcal{O}$, all S_t, S_t' at all t, it holds that

$$\log \frac{\Pr\left[\mathscr{A}\left(S_t\right) \in \mathcal{O}\right]}{\Pr\left[\mathscr{A}\left(S_t'\right) \in \mathcal{O}\right]} \leq \epsilon, \tag{3}$$

where $\mathcal{O}$ is the set of all possible outputs of $\mathscr{A}$. A mechanism satisfying ω-event privacy will protect the sensitive information that may be disclosed from a sequence of length ω.

According to the above definitions, we refer to [17] to conclude **Theorem 6**. The theorem enables a ω-event private scheme to view ϵ as the total available privacy budget in any sliding window of size ω and appropriately allocate portions of it across the time stamps.

Theorem 6. *Assume that $\mathscr{A}$ is a mechanism with input stream prefix $S_t[i] = D_i \in \mathscr{D}$ and output $\mathbf{s} = (\mathbf{s}_1, \ldots, \mathbf{s}_t) \in \mathcal{S}$. Supposing $\mathscr{A}$ can be decomposed into $\mathscr{A}_i(D_i) = \mathbf{s}_i$, $i \in [1, t]$, each $\mathscr{A}_i$ generates independent randomness and achieves ω-differential privacy. Then, $\mathscr{A}$ satisfies ω-event privacy if*

$$\forall i \in [t], \quad \sum_{k=i-\omega+1}^{i} \epsilon_i \leq \epsilon. \tag{4}$$

Based on this fundamental theorem, we will explore a novel adaptive ω-event differential privacy mechanism in our work. The proposed mechanism is designed for real-time privacy-preserving stream data aggregation under fog computing architecture.

2.5. Motivation and System Framework. Our motivation is to design a real-time stream data aggregation framework that can protect user privacy in any ω time stamp, allow batch queries, and obtain high-accuracy results. In order to achieve the motivation, we divide our work into two main tasks.

(i) **Protect privacy in any window of ω time stamp.** Servers may query aggregated data within ω time only one round of communication. Therefore, the proposed framework must protect privacy of data generated in ω time stamp. Besides the size of window ω should be adaptively adjusted according to the state of data changes.

Input: The raw data database X_i at the ith time stamp
Output: The aggregation secure statistics R_i
(1) Find out the optimal number of sampling points N
(2) Find out sets of sampling sensors at the current time stamp
(3) Obtain the grouping strategy G_{k_i} via *smart grouping*
(4) Allocate the budget for all sampling sensors
(5) Add Laplacian noise to group G_{k_i} with allocated budget at *the perturbation mechanism*
(6) Report the aggregated secure statistics R_i that is filtered by Kalman filtering.
(7) Update the sampling interval by *adaptive sampling*

Algorithm 1: A real-time stream data aggregation framework with adaptive ω-event differential privacy (Re-ADP).

(ii) **Improve the accuracy of aggregated data.** Because of the Laplacian differential privacy, the proposed framework needs to add random noise to data to guarantee privacy protection. Thus, the framework must reduce extra errors of aggregated data as much as possible on premise of privacy preserving.

In this article, we intend to design an adaptive ω-event based differential privacy-preserving strategy. This strategy in Figure 2 is composed of adaptive ω-event privacy analysis, smart grouping-based perturbation, and the filtering mechanism. Here, we outline the complete process of the proposed Re-ADP strategy, shown in **Algorithm** 1. The first component, illustrated in Section 3, is achieved based on the adaptive sampling and QoP measurement. The second one is presented in Section 4, which is designed based on K-means smart grouping and the corresponding perturbation mechanism. And we exploit the similar filtering mechanism in [18] to reduce errors of aggregated data so as to improve data availability.

3. QoP-Based Adaptive ω-Event Privacy Design

For privacy protection on infinite stream data aggregating, ω-event privacy is a convincing model. The objective is to make a trade-off between utility and privacy to protect all data sequences that occur within all windows of ω time stamp. However, it is not applicable to many realistic scenarios due to the fixed size of the sliding window. The key issue of the unrealistic assumption is that most real-time aggregate data streams collected from sensors are significantly different in various time periods. For example, within successive time stamps, it can be seen that traffic data varies sharply in the daytime but is relatively stable at night. Thus, we introduce a new QoP-based adaptive ω-event privacy mechanism in this section to dynamically adjust the window size ω within different time stamps. The following three subsections describe the key parts to achieve this mechanism, including the QoP definition, the adaptive sampling design, and the adaptive ω-event privacy design.

3.1. Quality of Privacy. Considering the window size ω and errors of aggregated statistics, QoP is proposed to measure the corresponding privacy quality. Assume $\mathbf{x} = \{x_1, \ldots, x_k\}$ and $\mathbf{r} = \{r_1, \ldots, r_k\}$ represent the raw time series in a window and the sanitized time series, respectively. Then, we exploit mean absolute error (MAE) to measure difference between these two time series.

$$MAE(x, r) = \frac{1}{k}\sum_{i=1}^{k} \left| r_i - x_i \right|. \tag{5}$$

Next, we employ a sampling mechanism in the proposed Re-ADP. It may perturb statistics at selected time stamp and approximate the nonsampled statistics with perturbed sampled statistics. Thus, (5) can be rewritten as follows:

$$MAE(x, r) = \frac{1}{k}\sum_{i}^{k}\sum_{j=i}^{i-1} \left| r_i - x_j \right|. \tag{6}$$

As a result, QoP in a window is defined as

$$QoP(x, r) = \sigma\left(\beta \frac{\omega}{MAE(x, r)}\right), \tag{7}$$

where ω is a window size and β is the weight between ω and $MAE(x, r)$. Here, β is set to 0.002 in our experiments. In addition, $\sigma(\cdot)$ is a logistic sigmoid function that is equal to

$$\sigma(x) = \frac{1}{1 - e^{-x}}. \tag{8}$$

The reason that we employ the logistic sigmoid function for normalization is that we do not need to know the general characteristics of the data. Intuitively, as sensor data generated in contiguous time stamps is not independent, there is close correlation among these data when data changes slowly. Meanwhile, with the possibility that sensitive information may be disclosed, the windows size ω should be increased when data changes slowly.

3.2. The Adaptive Sampling Design. In general, a report of noisy data denotes the expenditure of fixed budget ϵ. When protecting all time stamps, the budget allocated to each time stamp will be small if the window size ω is large. In this case, the report will show gigantic errors. This problem can be addressed by using a sampling mechanism (this mechanism can perturb sampled statistics while skipping nonsampled statistics). In this case, we can employ skipping some data points to save budget for future perturbation as

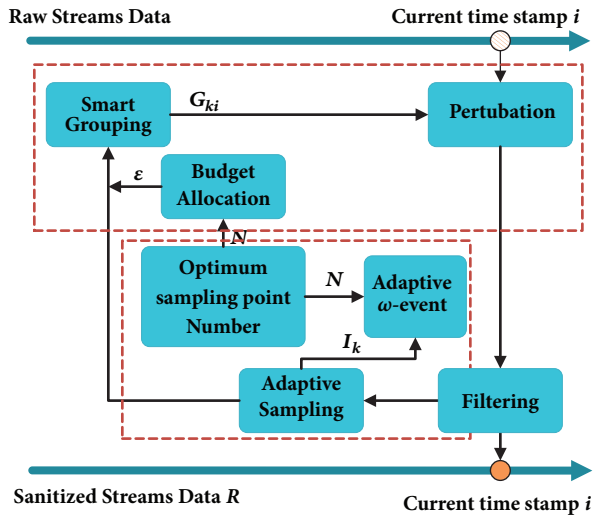


Figure 2: A high-level overview of Re-ADP.

well as improve communication efficiency. Without exploiting the model controller, our mechanism with a proportional-integral-derivative (PID) controller has the advantages of strong robustness and low complexity. Therefore, we exploit the PID controller to change the sampling rate based on dynamic historical data. Firstly, we define the feedback errors of sensor j.

$$E^j_{k_n} = \left| r^j_{k_n} - r^j_{k_{n-1}} \right|, \tag{9}$$

where k_n and k_{n-1} indicate the current sampling data point and the last sampling data point, respectively. It shows that data changes rapidly when the error $E^j_{k_n}$ increases. Then, the full PID error δ^j of sensor j is defined as follows:

$$\delta^j = C_p E^j_{k_n} + C_i \frac{\sum_{o=n-\pi-1}^{n} E^j_{k_o}}{\pi} + C_d \frac{E^j_{k_n}}{k_n - k_{n-1}}, \tag{10}$$

where C_p, C_d, and C_i denote the proportional gain, the integral gain, and the derivative gain, respectively.

Intuitively, the sampling interval needs to be small with rapid data change. Thus, a new sampling interval T is calculated by the following methods.

$$T = \max\left\{1, T_l + \theta_1 \left(1 - \left(\frac{\delta^j}{\theta_2}\right)^2\right)\right\}. \tag{11}$$

In (11), T and T_l denote the current sampling interval and the previous one of sensor j. θ_1 is used to regulate the sampling interval, and θ_2 is used to control the sensitivity of the PID error.

3.3. The Adaptive ω-Event Privacy Algorithm. On the basis of the two sections above, the adaptive ω-event privacy algorithm is proposed in **Algorithm** 2. Note that pseudocodes from line 1 to line 6 are experiment offline over the training set.

We assume that the starting and ending points of the window are both sampling points and there are n sampling points in the current window. As a result, the window size $\omega = \sum_{k=1}^{n-1} T_k$. According to (6) and (7), the QoP in a window can be calculated as follows:

$$QoP(x, r) = \sigma\left(\theta \frac{\sum_{k=1}^{n-1} T_k}{(1/k)\sum_i \sum_{j=i}^{i-1} \left| r_i - x_j \right|}\right). \tag{12}$$

After obtaining N over training test, the adaptive ω-event privacy mechanism is described from line 7 to line 10. In particular, we can adjust the new window size by moving the start point of the window forward or backward Δ time stamps.

4. Smart Grouping-Based Perturbation

A naive method to achieve differential privacy is to inject the Laplacian noise to statistics. Nonetheless, it is likely to introduce more perturbation errors, especially in statistics with small values. Therefore, we present a smart grouping-based perturbation to aggregate sensors with small statistics together in a dynamic way with the change of statistics.

The Smart Grouping Algorithm is presented in **Algorithm** 3. It mainly is divided into three steps. First, it screens out the sensors that needs to be grouped according to the predicted statistics (denoted by $p^i_{t_i}$) by exploiting the LSTM model. Then, it groups sensors that need to be grouped using the K-means algorithm. Finally, aggregated data will be perturbed based on the grouping result. We will elaborate on each step in detail in the following subsections.

4.1. Statistics Prediction with LSTM. To protect privacy of raw data, we use the predicted data instead of real values in the smart grouping-based perturbation algorithm. As mentioned above, whether a sensor needs to be grouped depends on the prediction data of the sensor. In addition, which group each sensor is assigned to also depends on the predicted value.

Input: The current window size ω_l; The fist and last sampling interval T_0, T_l at current window
Output: The new window size ω
(1) Initialize the optimal number of sampling points: $N = 0$
(2) Initialize the maximal QoP: $QoP_{max} = 0$
(3) **for** each $n \in \{2, 3, \cdots, n_{max}\}$ **do**
(4) Calculate the current QoP: QoP'
(5) **if** $QoP' > QoP_{max}$ **then**
(6) $QoP_{max} \longleftarrow QoP'$, $N \longleftarrow n$
(7) Count the number of sampling: n'
(8) **if** $n' \leq N$ and i is sampling point **then**
(9) $\Delta \longleftarrow T_l - T_0$;
(10) Update the new window size: $\omega \longleftarrow \omega_l + \Delta$;
(11) **return** The new window size ω

Algorithm 2: The adaptive ω-event privacy.

Input: S_{t_i}: the sensors that need to be sampled at t_i
Output: The noised data of sensors in S_{t_i}
(1) **for** each sensors $i \in S_{t_i}$ **do**
(2) Predict $p_{t_i}^i$ by using the LSTM model
(3) **if** $p_{t_i}^i > \tau$ **then**
(4) Let the sensor i itself as a group
(5) Add the group to G_{t_i}
(6) **else**
(7) Add the sensor into Φ
(8) Employ K-means algorithm to cluster sensors in Φ according to $p_{t_i}^i$.
(9) Add each group into G_{t_i} based on the cluster results.
(10) Introduce Laplacian noise into each group G_{t_i}.
(11) Allocate the group perturbed statistic to each sensor according to $p_{t_i}^i$.
(12) **return** The noised data of sensors in S_{t_i}

Algorithm 3: Smart grouping-based perturbation.

This means that the accuracy of the predicted value is critical to the accuracy of the final aggregated data. Thus, a good model must be formulated, which can describe characteristics of data change well and predict data accurately.

To achieve accurate prediction, we introduce the LSTM model. In general, a LSTM network [19] has been gradually applied to the time-series analysis [20–22] by profiting from some advantages. In particular, it is a special type of recurrent neural network (RNN), which skillfully solved the problem of gradient vanishing of RNN. A common LSTM unit is composed of a memory cell, an undate gate, an output gate, and a forget gate. The memory cell stores a value (or state) for either long or short terms. It has the ability to remove or increase information to cell state through the well-designed three gates that can transfer information. As a result, we adopt the LSTM network to formulate our model to characterize the nonlinear characteristics of data in our algorithm.

Considering the effectiveness of our Smart Grouping Algorithm, our LSTM network only consists of three layers (shown in Figure 3), i.e., the input layer, the hidden layer, and the output layer. The input layer has k neurons, where the value of k is determined by the number of previously aggregated data to be used for prediction. The output is just one neuron because we just need to predict the value of next time stamp. The hidden layer consists of several LSTM units. W_1 is a weight matrix between the input layer and the hidden layer, while W_2 is that between the output layer and the hidden layer. In addition, each context unit corresponds to a neuron in the hidden layer, which is used to record the output of the hidden layer in one recurrence.

As shown in Figure 3, the historical aggregated data is used as the training data to input to the LSTM model so as to predict the value for each sensor at current time. For example, suppose we need to predict the value generated from sensor j at time t_i (e.g., $p_{t_i}^j$). The previously aggregated data used for prediction is $(r_{t_{i-k}}^j, r_{t_{i-k-1}}^j, \cdots, r_{t_{i-1}}^j)$. We first calculate the output of a hidden layer unit (i.e., $a_{t_i}^j$). Figure 4 shows the detailed structure of a LSTM unit, and $a_{t_i}^j$ is calculated as follows.

First, the LSTM unit determines what information should be forgotten from the cell state by using

$$F_f = \sigma\left(W_f \cdot \left[a_{t_{i-1}}^j, m_{t_i}^j\right] + b_f\right) \tag{13}$$

where $\sigma(\cdot)$ is the logistic sigmoid function, W_f is weight matrix of the forget gate, and b_f is the bias vector of forget gate

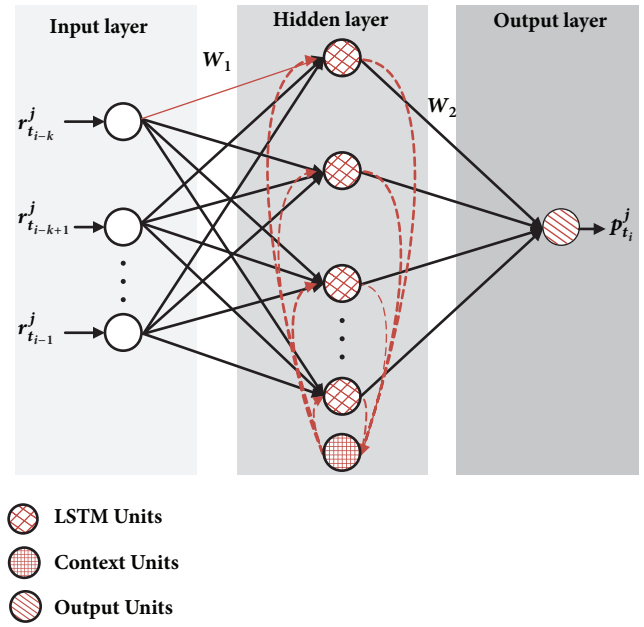


Figure 3: The architecture of the LSTM network.

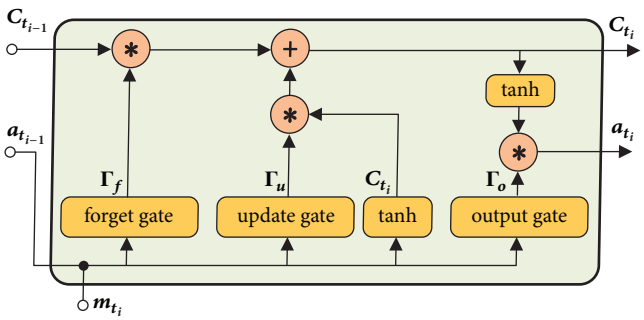


Figure 4: The structure of the LSTM unit.

layer. $a^j_{t_{i-1}}$ is the output of the hidden layer at time t_{i-1}, while $m^j_{t_i}$ is the input of the hidden layer at current time, which is computed as follows.

$$m^j_{t_i} = W_1 \cdot r^j_{t_{i-1}} \tag{14}$$

Next, LSTM employs the following equations to decide what new information needs to be stored in the cell state by using the update gate layer.

$$\Gamma_u = \sigma\left(W_u \cdot \left[a^j_{t_{i-1}}, m^j_{t_i}\right] + b_u\right) \tag{15}$$

$$\widetilde{C}_{t_i} = \tanh\left(W_c \cdot \left[a^j_{t-1}, m^j_{t_i}\right] + b_c\right) \tag{16}$$

where Γ_u indicates which value will be updated and $\widetilde{C}_{t_i}$ represents a vector of new candidate values. W_u and W_c are weight matrices of the input gate layer. And $\tanh(\cdot)$ is defined as follows:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{17}$$

Then, the cell state C_{t_i} is updated based on (18) in current time t_i.

$$C_{t_i} = \Gamma_u * \widetilde{C}_{t_i} + \Gamma_f * C_{t_{i-1}} \tag{18}$$

Here, the output at current time of the hidden layer is controlled by the update gate Γ_u and the forget gate Γ_f.

Finally, based on the latest cell state C_{t_i}, the output of the hidden layer at current time, $a^j_{t_i}$, can be calculated as follows:

$$\Gamma_o = \sigma\left(W_o \cdot \left[a^j_{t-1}, m^j_{t_i}\right] + b_o\right) \tag{19}$$

$$a^j_{t_i} = \Gamma_o * \tanh\left(C_{t_i}\right) \tag{20}$$

where Γ_o is the output gate that determines which part of C_{t_i} should be output.

According to (13), (15), and (19), we can be aware that the LSTM unit has the ability of determining which information is forgotten, updating and outputting intelligently. This ability enables us to predict time series of our network more accurately.

Final prediction data of sensor j at time t_i, e.g., $p^j_{t_i}$, is calculated as follows:

$$p^j_{t_i} = g\left(W_2 \cdot a^j_{t_i}\right) \tag{21}$$

where g is the activation function of the output layer.

Training of the LSTM network: in order to achieve real-time prediction, we should train the network related parameters offline in advance. In addition, we must employ the true statistics of the training set for the sake of the accuracy of the training model. Therefore, for sensor j at time t_i, the input is $(x^j_{t_{i-c}}, x^j_{t_{i-c-1}}, \cdots, x^j_{t_{i-1}})$, and the expected output is the true statistic $x^j_{t_i}$. And, based on the predicted statistic $p^j_{t_i}$, the loss function of our network is defined as below.

$$\mathscr{L}^j_{t_i} = \frac{1}{2}\left(x^j_{t_i} - p^j_{t_i}\right)^2 \tag{22}$$

Using the backpropagation algorithm [23], the training error is propagated to the neurons in the LSTM network. Then, we further calculate training errors caused by each neuron and adjust the corresponding weights to reduce the errors. Details of the training process can be established in [23]. Finally, given the historical aggregated data, the trained LSTM model can predict sensors data in real time.

4.2. K-Means Based Smart Grouping Algorithm. In this subsection, we present a Smart Grouping Algorithm based on the K-means method. The algorithm can smartly aggregate small statistics obtained from sensors in the noise scenarios. First of all, we allocate the budget to each sampling point ϵ_i and then generate an antinoise threshold τ dynamically. Clearly, we can utilize an inverse proportion to characterize the relationship between τ and the corresponding allocated budget. Then, we can obtain the predicted data of each sensor $p^i_{t_i}$ for each time t_i by using the trained LSTM model. According to the above processing, we can exploit K-means algorithm [24] to achieve

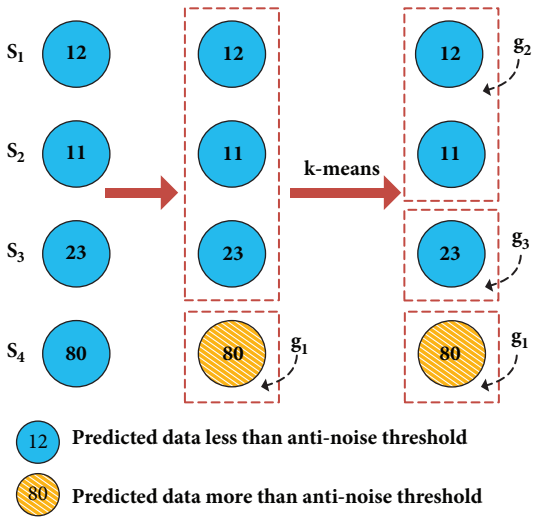


FIGURE 5: An example of smart grouping.

sensors data aggregation in the premise that $p_{t_i}^i$ is smaller than the antinoise threshold τ.

Compared with other algorithms, k-means algorithm is fast and efficient, which is suitable for large data scenarios. Thus, it accords with the data size and real-time requirement of our algorithm. Next, we will introduce how the K-means algorithm works in our scenario. Note that the input is the predicted data of sensors at each time stamp t_i, which need to be grouped as $\{\ldots, p_{t_i}^i, p_{t_i}^j, \ldots.\}$. In particular, we first randomly initialize the K cluster centers and then divide it into clusters where each cluster is closest to its nearest cluster centers for each sensor i. Here, we intend to employ the Euclidean distance to calculate the distance from the current point to the center point. Next, we update the cluster center according to the new clusters obtained from the previous step. The method to update cluster centers is to calculate the mean of all points in the cluster. And the convergent condition is that the minimum squared error of every point to the center point is less than a threshold value or the preset maximum number of iterations. Finally, the algorithm repeats the above two steps until convergence.

Figure 5 is an example to explain the whole process of the Smart Grouping Algorithm. Assuming there are four sensors that need to sampled at time stamp t_i, we define that the predicted statistics are 12, 11, 23, and 80, respectively. The antinoise threshold τ is 50. For S_4, S_4 is an independent group because $80 > 50$, which is added to G_{t_i} (the group strategy of the current time stamp). For S_1, S_2, and S_3, we input them to the K-means algorithm. Note that S_1 and S_2 are clustered into a group while S_3 becomes a single group. Thus, the final group strategy is $G_{t_i} = \{\{S_1, S_2\}, S_3, S_4\}$.

4.3. Smart Grouping-Based Perturbation. To achieve additional noise loading, we intend to exploit the Laplacian mechanism to directly inject noises into aggregated statistics [15] based on results from adaptive sampling. The aggregated statistics do not include the nonsampled statistics that can be approximated by the last aggregated statistics. In this article, we present a scheme of smart grouping-based perturbation. This scheme is composed of a perturbation component and an allocation component. Considering the utilization of the grouping algorithm, we apply the Laplacian mechanism in each group rather than in each sensor in our scheme design.

We assume that a group g has κ sensors and $f(g)$ represents a function to aggregate the number of data contributors in g. Intuitively, because all contributors can only appear in the collection range of a sensor at one time stamp, the sensitivity of the function f is equal to 1; i.e., $\Delta(f) = 1$. Then, the Laplacian mechanism can be employed in group g as follows:

$$\begin{aligned}\mathscr{A}(g) &= f(g) + Lap(\lambda(g)) \\ &= \sum_{j=1}^{\kappa} \sum g[j] + Lap\left(\frac{\Delta(f)}{\epsilon_i}\right),\end{aligned} \tag{23}$$

where $g[j]$ is the jth sensor of the group g and $\lambda(g)$ denotes the scale of Laplacian noises injected into $f(g)$. In order to avoid exceeding the total budget, our scheme considers the smallest budget of a sensor in a group as the budget of the whole group. In this case, the proposed RescueDP strategy does not make full use of the total budget. In addition, we also fix the sampling points in our scheme and allocate the total budget to each sampling point uniformly. Therefore, $\epsilon_i = \epsilon/N$, which leads to making full use of the total budget as well as ensuring not exceeding the total budget.

Next, considering the predicted statistics in each sensor, we allocate the perturbed statistic. The allocation method can avoid errors resulting from the average operation in the RescueDP strategy. Our allocation method is shown as follows:

$$\mathscr{A}(g[j]) = \alpha_j \mathscr{M}(g), \quad \forall j = 1, \ldots, \kappa, \tag{24}$$

where the weight of a sensor, α_j, can be calculated by the predicted statistics of a sensor; i.e.,

$$\alpha_j = \frac{p_{t_i}^j}{\sum_{j=1}^{\kappa} p_{t_i}^j}. \tag{25}$$

According to the smart grouping-based perturbation scheme, the perturbed statistics of a sensor are more accurate.

5. Performance Discussion

In this section, we first analyze the privacy of our proposed Re-ADP framework in theory and then provide several numerical simulations to study the performance of our framework in terms of MAE and QoP.

5.1. Privacy Analysis

Theorem 7. *The proposed Re-ADP framework satisfies ϵ-differential privacy.*

Proof. In the Re-ADP framework, perturbation is the only possible mechanism to disclose private information because it is the only one to access raw data. As a result, if the perturbation mechanism can be proved to satisfy ϵ-differential privacy, the Re-ADP framework can meet the requirement of ϵ-differential privacy subsequently.

On the basis of the smart grouping strategy G_{t_k} at time stamp t_k, each group includes several sensors. We assume that g_m with κ_m sensors is an arbitrary group of G_{t_k}. According to (12), the Laplacian mechanism on group g_m is as follows:

$$\begin{aligned}\mathscr{A}(g_m) &= f(g_m) + Lap(\lambda(g_m)) \\ &= \sum_{j=1}^{\kappa_m} \sum g_m[j] + Lap\left(\frac{\Delta(f)}{\epsilon_{t_k}}\right),\end{aligned} \tag{26}$$

where $g_m[j]$ is the jth sensor of g_m and $\Delta(f) = 1$.

Based on **Definition 1**, $\mathscr{A}(g_m)$ satisfies ϵ_{t_k}-differential privacy. According to Axiom 2.1.1 in [25], postprocessing sanitized data will not reveal privacy as long as sensitive information is not available directly in the postprocessing algorithm. As a result, $\mathscr{A}(g_m[j])$, $\forall j = 1, \cdots, \kappa_i$, also satisfies ϵ_{t_k}-differential privacy. Assume that ϵ'_{t_k} and ϵ_{t_k} represent the budget consumed and the budget allocated for a sensor at timestamp t_k, respectively. If all allocated budget is employed for perturbation in our algorithm, then $\epsilon'_{t_k} = \epsilon_{t_k}$ holds.

Based on **Theorem 6**, the perturbation mechanism of a sensor satisfies ϵ-differential privacy for every t_i and $t_i \in [t]$, if it holds that

$$\sum_{t_k=t_i-\omega+1}^{t_i} \epsilon'_{t_k} \leq \epsilon. \tag{27}$$

The above formula always holds for any sliding window ω timestamp for the reason that $\epsilon'_{t_i} = \epsilon_{t_i}$ holds in our budget allocation algorithm. Thus, the perturbation mechanism on each group can satisfy ϵ-differential privacy. In other words, the Re-ADP algorithm also satisfies ϵ-differential privacy. And this completes the proof of **Theorem 7**. □

5.2. Numerical Simulation. We compare the performance of the proposed Re-ADP strategy with MLDP in [14] and the RescueDP strategy in [26] over two real datasets. The MLDP is a privacy-preserving data aggregation scheme under fog computing based on machine learning, while the RescueDP is the latest strategy that provides ϵ-event privacy for real-time aggregate data publishing. In the simulation, we employ MAE and QoP as metrics to study the performance of the three schemes. The specific expressions of these metrics are given by (6) and (12). Our experiment is conducted in Python environment in Windows 10 operating system. Each experiment is run 100 times and points in the results are the average values of 100 times of each experiment.

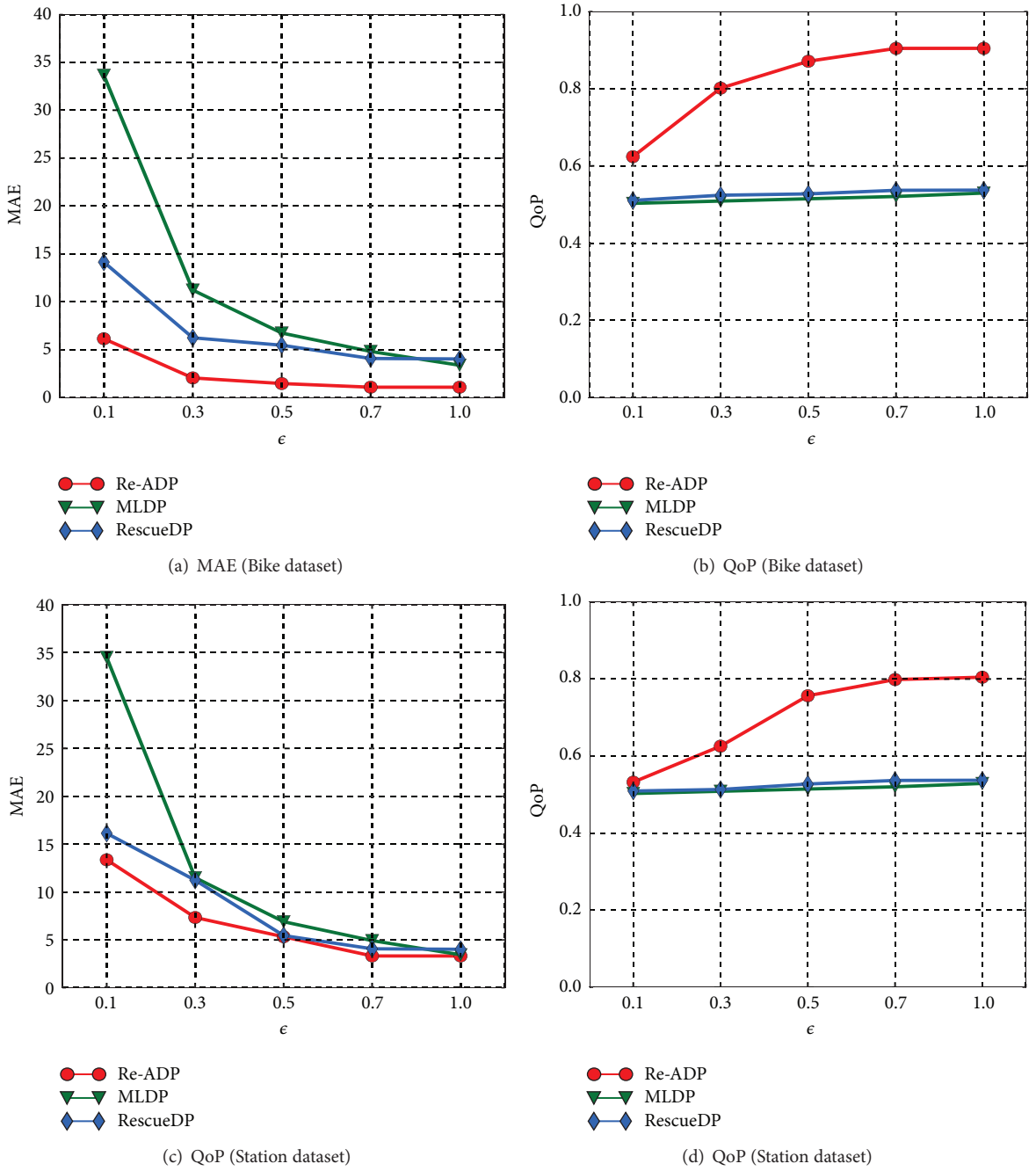
The real-world test datasets to discuss the performance in our experiment are Bike data [27] and Station date [28]. The dataset of Bike provides an accurate data containing the bike share trip in Washington DC for one year from January 1, 2016, to December 31, 2016. It contains a total of 3,333,791 bike share trajectories. Each of them consists of the bike number, the end station and time, and the start station and time. We transform it into a dataset that consists of 368 sensors to count the number of bikes at each bike parking spot in real time. The first three-quarters of the data is split as the training set and the fourth-quarter of the data is used as test set. The dataset of Station consists of the number of passengers of 2116 stations between January 1, 2016, and December 30, 2016. It contains 9,917,584 records and each record reports the number of passengers of a station. Because many stations have very little throughput, we chose 1393 sensors with more throughput to report. And the division of the test set and the training set is the same as the Bike dataset.

In our experiment, the parameters of the PID control are set as follows: $C_p = 0.9, C_d = 0.1, C_i = 0, \pi = 10, \theta_1 = 15$, and $\theta_2 = 10$ for the adaptive sampling mechanism. In **Algorithm 2**, n_{max} is set to be 100. In addition, we obtain $N = 10$ for the Bike dataset and $N = 8$ for the Station dataset by constantly iterating over training set. In K-means based Smart Grouping Algorithm, we set $K = 5$ for Bike dataset and $K = 8$ for the Station dataset, and the result is also the best performance on the training set. The parameters of the LSTM network are set as follows: the previous $k = 50$ history data is used for the input of the LSTM network. So the numbers of input layers' neurons is 50, and 100 is the number of hidden layers' neurons. Besides, we train by iterating 1,000 times. Note that a training process that takes about two minutes is time-consuming. However, we only conduct this process offline, which does not affect the real-time nature of the algorithm.

5.2.1. Utility versus Privacy. Figure 6 provides the trade-off analysis between utility and privacy. It is clear that when ϵ increases, the MAE of three schemes decreases gradually. The reason is that the larger ϵ represents the smaller noise that needs to be injected. Moreover, for two real-world test datasets, the Re-ADP scheme outperforms the other two schemes greatly, especially in a small privacy budget. Also, the QoP of Re-ADP is obviously superior to the other two schemes in a sufficient privacy budget.

The superior performance of the Re-ADP scheme results from the following three aspects. First, due to the design of the optimal number of sampling points and the corresponding privacy budget allocation mechanism, the privacy budget is fully used for private perturbation. Second, the adaptive ω-event privacy mechanism in the Re-ADP scheme satisfies the privacy window adaptively, which improves the practicability of the scheme. Finally, LSTM-based prediction can provide a high-accuracy prediction result for the smart grouping mechanism.

5.2.2. Effect of Adaptive ω-Event Privacy Mechanism. In order to highlight the advantages of adaptive ω-event privacy mechanism, we compare our Re-ADP scheme with a variant version, Re-ADP(f), which only adapts fixed ω-event privacy mechanism. Figure 7 demonstrates the comparison results in terms of MAE and QoP. It can be clearly seen that the adaptive ω mechanism can increase QoP while decreasing MAE significantly in both real-world datasets. Therefore, we can draw



(a) MAE (Bike dataset)

(b) QoP (Bike dataset)

(c) QoP (Station dataset)

(d) QoP (Station dataset)

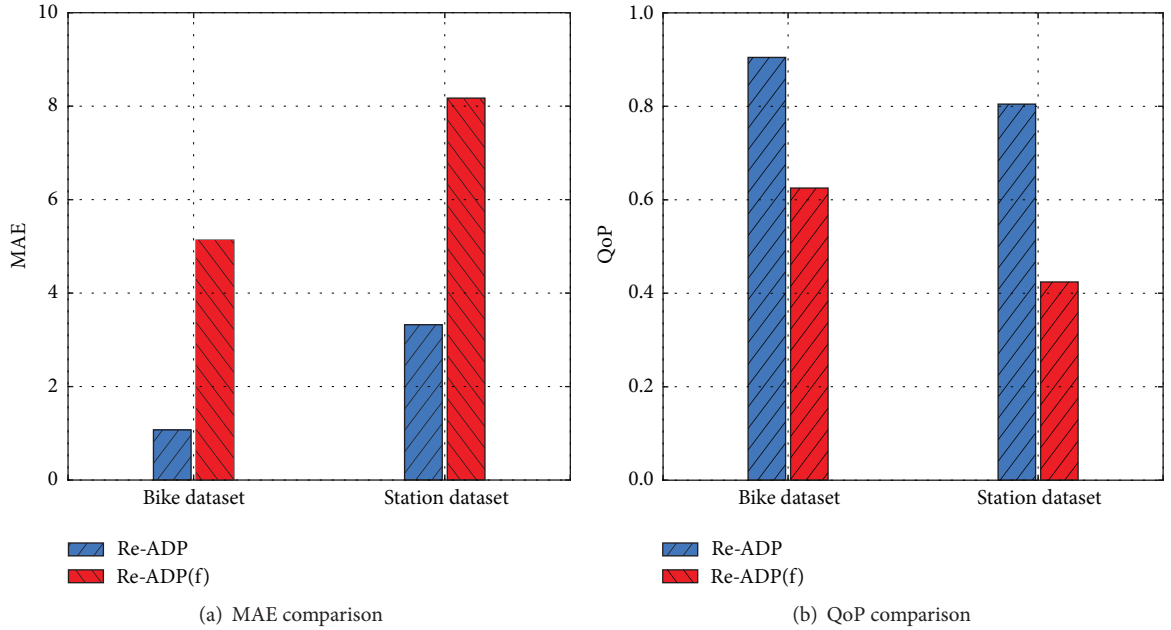
Figure 6: Utility comparison when ϵ changes.

the conclusion that the adaptive ω-event privacy mechanism advances the quality of reported data considerably.

5.2.3. Effect of Smart Grouping Mechanism. In this part, we investigate the performance of our smart grouping mechanism. As shown in Figure 8, both MAE and QoP of the smart grouping mechanism exceed the Re-ADP without the smart grouping. The excellent performance of smart grouping chiefly benefits from the K-means-based grouping algorithm and the application of the deep learning algorithm.

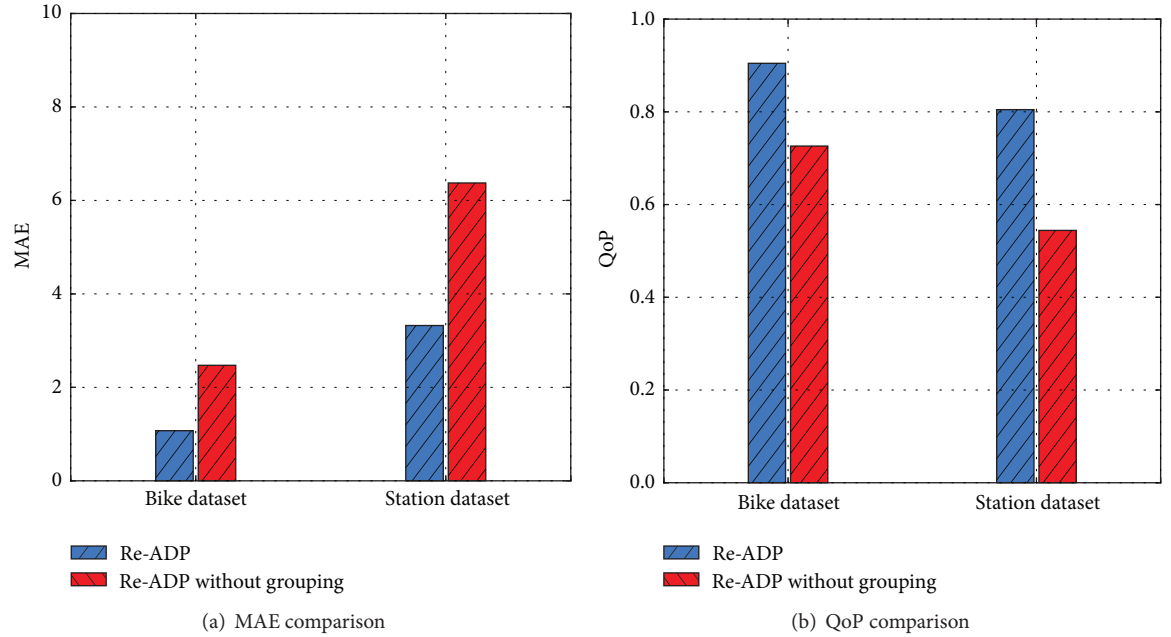
6. Related Work

Many methods have been proposed to ensure the privacy of aggregated data generated from IoT devices [29–33]. Wu et al. [34] proposed a Dynamic Trust Relationships Aware Data Privacy Protection (DTRPP) mechanism for Mobile Crowd Sensing (MCS), which evaluates the trust value of a public key ingeniously. Zhang et al. [35] designed a priority-based health data aggregation scheme (PHDA) in cloud-assisted wireless body area networks. In the scheme, a credible relay



(a) MAE comparison (b) QoP comparison

Figure 7: The performance of the adaptive ω mechanism ($\epsilon = 1$).



(a) MAE comparison (b) QoP comparison

Figure 8: The performance of the smart grouping mechanism ($\epsilon = 1$).

node can be selected according to the social relationship between nodes to help aggregation data and then forward it to cloud servers. PHAD also provides a lightweight privacy-preserving aggregation scheme, which can not only resist the forgery attack but reduce communication overhead. Li et al. [36] presented a privacy-aware data aggregation protocol for mobile sensing, which can aggregate time-series data to prevent untrustworthy aggregators from disclosing privacy. Using an additive homomorphic encryption and a novel key management scheme, the aggregator can only obtain the sum of all users data. Still, both schemes cannot cope with complex attacks that can also mine some privacy from the raw sum data.

In addition, all existing methods to achieve privacy preserving are based on encryption technologies. These complicated encryption technologies usually introduce high computation overhead, which may not be suitable for energy-constrained sensor networks. Some researchers suggest exploiting differential privacy, a convincing model for providing privacy, to protect aggregated data generated by IoT

devices. Han et al. in [37] proposed a scheme to provide privacy preserving for health data aggregation. It employs a differential privacy model to resist differential attacks that most existing data aggregation schemes have suffered from. Yang et al. in [14] also proposed a differential privacy model based on machine learning algorithms. The model can reduce communication overhead as well as protect the privacy of sensitive data rigorously for the fog computing architecture. Also in fog computing, Wang et al. [38] put forward a privacy-preserving content-based publish-subscribe scheme with differential privacy in a publish-subscribe system, which can protect against collusion attacks.

Although these works apply differential privacy to protect privacy of aggregated data, there is a serious deficiency in a real scenario. They may greatly reduce the availability of aggregated data streams. Thus, some studies are committed to solving this challenge. Cao et al. in [39] studied a protection method for sensitive streams within a window instead of the whole infinite stream. Considering window-based applications, they explored a stream-based management system to cope with numerous aggregate queries simultaneously. In [18], Fan and Xiong intended to hide all events of users and designed a user-level privacy strategy for a finite stream. For the received perturbing data, they employed the Kalman filter [40] in their strategy to improve accuracy of differentially private data release. Considering multiple events occurring at continuous time segments, Kellaris et al. presented a ω-event ϵ-differential privacy model in [17]. This model combined the advantages of event-level privacy model and user-level privacy model skillfully. In the model, they employed a sliding window to capture a wide range of ω-event privacy and designed a scheme to distribute and absorb the privacy budget on the assumption that statistics do not change significantly. On this basis, Wang and Zhang further designed an online aggregate monitoring scheme for infinite streams in [26]. Their scheme integrated adaptive sampling, a budget mechanism, and dynamic grouping and perturbation to provide privacy preserving of statistics.

Despite the fact that ongoing studies of differential privacy on streams data aggregation have played a vital role, there still exist challenges to be dealt with. We point out that the fixed sliding windows employed in most existing frameworks may not be practical. Moreover, existing metrics are only suitable for static data rather than streaming media. Motivated by these challenges, in this paper, we present a real-time privacy-preserving streams data aggregation framework based on adaptive ω-event differential privacy for the fog computing architecture.

7. Conclusion

Considering privacy disclosure of aggregated data in fog computing, we present a real-time stream data aggregation framework with adaptive ω-event differential privacy (Re-ADP) in this paper. For the four layers of our system model, this framework is composed of three components, i.e., adaptive ω-event privacy analysis, smart grouping-based perturbation, and the filtering mechanism. In particular, we can employ the first component to protect privacy of infinite stream over any successive ω time stamps. Then the second component is to achieve smart grouping based on K-means and inject additional noise into aggregated data, and we exploit an existing filter to improve data availability in the third component. Finally, we provide a theory to prove that the proposed Re-ADP framework satisfies differential privacy in theory. Extensive experiments over real-world datasets show that the Re-ADP scheme outperforms existing methods and improves the utility of real-time data publishing with strong privacy preserving.

Data Availability

The datasets used to support the findings of this study have been openly accessed. The Bike dataset can be found at https://www.capitalbikeshare.com/system-data. And the Station dataset can be accessed at https://www.kaggle.com/saulfuh/bart-ridership/data. Also, the authors have cited these datasets in the References.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant nos. 61471028 and 61571010) and the Fundamental Research Funds for the Central Universities (Grant nos. 2017JBM004 and 2016JBZ003).

References

[1] L. Zhang, Z. Cai, and X. Wang, "FakeMask: A Novel Privacy Preserving Approach for Smartphones," *IEEE Transactions on Network and Service Management*, vol. 13, no. 2, pp. 335–348, 2016.

[2] 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*, pp. 1-1, 2017.

[3] W. Zhang, Z. Zhang, and H. Chao, "Cooperative Fog Computing for Dealing with Big Data in the Internet of Vehicles: Architecture and Hierarchical Resource Management," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 60–67, 2017.

[4] 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.

[5] J. Ni, K. Zhang, X. Lin, and X. S. Shen, "Securing Fog Computing for Internet of Things Applications: Challenges and Solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 601–628, 2018.

[6] F. Tzer, "Privacy issues in vehicular ad hoc networks," in *Proceedings of the International Conference on Privacy Enhancing Technologies*, pp. 197–209, 2005.

[7] 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, article no. 475, pp. 1–23, 2017.

[8] X. Zheng, Z. Cai, J. Li, and H. Gao, "Location-privacy-aware review publication mechanism for local business service systems," in *Proceedings of the IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pp. 1–9, May 2017.

[9] 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.

[10] Y. Lu, X. Wang, C. Hu, H. Li, and Y. Huo, "A traceable threshold attribute-based signcryption for mHealthcare social network," *International Journal of Sensor Networks*, vol. 26, no. 1, pp. 43–53, 2018.

[11] M. Li, W. J. Lou, and K. Ren, "Data security and privacy in wireless body area networks," *IEEE Wireless Communications Magazine*, vol. 17, no. 1, pp. 51–58, 2010.

[12] S. Li, K. Xue, Q. Yang, and P. Hong, "PPMA: Privacy-Preserving Multisubset Data Aggregation in Smart Grid," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 2, pp. 462–471, 2018.

[13] C. Dwork, "Differential privacy," in *Proceedings of the ICALP*, pp. 1–12, 2006.

[14] M. Yang, T. Zhu, B. Liu, Y. Xiang, and W. Zhou, "Machine Learning Differential Privacy With Multifunctional Aggregation in a Fog Computing Architecture," *IEEE Access*, vol. 6, pp. 17119–17129, 2018.

[15] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of cryptography*, vol. 3876 of *Lecture Notes in Comput. Sci.*, pp. 265–284, Springer, Berlin, 2006.

[16] F. McSherry, "Privacy integrated queries: an extensible platform for privacy-preserving data analysis," *Communications of the ACM*, vol. 53, no. 9, pp. 89–97, 2010.

[17] G. Kellaris, S. Papadopoulos, X. Xiao, and D. Papadias, "Differentially private event sequences over infinite streams," in *Proceedings of the VLDB Endowment*, vol. 7, pp. 1155–1166, 2014.

[18] L. Fan and L. Xiong, "An adaptive approach to real-time aggregate monitoring with differential privacy," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 9, pp. 2094–2106, 2014.

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[20] S. Gulshad, D. Sigmund, and J.-H. Kim, "Learning to reproduce stochastic time series using stochastic LSTM," in *Proceedings of the 2017 International Joint Conference on Neural Networks, IJCNN 2017*, pp. 859–866, May 2017.

[21] S. Kaushik, A. Choudhury, N. Dasgupta, S. Natarajan, L. A. Pickett, and V. Dutt, "Using LSTMs for Predicting Patient's Expenditure on Medications," in *Proceedings of the 2017 International Conference on Machine Learning and Data Science (MLDS)*, pp. 120–127, December 2017.

[22] Z. Zhao, W. Chen, X. Wu, P. C. Chen, and J. Liu, "LSTM network: a deep learning approach for short-term traffic forecast," *IET Intelligent Transport Systems*, vol. 11, no. 2, pp. 68–75, 2017.

[23] F. J. Pineda, "Generalization of back-propagation to recurrent neural networks," *Physical Review Letters*, vol. 59, no. 19, pp. 2229–2232, 1987.

[24] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, University of California Press, Berkeley, CA, USA, 1967.

[25] D. Kifer and B. Lin, "Towards an axiomatization of statistical privacy and utility," in *Proceedings of the the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of Database Systems*, pp. 147–158, ACM, Indianapolis, Indiana, USA, June 2010.

[26] Q. Wang, Y. Zhang, X. Lu, Z. Wang, Z. Qin, and K. Ren, "RescueDP: Real-time spatio-temporal crowd-sourced data publishing with differential privacy," in *Proceedings of the IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pp. 1–9, 2016.

[27] "Capital bikeshare data," https://www.capitalbikeshare.com/system-data.

[28] "Station data," https://www.kaggle.com/saulfuh/bart-ridership/data.

[29] D. He, N. Kumar, S. Zeadally, A. Vinel, and L. T. Yang, "Efficient and Privacy-Preserving Data Aggregation Scheme for Smart Grid Against Internal Adversaries," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2411–2419, 2017.

[30] Z. You, S. Chen, and Y. Wang, "An efficient traffic data aggregation scheme for WSN based intelligent transportation systems," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 6, no. 6, pp. 1117–1129, 2015.

[31] A. Ara, M. Al-Rodhaan, Y. Tian, and A. Al-Dhelaan, "A Secure Privacy-Preserving Data Aggregation Scheme Based on Bilinear ElGamal Cryptosystem for Remote Health Monitoring Systems," *IEEE Access*, vol. 5, pp. 12601–12617, 2017.

[32] 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.

[33] X. Dong, J. Zhou, K. Alharbi, X. Lin, and Z. Cao, "An ElGamal-based efficient and privacy-preserving data aggregation scheme for smart grid," in *Proceedings of the 2014 IEEE Global Communications Conference, GLOBECOM 2014*, pp. 4720–4725, December 2014.

[34] D. Wu, S. Si, S. Wu, and R. Wang, "Dynamic Trust Relationships Aware Data Privacy Protection in Mobile Crowd-Sensing," *IEEE Internet of Things Journal*, no. 99, pp. 1-1, 2017.

[35] K. Zhang, X. Liang, M. Baura, R. Lu, and X. Shen, "PHDA: a priority based health data aggregation with privacy preservation for cloud assisted WBANs," *Information Sciences*, vol. 284, pp. 130–141, 2014.

[36] Q. Li, G. Cao, and T. F. L. Porta, "Efficient and privacy-aware data aggregation in mobile sensing," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 2, pp. 115–129, 2014.

[37] S. Han, S. Zhao, Q. Li, C.-H. Ju, and W. Zhou, "PPM-HDA: privacy-preserving and multifunctional health data aggregation with fault tolerance," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 9, pp. 1940–1955, 2016.

[38] Q. Wang, D. Chen, N. Zhang, Z. Ding, and Z. Qin, "PCP: A Privacy-Preserving Content-Based Publish-Subscribe Scheme with Differential Privacy in Fog Computing," *IEEE Access*, vol. 5, pp. 17962–17974, 2017.

[39] J. Cao, Q. Xiao, G. Ghinita, N. Li, E. Bertino, and K. Tan, "Efficient and accurate strategies for differentially-private sliding window queries," in *Proceedings of the the 16th International Conference*, pp. 191–202, ACM, Genoa, Italy, March 2013.

[40] A. Jain, E. Y. Chang, and Y.-F. Wang, "Adaptive stream resource management using kalman filters," in *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data*, pp. 11–22, ACM, 2004.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 6031715, 12 pages
https://doi.org/10.1155/2018/6031715

Research Article

A Probabilistic Privacy Preserving Strategy for Word-of-Mouth Social Networks

Tao Jing, Qiancheng Chen, and Yingkun Wen

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

Correspondence should be addressed to Qiancheng Chen; 16120048@bjtu.edu.cn

Received 3 May 2018; Accepted 21 June 2018; Published 8 July 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Tao Jing 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.

An online social network (OSN) is a platform that makes people communicate with friends, share messages, accelerate business, and enhance teamwork. In the OSN, privacy issues are increasingly concerned, especially in private message leaks in word-of-mouth. A user's privacy may be leaked out by acquaintances without user's consent. In this paper, an integrated system is designed to prevent this illegal privacy leak. In particular, we only use the method of space vector model to determine whether the user's private message is really leaked. Canary traps techniques are used to detect leakers. Then, we define a trust degree mechanism to evaluate trustworthiness of a communicator dynamically. Finally, we set up a new message publishing system to determine who can obtain the message of publisher. Secrecy performance analysis is provided to verify the effectiveness of the proposed message publishing system. Accordingly, a user in social networks can check whether other users are trustworthy before sending their private messages.

1. Introduction

Fog computing is a distributed collaborative architecture that enables specific applications or services between actual data sources and the cloud to be managed in the most efficient place [1]. This type of computing is effectively extending cloud computing capabilities and services to the edge of the network, bringing their advantages and functions to the point where data can be executed and manipulated in the closest proximity. In other words, fog computing is an extension of the concept of cloud computing. Different from cloud computing, fog computing wins by volume, emphasising quantity, no matter how weak a single computing node is. Cloud computing adjusts computing power, typically by a high performance computing device in a stack [2].

Fog computing-based typical services in online social networks (OSNs), such as Wechat, Facebook, Twitter, and LinkedIn, gradually become a primary mode to interact and communicate between participants. Each user in an OSN is a node component that makes up the social network topology. These nodes do not have strong computing and storage capabilities, but they can help with things like data transfer. Therefore, it is a reliable assumption to apply fog computing to social networks. Edge nodes in social networks are more mobile and more decentralized. The most concerned issue in social networks is privacy leaks. In the process of communication among these edge nodes, there is a novel way of privacy leakage through word-of-mouth. Word-of-mouth is a form of privacy disclosure on social networks. This kind of privacy leakage exists in a large number of real social networks but is seldom studied. For simplicity, we call it a word-of-mouth social network.

A word-of-mouth social network exists in the real human-centric world, which can pass messages from one person to another by oral communication and finally lead to the rapid spread of messages. Sometimes, the spread of private information may be peeped, misused, and taken illegally by other strangers [3]. As a result, it is important to prevent privacy disclosure caused by word-of-mouth. In fact, the disclosure degree of one user's privacy spreading is related to how to control others to access his data. In addition, it also depends on how much and what data the user wants to release. To tackle this issue, access control has been envisioned as a promising and effective approach to protect privacy of a person's account [4–6]. We intend to carefully

design a strategy among users in the word-of-mouth social network to share their data within a trusted user set.

Considering the human-centric network, our objective is to design an approach to achieve trustworthy word-of-mouth information release. This approach can trace the source of privacy disclosure and automatically adjust the trust degree of nodes in OSNs. Our exploration of this uncharted area needs to answer the following three challenges. First, how can we determine whether one user's privacy has been already disclosed and detect the leaker? Second, how to update social relationship of a user after detecting information disclosure? Finally, how to prevent acquaintances from disclosing privacy through the manner of word-of-mouth?

Aiming at the first challenges, we intend to use the vector space model (VSM) and the canary trap technology to achieve message disclosure detection. The authors of [7] present a mathematical expression of a VSM to measure the similarity of two sets. Then Li et al. improve the calculation method of the VSM in [8]. They apply semantic resources to reduce the dimensionality of feature items. Similarly, we apply the VSM in this article to calculate the similarity between the suspected leaked messages and the published ones to determine whether the messages is compromised. After determining message leakage, we employ the canary trap technology to trace the source of the leaked messages. In brief, different versions of sensitive data are sent to suspected leakers to find which version gets leaked [9].

Next, we introduce the trust degree of a user to ensure secure data sharing to cope with the second challenge. The trust mechanism has been widely studied to achieve privacy preserving in traditional OSNs. The authors of [10] investigate a recommendation belief based on a distributed trust management model for peer-to-peer networks. They quantify and evaluate the reliability of nodes and introduce a similarity function to construct the recommended credibility. In [11], the authors propose a method to check users' credibility before they enter a network. In addition, the authors in [12] design a secure recommendation system for mobile users to learn about potential friends opportunistically. Different from the existing works, we employ the dynamic trust degree to classify the recipients and to rank the sensitivity of publisher to information that needs to be published. The reason is that a user will unintentionally disclose privacy and later regret the behavior if the user is in an emotional state at the time of posting [13]. In order to avoid incorrect access to the sensitive-privacy information, the premise of access control for users is to classify the privacy information clearly.

Moreover, in order to prevent the privacy leakage, one user's (publisher's) privacy should only be shared with other ones (recipients) in a "correct" user set in OSNs [14–16]. The set, composed of numerous trusted recipients, can be updated based on the dynamic trust value that is a personal perception of a publisher to recipients. In this paper, we allow the publisher to rate recipients based on the value to determine whether they are trustworthy or not and dynamically adjust privacy settings of recipients by privacy disclosure. We design a systematic information publishing algorithm to reduce the leakage probability of private messages. In this case, we can reduce the risk of illegal spread of messages [17, 18].

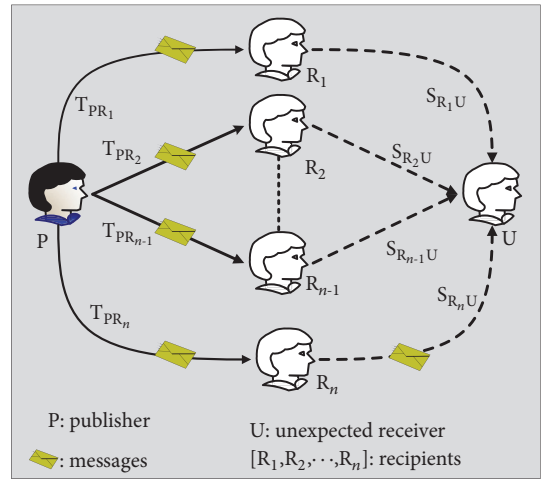


Figure 1: A privacy disclosure model in word-of-mouth OSNs.

To the best of our knowledge, there are few solutions to preserve privacy in word-of-mouth OSNs, which motivates our work. The main contributions of this paper are summarized as follows.

(i) We employ the VSM to measure the similarity between leaked messages and original ones so as to detect whether privacy disclosure exists. According to the tracing results by using the canary trap method, we determine which recipients are the possible leakers.

(ii) We define a dynamic approach to compute trust degree between users. To handle privacy leaks fairly, we design a novel utility function to punish the trust degree of leakers.

(iii) We design a probabilistic privacy preserving strategy based on a publisher's sensitivity to messages and recipients' trust degree for publishing a user's messages securely.

The rest of the paper is organized as follows. Section 2 introduces the privacy disclosure model in word-of-mouth OSNs. In Section 3, we present our privacy disclosure detection method via computing the similarity between leaked messages and original ones. Section 4 introduces the calculation method of trust degree and the punishment mechanism for leakers. Next, a trust degree based publishing system is proposed to achieve messages sharing with privacy preserving in Section 5. Then, we analyze the secure performance of our strategy in Section 6 and survey related work in Section 7. Finally, we conclude our work in Section 8.

2. System Model

In this section, we propose a privacy disclosure model as shown in Figure 1. This model can be formulated as a weighted directed social graph. Each node in the graph

denotes a user while a directed edge between two nodes represents the direction of messages transmission. We define T_{PR_i} and S_{R_iU} as the trust degree of the publisher P to its recipients R_i, $i \in [1 \cdots n]$, and the similarity between the unexpected receiver U with P's recipients, respectively. According to the graph of social relationship, we can extract the corresponding social attributes to explore a dynamic trust evaluation mechanism to achieve privacy preserving.

The privacy disclosure model in Figure 1 indicates that P is aware that its private messages are leaked to U after P sends the messages to R_i. The threat of privacy disclosure may be caused by one or more users in R_i. So the publisher is aware that his information is leaked and hoping to know which recipient sold him out. In this article, we hope to complete the tracking process of leakers. Because it is a post-evaluation method, we can only determine the probability that information will be leaked by a certain recipient. Recipients who have disclosed the privacy of the publisher should be penalized to protect the publishers' private message.

In this paper, we intend to employ the VSM algorithm to help A to determine whether the messages are really leaked out. Next, using the canary trap method to assist A to trace the possible leakers in the recipients set, and then give penalties to these leakers. (Here, what we consider is that recipients' motivation for disclosing information is not malicious. They are not trying to damage the publisher's interests. Therefore, the message disclosure should get the penalty of trust without considering legal or other punishment.) For the leakers, we dynamically adjust their trust degree to explore our privacy preserving strategy. Here, we present a novel utility function to measure the update standard of recipients' trust degree based on the centrality degree $Cen(i)$ of recipients and the similarity S_{ij} between two users. In our model, social networking platforms do not need to monitor messaging between users. The task of the platform is to help users find the leaker when they submit information detection applications.

3. Detection and Tracing Scheme for Leakers

A publisher realizes that his information may have been leaked, but he/she does not have enough confidence to determine what happened. In our model, the publisher can apply to the platform for leakage detection. The messaging platform uses the VSM to detect whether the publisher's information is actually leaked. In this section, we first compute the similarity score between suspected leaked messages and the corresponding published ones to determine whether the private message is leaked or not. The implementation of the VSM-based algorithm is actually quantifying the process by which publishers realize that privacy is compromised. In the second part, after privacy disclosure detection, we intend to employ the canary trap technology to trace privacy leakers.

3.1. Privacy Disclosure Detection. Comparing with images, files, and events, we believe that only texts leakage cannot be directly judged visually. Therefore, we only consider the private message to be a text and use the VSM method to calculate the similarity between suspected text and the original one.

In general, we can characterize text as a form of space vector based on the VSM. Therefore, the similarity between two texts can be measured by computing the similarity between two vectors. Assume that $(t_1, t_2, \ldots, t_n)$ indicate the text to be detected and $(w_1, w_2, \ldots, w_n)$ denote the corresponding coordinate values of the n-dimension space. Then, we intend to exploit the VSM to find out a score that indicates the degree of semantic equivalence between two texts. The following details describe the procedure to determine whether two texts are similar or not.

3.1.1. Text Preprocessing. First, we use the NLPIR word segmentation system to complete the word segmentation (NLPIR system is a software developed by the Institute of Computing Technology, Chinese Academy of Sciences; the principle of this system is based on the information cross entropy to automatically discover new language features and adapt to the language probability distribution model of the test corpus to realize adaptive participle), and obtain n word sets $(s_1, s_2, \ldots, s_n)$ that contain all the words that appear in the text. Next, we continue to remove the stop word that refers to the words with high frequency but no practical meaning. Such words include prepositions, adverbs, and conjunctions. They usually have no definite meaning in themselves, and only when they are placed in a complete sentence can they have a certain effect. The widespread use of a stop word in documents can easily cause interference with effective information. It is very significant to eliminate noise before feature weighting and selection. As a result, we remove all stop words in the word set s_j, $j \in [1 \cdots n]$.

3.1.2. Feature Extraction and Weight Calculation. As feature items, higher analytical accuracy of phrases and sentences may result in higher analysis error rate. In this paper, we choose words as a feature of text rather than phrases and sentences. In order to better reflect the performance difference of feature terms in the text content, we assign a weight value to each feature term. In particular, we calculate the weight of feature terms of each text segment separately. For the jth paragraph of the ith text s_{ij}, the feature term weight refers to the performance of the feature term t_k in the text segment s_{ij}. The formula for computing weights of feature terms can be indicated as follows:

$$w_{ijk} = tf_{ijk} = \frac{c_{ijk}}{l_{ij}}, \tag{1}$$

where tf_{ijk} is the frequency of the occurrence of feature term t_k in text segment s_{ij}, c_{ijk} denotes the number of feature terms t_k in the text segment s_{ij}, and l_{ij} represents the number of words contained in the text segment s_{ij}.

3.1.3. Text Vectorization. After extracting feature terms of each paragraph and assigning the corresponding weights, the text can be expressed as the form of a vector matrix. If the text D is divided into n parts and each part has m feature terms. The text D can be expressed as follows:

Input: text D
Output: the vectorization result $V[D]$ of text D
(1) $V[D] = [\overrightarrow{V_1}\overrightarrow{V_2}\cdots\overrightarrow{V_n}]^{-1}$
(2) $V_i = [x_1, x_2, \ldots, x_n]$
(3) Let section(D) be the segmentation result of the text D
(4) **For** each segmentation in section(D) **do**
(5) Word set $s_j \longleftarrow$NLPIR_segmentword(s_i);
(6) **For** each w_i in words set s_j **do**
(7) **While** stop words list contains w_i **do**
(8) Remove w_i from word set s_j
(9) T$\longleftarrow$extract features in s_j
(10) **For** each w_j in T **do**
(11) $m \longleftarrow$count(w_j)/countword(s_j)
(12) $V_i \longleftarrow mT$
(13) $V[D] \longleftarrow V[D] + V_i$
(14) Return $V[D]$

Algorithm 1: Vectorization procedure.

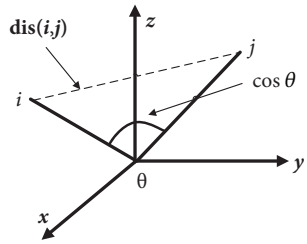


Figure 2: Cosine similarity and Euclidean distance in three dimensions.

$$D = [s_1, s_2, \ldots, s_n]^{-1} \times [t_1, t_2, \ldots, t_n] = \begin{pmatrix} w_{11} & \cdots & w_{1m} \\ \vdots & \ddots & \vdots \\ w_{n1} & \cdots & w_{nm} \end{pmatrix}. \quad (2)$$

The process of text vectorization is shown in Algorithm 1. Here, we segment paragraphs and words, remove the stop words, extract the feature terms, and calculate the weight of feature terms.

In Algorithm 1, "section(D)" denotes the processing procedure of text segmentation, "NLPIR_segmentword" represents a segmentation interface, "stop words list" represents the disabled word list defined in the text preprocessing part, "count" is a function that calculates the number of occurrences of each feature term in the text, and "countword" is a function that calculates the number of words contained in the text.

3.1.4. Similarity Measurement. Cosine similarity and Euclidean distance are the most common methods to measure similarity, as shown in Figure 2.

From Figure 2, we can see that cosine similarity measures the angle of vector space, which shows the difference of vectors in direction. Euclidean distance measures the absolute distance between two points. The cosine similarity is not sensitive to the absolute data, so it is applicable to the similarity analysis of specific features. Therefore, we take advantage of the cosine distance calculating method to measure the similarity. Intuitively, two vectors (texts) are independent or irrelevant if $Sim(i_1, i_2)$ is equal to 0, while $Sim(i_1, i_2)$ is equal to 1 when two vectors (texts) are the same. The similarity between information i_1 and information i_2 can be calculated as follows:

$$Sim(i_1, i_2) = \frac{1}{n}\sum_{j=1}^{n} \cos\theta_j = \frac{1}{n}\sum_{j=1}^{n} \frac{\overrightarrow{v_{1j}} \cdot \overrightarrow{v_{2j}}}{\left\|v_{1j}\right\| \cdot \left\|v_{2j}\right\|} = \frac{1}{n}\sum_{j=1}^{n} \frac{2\sum_{k=1}^{m_j} w_{1jk} w_{2jk}}{\sqrt{\left(\sum_{k=1}^{m_j} {w_{1jk}}^2\right)\left(\sum_{k=1}^{m_j} {w_{2jk}}^2\right)}}. \quad (3)$$

In order to verify the feasibility of our solution, we implement the algorithm using Python tools on Windows. Due to limited space, we only introduce the experimental text we selected and experimental results. We carried out vector extraction operations on three sentences shown in Table 1 and calculated the similarity between them. Finally, according to the cosine distance, we obtain the similarity between three texts, respectively, as shown in Table 1.

Table 1: The result of similarity calculation.

sentences	similarity scores
1. Julie loves me more than Linda loves me	(1,2) 0.753602532225
2. Jane likes me more than Julie loves me	(1,3) 0.128408027002
3. He likes basketball more than baseball	(2,3) 0.257423195662

We simply showed how to calculate the similarity between texts. When the publisher realized that their sensitive information had been compromised, OSNs calculate the similarity between suspicious information and target information, to determine whether publisher's information is actually compromised or not. After a lot of experiments, we concluded that when the similarity score is greater than 0.75, the meaning of the sentence is basically the same. Therefore, in this chapter, we say that the text-privacy has been leaked when the similarity score between the suspect text and the target text is greater than 0.75.

When a publisher realizes that their a sensitive message has been compromised, we would compute the similarity between a suspicious message I_s and the original message of publisher. In this case, we can determine whether publisher's privacy is actually compromised or not. The VSM approach quantifies the process of publishers' awareness of the leakage of private message. Once the publisher is aware of a privacy breach, they can adopt a canary trap approach to detect which users have recently leaked their private message.

3.2. Canary Trap Techniques. After conducting the VSM-based privacy disclosure detection, we assume that a publisher's privacy has been leaked. In order to detect leakers, it is necessary to have a strategy to determine whether or not such message is used by some user illegally. The canary trap is an

Table 2: Canary traps for different types of message.

Message types	Trapping settings
Digital images	Different watermarks
Database files	Different values of some cells
Events	Different values of some attributes
Texts	Mixture of different paragraphs

approach to detect information leakage source. The basic idea is that the publisher sends each suspect a different version of sensitive files and focuses on which version is leaked.

We consider different types of message using different traps for distinction according to the leakage message. We divide the types of message into four categories: digital images, database files, events, and texts as shown in Table 2. Accordingly, we provide four different types of message traps in our canary trap algorithm.

Recent events indicate that a user in publishers' friend circle leaks publisher's private message. After verifying the fact that private message is disclosed, we use the canary trap technique to find out leakers. First, we consider that the type of message trap is a digital image. We send an image embedded with different fingerprints to n users R_1, R_2,...,R_n. The digital fingerprint is embedded in each user's copy with a unique ID that can be extracted to help track the leaker when unauthorised leaks are found. We use digital watermarking to embed unique fingerprints in each copy of an image before releasing the image.

Definition 1 (digital watermarking). Digital watermarking technology directly inserts some identification information into the digital carrier and modifies the structure of a specific area without affecting the value of the original carrier. Digital watermarks are not easy to detect and remodify but can be identified by the manufacturer. Through these watermark information hidden in the carrier, we can achieve the purpose of confirming that the content creator and purchaser transmit secret information or determine whether the carrier has been tampered with.

For digital images represented by vector X, n recipients R_n is required, and the image owner generates unique fingerprint w_i for R_n. Watermark images will be passed to the recipient R_n that can be expressed as follows:

$$Y_i = X + JND \cdot w^i, \tag{4}$$

where Y_i is the digital image embedded with a fingerprint. JND is used to achieve the imperceptibility of embedding fingerprint in Y_i, which makes each copy unique. In this case, we can identify the leaker if dishonest users illegally repost its copy. We assume that recipients will not be able to decrypt the fingerprint by collusion, and a digital fingerprint identification system can track the leaker, as shown in Figure 3.

Secondly, for a database file, we modify data that is not important to implement different versions. What is not important here is that changing of its value does not affect the trend of the database. We need to generate n different versions of data. Supposing the number of data units that need to be changed is x, we can get $x = \log_2 n$. Similarly, once a database has been detected and leaked, we can judge the leaker.

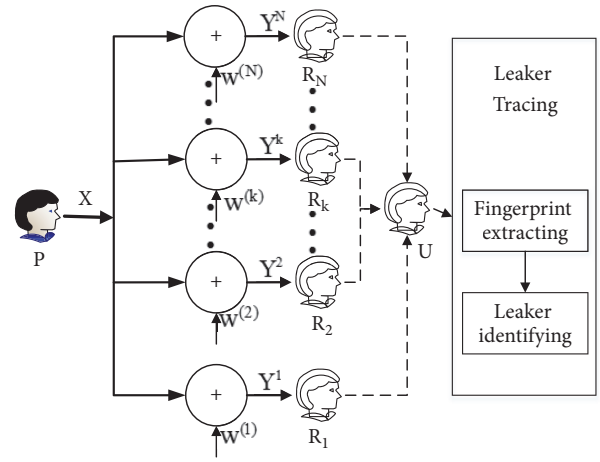


Figure 3: Digital fingerprinting system.

Thirdly, if we want to select a short event as an information trap, we modify the basic attributes like time, address, or target of the event. In short, n users R_1, R_2,...,R_n may obtain n completely different times.

Finally, we define that a text has m paragraphs if we intend to select the text as an information trap. There are six different versions of each paragraph, and the mixing of these paragraphs is unique to each numbered copy of the text. Each version has minor changes, such as the font or spacing of words used in text. Unless someone tries to find the difference, it would not be noticed. There are more than 6^m possible permutations, but the actual text has only n numbered copies (assume $6^m > n$). If someone refers to two or three paragraphs of these paragraphs, we know which copy he sees, so we know who leaked private message.

We consider two special cases of false positive and false negative. First, the false positive condition means that the leaker in the last time privacy breach event might not have been detected by this canary trap experiment. But a person with a high frequency of leaks can not protect himself from every detection. Second, we are considering another possible scenario in which although a user accidentally leaked in the canary trap we deliberately falsified information. We cannot be absolutely sure that the previously detected sensitive information must be the user who leaked information in the canary trap. This is a false negative result. However, even if the previous message is not leaked by this user, he should still be punished for his mistake.

Although this approach can only determine the leaker of privacy in a certain probability, we still want to determine the person who leaks the privacy as fair as possible. What we need is to fairly assign different levels of trust penalties to different users. Therefore, we introduce the concept of similarity and centrality to distribute the trust degree penalty value fairly. This process will be described in detail in the next section.

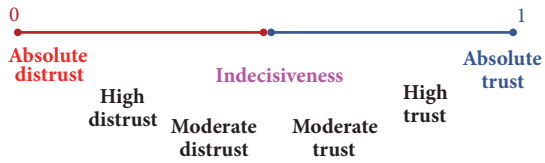


Figure 4: Trust degree range in an online social network.

4. Trust Degree Mechanism

In this section, we construct a trust degree mechanism to calculate and punish the trust degrees of the publisher to recipients. Trust acts as the glue that holds networks together, enabling networks to function effectively even though they lack a hierarchal power structure.

In the first part, we introduce a dynamic calculation approach of trust degree when there is no leakers. In the second part, we introduce how to punish the trust degree of leakers when there exist leakers.

4.1. Dynamic Calculation Approach of Trust Degree. In our mechanism, the value of trust degree ranges from 0 to 1. 0 represents absolute distrust while 1 means absolute trust; specific classification is shown in Figure 4.

The trust degree of a user to another user consists of two parts: direct trust degree (DT) and recommendation trust degree (RT). DT indicates the direct trust level of the publisher to the recipient based on the direct interaction experience. RT represents the recommendation trust and relies on rating information from the other recommender. We set P, R, and M to represent publisher, recipient, and recommender, respectively. The trust degree of the user P to the user R is defined as $T(P, R)$, which can be computed as follows:

$$T(P,R) = \alpha \times DT(P,R) + \beta \times RT(P,R), \quad (5)$$

where $DT(P, R)$ indicates the value of direct trust degree and $RT(P, R)$ indicates the value of recommendation trust degree. α and β are the trust degree regulation factors. Their values are related to the proportion that the publisher pays attention to DT and RT. Generally, we can set α and β according to the interaction frequency between P (or M) and R. Specifically, if P transacts with R more frequently than M, then we can give α a higher value and β a lower value, and vice versa. Therefore, we need to calculate DT and RT, respectively.

DT(P,R): DT between the publisher P and the recipient R can be calculated as follows.

$$DT(P,R) = \frac{\sum_{k=1}^{K} E_k(P,R) \times f_k \times w_k(P,R)}{\sum_{k=1}^{K} f_k}. \quad (6)$$

(i) We assume that the publisher P has conducted a total of K transactions in the past with the recipient R.

(ii) The evaluation value of the kth transaction is $E_k(P, R)$. $E_k(P, R)$ is provided by the user P and belongs to $[0, 1]$.

(iii) f_k is a time fading function and is defined as follow.

Definition 2 (time fading function). In order to improve the authenticity and dynamic adaptivity of direct trust, we consider that the interaction behavior of the past has attenuated the direct trust effect compared to the current interaction behavior. Specifically, compared with the Kth current transaction, the importance of the kth transaction in the past same transactions is depreciated. We define this attenuated function as time fading function that can be calculated as follows.

$$f_k = \delta^{K-k}, \quad 0 < \delta < 1, \ 1 \le k \le K \quad (7)$$

where $f_k = 1$ is previous interaction without attenuation and $f_1 = \delta^{K-1}$ is the first interaction with the largest attenuation.

(4) Last, weight of the interaction behavior of the kth transaction is represented as $w_k(P, R)$. The interaction behavior is defined by user P and can be divided into five grades according to the magnitude of behavior. We assign different weights to different interactions, which can distinguish the effect of different interactions on trust to a certain extent. Weights of each grade from big to small are 1, 0.8, 0.6, 0.4, 0.2. Therefore, the direct trust degree $DT(P, R)$ of the $(K + 1)$th interaction can be calculated by (6).

RT(P,R): RT of user P to user R is converged by P for the direct trust of the all recommendation users to R. Here, RT is a comprehensive evaluation for R by all users who have been interacted with R, which represents the overall credibility of R in social networks. So we define the value of RT of user P to user R as

$$RT(P,R) = \frac{\sum_{M \in G} DT(M,R) \times C_{PM}}{\sum_{M \in G} C_{PM}}, \quad C_{PM} \ge \Theta \quad (8)$$

where $RT(P, R)$ is the recommendation trust degree of recipients, C_{PM} is the credibility of P put to M, and G represents a collection of all trusted recommended users. Value of C_{PM} is equal to the direct trust $DT(P, M)$ of publisher P to recommender M. In this process of calculating recommendation trust, the publisher needs to provide a constant threshold Θ. The publisher P adopts recommendation information only when trustworthiness $C_{P,M}$ of the recommender is greater than Θ.

4.2. Trust Degree Punishment of Leakers. In the previous section, we detect private message leakers through the canary trap techniques. In this section, we punish these leakers by reducing the trust degrees of them. We punish trust degree of leakers based on a utility function. The utility function is integrated by the similarity between recipients and unexpected receiver with the centrality of the recipients. Therefore, we introduce two concepts: the similarity between two users and the centrality of a user.

Definition 3 (similarity). The similarity indicates the degree of separation between two users. It can be calculated by the amount of common friends in social networks. Sociologists have found that if two people have one or more friends, they have a greater chance of knowing and meeting each other.

Definition 4 (centrality). The centrality of a user is the index of the relative importance of quantized user in social networks. There are many manners to define the centrality of the user, such as betweenness centrality, closeness centrality, and degree centrality. Among these manners, we apply the direct centrality calculating approach that is defined as the number of other users in direct contact with the user to calculate similarity.

Accordingly, the centrality of user i can be expressed as follows.

$$Cen_i(\tau) = \frac{\sum_{k=1}^{N} d_{ik}(\tau)}{N}, \tag{9}$$

where $d_{ik}(\tau) = 1$ or $d_{ik}(\tau) = 0$ indicates whether there is a connection or not between the user i and the user k at time τ. N is the number of users in the network. The similarity between two users can be derived as follows.

$$S_{i,j}(\tau) = 1 + \left|F_i(\tau) \cap F_j(\tau)\right|, \tag{10}$$

where $F_i(\tau)(F_j(\tau))$ denotes the set of friends of user $i(j)$ at time τ. When we compute the utility function below, we convolve with the centrality and the similarity. If the similarity is equal to zero, the utility function makes no sense. However, when two users have no mutual friends, the similarity between them is equal to zero. So we add 1 in (10).

We apply the utility function as the standard metrics to punish the recipient who spread publisher's privacy information. If leakers who make mistakes in the canary trap are friends of the unexpected receiver U, we calculate the social similarity between leakers and the user U by comparing their friendship lists. We also calculate the social centrality of leakers. Note that social similarity and social centrality only reflect the characteristics of the network structure. In addition, we also need to consider the dynamics of social networks. Because of the dynamic change of nodes, the comprehensive utility should be a time-varying function. To address dynamic characteristics and avoid cumulative effects, we define the utility function as the convolution of similarity and centrality with a factor decaying as time. The total utility function value is the convolution of $Cen_R(\tau)$ with $S_{R,U}(\tau)$. $Cen_R(\tau)$ is the centrality of the recipient and $S_{R,U}(\tau)$ is the similarity between the recipient R and the unexpected receiver U.

$$\begin{aligned} Y_{R,U}(T) &= S_{R,U}(T) \otimes \frac{1 - Cen_R(\tau)}{T} \\ &= \int_{\tau=0}^{T} S_{R,U}(\tau) \cdot \frac{1 - Cen_R(\tau)}{T - \tau}, \end{aligned} \tag{11}$$

where the convolution operation provides a time-decaying description of all prior values of similarity and centrality. And $Y_{R,U}(T)$ is updated each time by accumulation of similarity and centrality when a leak event occurs.

In the previous section, we can get recipients who make mistakes in the canary trap. Now, we punish them by reducing their trust degree values. Specific penalties scale depends on the utility function value of a leaker l_i. The higher the degree of utility function, the more the value of trust degree is reduced. Above we have proved that the higher the utility function of the leaker, the greater the probability of leaking information. Therefore, we calculate the value of the trust penalty to a leaker based on the ratio of the utility function of the leaker and the sum of all utility function value. We define this specific measure of punishment as follows:

$$T_{sub}(l_i) = \frac{Y_{l_i,U}(T)}{\sum_{i=1}^{L} Y_{l_i,U}(T)} * T_{sub}, \tag{12}$$

where $T_{sub}(l_i)$ is a specific penalty in trust degree of leaker l_i and L is a collection of leakers who make mistakes in canary traps. T_{sub} is a total attenuation in trust degree caused by a leak event of privacy information, and this value is determined on the publisher. After attenuation, the trust degree of a leaker can be calculated as follows.

$$T(P, l_i)' = T(P, l_i) - T_{sub}(l_i). \tag{13}$$

The utility function is proportional to the similarity between the leaker and the unexpected receiver, and inversely proportional to the centrality of the leaker. A leaker with a high utility function will be penalized with a high trust value. But the total trust penalty will be equal to the publisher's expectation of a trust penalty.

$$\sum_{i=1}^{|L|} T_{sub}(l_i) = T_{sub}. \tag{14}$$

Users who make more mistakes in canary traps will be punished with more trust values. Users who receive a trust penalty are likely to fall into lower security classes (this concept is described in the next section). So recipients who leak users' privacy too often will be less likely to receive private information from publishers.

So far, we complete the calculation process and punishment process of the trust degree mechanism. In the first part, we propose a calculation approach of trust degree in the case of no leakers. In the calculation approach of trust degree, the interaction between users can help to enhance the trust between users. We comprehensively take into account the direct trust of the publisher to the recipient and the comprehensive evaluation of the recipient by other users around the social network. This approach of trust calculation takes into account the difference in degree of influence of interaction between different time periods. Every interaction between users changes the value of trust degree. The recommendation trust of users also directly changes the publisher's trust in the recipient. In the second part, we demonstrate how to punish the trust degree of leakers in the case of existing leakers. Compared with the traditional method of trust degree mechanism, we has a better real-time and dynamic trust degree mechanism.

5. Design of Message Publishing System

In this section, we design a message publishing system to reduce the risk that publishers' private message would be

Input: Publisher's weighted social graph;
Output: Publishing rules;
(1) **Phase I:** The publisher classifies the sensitivity of the message;
(2) Let I be the message that user want to be published;
(3) $S_I \longleftarrow sensitivity(I)$;
(4) Execute sensitivity classification after each publishing.
(5) **Phase II:** Divide the secure level of the recipient;
(6)Let R be the recipient;
(7) $SL(R) = X, X \in [1, 2, 3, 4, 5]$.
(8) **Phase III:** Determine whether to share message I with R or not;
(9) **if** $S_I > SL(R)$ **then**
(10) **Return:** Message I can be sent to recipient R;
(11)
(12) **else**
(13) **Return:** Message I can not be sent to recipient R;
(14)
(15) **end**

Algorithm 2: Rating scheme.

compromised. First, we propose a rating scheme to classify security level of recipients and sensitivity of publisher. Next, we introduce our message publishing system model formally and analyze the work flow of the system.

5.1. Rating Scheme. We propose a rating scheme based on the trust degree of a publisher to recipients and publisher's sensitivity to message. The rating scheme protects the private message of the publisher in OSNs.

Definition 5 (publisher sensitivity rating). Rating of the sensitivity of a publisher is based on the fact that different publishers have different sensitivities. Lower sensitivity of a publisher has relatively lower demand for privacy protection while higher sensitivity of a publisher requires a strong protection. We divide the sensitivity of the publisher on specific message into five levels, expressed as $S_i(S_i = 1, 2, 3, 4, 5)$. Level 5 is the lowest sensitivity and level 1 is the highest sensitivity. Since the publisher may have different sensitivity requirements for information at different time, it is necessary to perform sensitivity rating at each publish request.

Definition 6 (the secure level rating). A publisher needs to evaluate the trust degree of recipients that they want to interact with. According to the trust degree mechanism proposed in the previous section, the publisher can obtain the trust degree of all recipients including leakers. The secure level (SL) of recipients is divided into five levels according to the trust degree $T(P, R)'$. We quantify this trust degree rating process, and the secure level is upgraded from 1 to 5, as shown in Table 3.

In Algorithm 2, we obtain the publisher's request level for message sensitivity and classify the recipient's secure level. Considering message sensitivity and the secure level of a recipient, the proposed algorithm can help the publisher determine whether to share their private message with recipients or not..

Table 3: Security level of receivers in OSNs.

Security level	Trust level	Value of trust degree
1	Absolute trust	0.8 to 1
2	High trust	0.6 to 0.8
3	Indecisiveness	0.4 to 0.6
4	High distrust	0.2 to 0.4
5	Absolute distrust	0 to 0.2

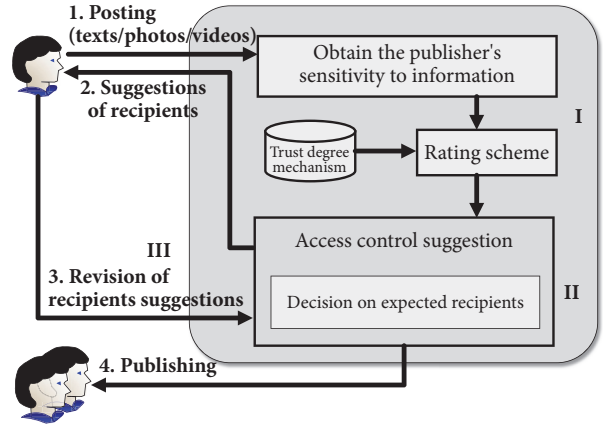


Figure 5: Overview of message publishing system.

5.2. System Overview. In order to prevent the leakage of privacy in OSNs, it is necessary to develop guidelines to determine the message publishing system. The system is exploited before a user posts on a social network site. The system determines who should be allowed to access messages and inform the user of the recommendation. In addition, users can add or exclude their own access to the proposed nonproprietary information.

The overview of the proposed system is shown as in Figure 5. It consists of three components: the rating scheme

module, the access control suggestion module, and the revision of recipient suggestion module. We describe these modules as below.

(i) **Rating scheme module:** A user needs to provide their sensitivity to the post while applying for publishing a post, because different users have different sensitivity to different messages. The system then queries the recipient's trust degree to build their secure levels. In this way, the system obtains the sensitivity level of the publisher to the messages and the secure level of the recipient. Then, the system can compare these values and formulate a access control suggestion for the next module.

(ii) **Access control suggestion module:** According to the level of disclosure determined in the previous step, this module determines the intended recipient based on the level of trustworthiness in the self-network of OSNs. We take into account the situation where the user is expected to send but the level of trust degree does not meet the requirements. We also take into account the situation where the level of trust degree meets the requirements but the user is not expected to send. During this procedure, it considers the revised messages associated with the previous posting and adds or excludes the intended recipient.

(iii) **Recipient suggestion revision module.** This module informs the publisher about the associated access control suggestion (analysis results). The access control suggestion contains a list of recipients. When a publisher gets a list of recipients, he/she can choose whether to publish information to the recipients in the list or revise the recipients in the list. The authorization/revision messages are stored in the system and would be considered in future access control recommendations.

The proposed message publishing system can help the publisher to choose the right recipients to access these messages. Because our system can prevent recipients who may leak privacy of the publisher from receiving messages, the risk of privacy leakage for the publisher can be reduced effectively.

6. Security Performance Analysis

In this section, we study the secure performance in terms of attack analysis, secrecy analysis, and access control analysis.

6.1. Attack Analysis. The leaking model we proposed indicates that recipients may spread privacy to other friends or strangers without user's consent. According to the privacy disclosure model in word-of-mouth OSNs shown in Figure 1, P shares its private messages to its recipients R_i, $i \in [1 \cdots n]$, but it is aware that the messages are known by U after a period of time. Users who have disclosed the private messages can probably be these recipients, and this manner of privacy disclosure is known as word-of-mouth.

By calculating the similarity between a suspicious message and the original one of publisher, we exploit the VSM to help P know whether its privacy is disclosed. Then, a canary trap technique is used to trace the source of disclosure. In this technique, we consider a possible scenario that we cannot be absolutely sure that the previously detected sensitive information must be the user who leaked in the canary trap, because the recipient leaked the information in this canary trap does not mean that he also leaked the previous privacy of the publisher. However, even if the previous messages are not leaked by the user, it should still be punished for its mistake. What we need is to fairly assign different levels of trust penalties to different users. As a result, we combine the centrality degree $Cen(R)$ of recipient R with the similarity S_{RU} between R and unexpected receiver U to obtain a utility function. The specific penalties scale depends on the utility function value of the leaker. Intuitively, the higher the degree of the utility function is, the more the value of trust degree is reduced. According to the dynamic evaluation of recipients' trustworthiness, decreasing of the recipient's trust degree directly affects the ability to obtain private messages. Finally, we set up a new message publishing system to determine who can obtain private messages.

6.2. Secrecy Analysis. According to the sensitivity degree of a publisher on private messages, our privacy preserving strategy first classifies users into different sensitive levels. Next, the publisher computes and classifies the trust degree of recipients. Finally, sensitivity rating and the secure level of the publisher are used to determine whether a recipient is allowed to obtain publisher's private messages or not. The trust-based privacy preserving strategy may change along with time, publisher itself, and other factors. In addition, the process may start by issuing an interactive request from the publisher to the recipients, so that the recipients cannot access a message without authorization.

6.3. Access Control Analysis. When a user in an online social network is ready to publish private messages, it considers that the private messages can be known only by a small group of its recipients rather than by random strangers. The proposed message publishing system can help the publisher to choose the right recipients to access these messages. Our system requires users to submit their own sensitivity levels to the post before applying for a post, because different users have different sensitivity to different messages. After the publisher sensitivity level acquisition process is complete, we count the secure level of the publisher's recipients through our dynamic trust degree mechanism and then give the publisher a list of suggested recipients of messages. The publisher can make revisions to the recipient list to delete or add and give feedback about the revision information to the access control system. This feedback also would be considered in future published access control recommendations. Because our system can prevent recipients who may leak privacy of the publisher from receiving messages, the risk of privacy leakage for the publisher can be reduced effectively.

7. Related Work

A service of online social networks (OSNs), such as Wechat, Facebook, Twitter, and LinkedIn, gradually becomes a

primary mode to interact and communicate between participants. A person can use these social applications at any time to contact with friends and send messages regardless of age, gender, and even socioeconomic status [19–21]. These messages should contain sensitive and private information, e.g., location [22], channel state information [23], routing information [24], social relationships [25], browsing data of Internet [26], health data [27], and financial transactions [28]. Obviously, the person should be worried about disclosures of personal information, which may be harmful to him either in virtual or real world [13, 29, 30]. As a result, an efficient privacy preserving strategy should be investigated to detect these threats [31].

The existing works in privacy preserving focus on information disclosure caused by malicious nodes (e.g., attackers or eavesdroppers) in OSNs [32, 33]. Essentially speaking, the authors of these works first convert nodes and the corresponding links in OSNs into vertices and edges of a weighted graph and then exploit graph theory and cryptography technologies to develop various protection solutions [34–36]. Although these solutions can protect personal information from network attacking and illegal eavesdropping, there still exists a more serious threat to users' privacy, i.e., information disclosure by word-of-mouth.

Text similarity calculation has been widely used in Internet search engine [37], intelligent question and answer, machine translation, information filtering, and information retrieval. In this paper, we use the text similarity calculation to determine whether the publisher's text information is leaked. In terms of algorithm, the most commonly used VSM in text similarity calculation was first proposed by Gerard Salton and McGill in 1969 [38]. The basic idea of the algorithm is to map the document to an n-dimensional vector, so as to transform the processing of text into a vector operation on a spatial vector. The similarity between documents is determined by comparing the relations between vectors. Among them, the most widely used weight calculation method is TF-IDF algorithm [39] and various improved algorithms. The most commonly used similarity measurement method is cosine similarity measurement [40].

To sum up, the privacy protection strategy in social network can be divided into two ways: role access control and data anonymity. The anonymous method is mainly used for multidimensional data such as network topology. For specific privacy such as user attributes, access control based on trust is a reliable way to protect privacy.

8. Conclusion

Considering privacy word-of-mouth disclosure by acquaintances, we put forward a novel privacy preserving strategy in this paper. In particular, we carefully combine the privacy leaking detection with the trust degree mechanism. The traditional privacy protection schemes are mainly on computing a trust degree threshold. In their schemes, a friend whose trust degree exceeds the threshold is considered believable. In contrast to these schemes, our strategy incorporates two new points of classifying each publisher's sensitivity and each ready interactive recipient. Our proposed publishing system consists of publisher's sensitivity level to information rating and recipient's secure level rating. What is noteworthy is that our scheme still gives the decision of message release to the publisher after giving the suggestion of access control. The publisher can make their own changes to the recipient list, and these revisions also will be considered in future published access control recommendations. Therefore, our information publishing strategy greatly reduces the risk of illegal spread of user's private messages.

In future, we want to provide a solution that is intended to prevent the recipient from illegally forwarding the message itself rather than the content of the message. Also, we may explore a sensitive message transmission interface for OSNs applications to protect users' privacy.

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 work was supported by the National Natural Science Foundation of China (Grants nos. 61471028, 61571010, and 61572070) and the Fundamental Research Funds for the Central Universities (Grants nos. 2017JBM004 and 2016JBZ003).

References

[1] J. Barmagen, "Fog computing: introduction to a new cloud evolution," *Jos Francisco Fornis Casals*, pp. 111–126, 2013.

[2] H. R. Arkian, A. Diyanat, and A. Pourkhalili, "MIST: Fog-based data analytics scheme with cost-efficient resource provisioning for IoT crowdsensing applications," *Journal of Network and Computer Applications*, vol. 82, pp. 152–165, 2017.

[3] 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*, 2016.

[4] H. Yiliang, J. Di, and Y. Xiaoyuan, "The revocable attribute based encryption scheme for social networks," in *Proceedings of the International Symposium on Security and Privacy in Social Networks and Big Data, SocialSec 2015*, pp. 44–51, chn, November 2015.

[5] S. Machida, T. Kajiyama, S. Shigeru, and I. Echizen, "Analysis of Facebook Friends Using Disclosure Level," in *Proceedings of the 2014 Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, pp. 471–474, Kitakyushu, Japan, August 2014.

[6] S. Machida, T. Kajiyama, S. Shigeru, and I. Echizen, "Analysis of facebook friends using disclosure level," in *Proceedings of the 10th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP 2014*, pp. 471–474, jpn, August 2014.

[7] M. Suzuki, N. Yamagishi, T. Ishidat, M. Gotot, and S. Hirasawa, "On a new model for automatic text categorization based on

vector space model," in *Proceedings of the 2010 IEEE International Conference on Systems, Man and Cybernetics, SMC 2010*, pp. 3152–3159, tur, October 2010.

[8] L. Xu, S. Sun, and Q. Wang, "Text similarity algorithm based on semantic vector space model," in *Proceedings of the 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pp. 1–4, Okayama, Japan, June 2016.

[9] F. P. Miller, A. F. Vandome, and J. Mcbrewster, "Canary Trap," in *1em plus 0.5em minus 0*, 4em Alphascript Publishing, 2010.

[10] J. Hu, Q. Wu, and B. Zhou, "RBTrust: A Recommendation Belief Based Distributed Trust Management Model for P2P Networks," in *Proceedings of the 2008 10th IEEE International Conference on High Performance Computing and Communications (HPCC)*, pp. 950–957, Dalian, China, September 2008.

[11] P. Yadav, S. Gupta, and S. Venkatesan, "Trust model for privacy in social networking using probabilistic determination," in *Proceedings of the 2014 4th International Conference on Recent Trends in Information Technology, ICRTIT 2014*, ind, April 2014.

[12] Y. He, F. Li, B. Niu, and J. Hua, "Achieving secure and accurate friend discovery based on friend-of-friend's recommendations," in *Proceedings of the 2016 IEEE International Conference on Communications, ICC 2016*, mys, May 2016.

[13] Y. Wang, G. Norcie, S. Komanduri, A. Acquisti, P. G. Leon, and L. F. Cranor, ""I regretted the minute I pressed share"," in *Proceedings of the the Seventh Symposium*, p. 1, Pittsburgh, Pennsylvania, July 2011.

[14] M. Gambhir, M. N. Doja, and Moinuddin, "Action-based trust computation algorithm for online social network," in *Proceedings of the 4th International Conference on Advanced Computing and Communication Technologies, ACCT 2014*, pp. 451–458, ind, February 2014.

[15] F. Nagle and L. Singh, "Can friends be trusted? Exploring privacy in online social networks," in *Proceedings of the 2009 International Conference on Advances in Social Network Analysis and Mining, ASONAM 2009*, pp. 312–315, grc, July 2009.

[16] Y. Yustiawan, W. Maharani, and A. A. Gozali, "Degree Centrality for Social Network with Opsahl Method," in *Proceedings of the 1st International Conference on Computer Science and Computational Intelligence, ICCSCI 2015*, pp. 419–426, idn, August 2015.

[17] A. Pandey, A. Irfan, K. Kumar, and S. Venkatesan, "Computing Privacy Risk and Trustworthiness of Users in SNSs," in *Proceedings of the 5th International Conference on Advances in Computing and Communications, ICACC 2015*, pp. 145–150, ind, September 2015.

[18] F. Riquelme and P. González-Cantergiani, "Measuring user influence on Twitter: a survey," *Information Processing & Management*, vol. 52, no. 5, pp. 949–975, 2016.

[19] Z.-J. M. Shen, "Integrated supply chain design models: a survey and future research directions," *Journal of Industrial and Management Optimization*, vol. 3, no. 1, pp. 1–27, 2007.

[20] A. M. Vegni and V. Loscrí, "A survey on vehicular social networks," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, article no. A3, pp. 2397–2419, 2015.

[21] J. Heidemann, M. Klier, and F. Probst, "Online social networks: a survey of a global phenomenon," *Computer Networks*, vol. 56, no. 18, pp. 3866–3878, 2012.

[22] 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.

[23] 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.

[24] M. Wang, J. Liu, J. Mao, H. Cheng, J. Chen, and C. Qi, "RouteGuardian: Constructing," *Tsinghua Science and Technology*, vol. 22, no. 4, pp. 400–412, 2017.

[25] X. Zheng, G. Luo, and Z. Cai, "A Fair Mechanism for Private Data Publication in Online Social Networks," *IEEE Transactions on Network Science and Engineering*, pp. 1-1.

[26] J. Mao, W. Tian, P. Li, T. Wei, and Z. Liang, "Phishing-Alarm: Robust and Efficient Phishing Detection via Page Component Similarity," *IEEE Access*, vol. 5, pp. 17020–17030, 2017.

[27] 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.

[28] H. Alhazmi, S. S. Gokhale, and D. Doran, "Understanding social effects in online networks," in *Proceedings of the 2015 International Conference on Computing, Networking and Communications, ICNC 2015*, pp. 863–868, usa, February 2015.

[29] M. Fire, R. Goldschmidt, and Y. Elovici, "Online social networks: Threats and solutions," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 2019–2036, 2014.

[30] M. Sleeper, J. Cranshaw, P. G. Kelley et al., ""I read my Twitter the next morning and was astonished" a conversational perspective on Twitter regrets," in *Proceedings of the 31st Annual CHI Conference on Human Factors in Computing Systems: Changing Perspectives, CHI 2013*, pp. 3277–3286, fra, May 2013.

[31] W. Dong, V. Dave, L. Qiu, and Y. Zhang, "Secure friend discovery in mobile social networks," in *Proceedings of the IEEE INFOCOM*, pp. 1647–1655, April 2011.

[32] C. Akcora, B. Carminati, and E. Ferrari, "Privacy in social networks: How risky is your social graph?" in *Proceedings of the IEEE 28th International Conference on Data Engineering, ICDE 2012*, pp. 9–19, usa, April 2012.

[33] B. Zhou and J. Pei, "Preserving privacy in social networks against neighborhood attacks," in *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering, ICDE'08*, pp. 506–515, mex, April 2008.

[34] Q. Liu, G. Wang, F. Li, S. Yang, and J. Wu, "Preserving Privacy with Probabilistic Indistinguishability in Weighted Social Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 5, pp. 1417–1429, 2017.

[35] L. Zhang, X.-Y. Li, K. Liu, T. Jung, and Y. Liu, "Message in a Sealed Bottle: Privacy Preserving Friending in Mobile Social Networks," *IEEE Transactions on Mobile Computing*, vol. 14, no. 9, pp. 1888–1902, 2015.

[36] R. Zhang, J. Zhang, Y. Zhang, J. Sun, and G. Yan, "Privacy-preserving profile matching for proximity-based mobile social networking," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 9, pp. 656–668, 2013.

[37] H. J. Zeng, Q. C. He, Z. Chen, W. Y. Ma, and J. Ma, "Learning to cluster web search results," in *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 210–217, July 2004.

[38] Salton and Buckley, "Team weighting approaches in automatic text retireval, readings in information retrieval," in *Proceedings of the in International Conference on Computer Vision Theory and Applications*, pp. 652–657, 1998.

[39] T. Peng, L. Liu, and W. Zuo, "PU text classification enhanced by term frequency-inverse document frequency-improved weighting," *Concurrency and Computation: Practice and Experience*, vol. 26, no. 3, pp. 728–741, 2014.

[40] V. Oleshchuk and A. Pedersen, "Ontology based semantic similarity comparison of documents," in *Proceedings of the 14th International Workshop on Database and Expert Systems Applications, DEXA 2003*, pp. 735–738, cze, September 2003.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 3173067, 9 pages
https://doi.org/10.1155/2018/3173067

Research Article

Fog Computing-Based Differential Positioning Method for BDS

Lina Wang [1,2,3] **and Linlin Li** [1]

[1] *School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China*
[2] *Beijing Engineering and Technology Research Center for Convergence Networks and Ubiquitous Services, University of Science and Technology Beijing, Beijing 100083, China*
[3] *Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing, China*

Correspondence should be addressed to Lina Wang; wln_ustb@126.com

Received 16 April 2018; Accepted 6 June 2018; Published 5 July 2018

Academic Editor: Ke Xiong

Copyright © 2018 Lina Wang and Linlin Li. 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 one of the four global satellite navigation and positioning systems, BeiDou satellite navigation system (BDS) has received increasingly more attention. The differential positioning technology of BDS has greatly enhanced its accuracy and meets the needs of high-precision applications, but its positioning time still has much room for improvement. Fog computing allows the use of its services with low latency and mobility support to make up for the disadvantages of differential positioning algorithm. The paper proposes the fog computing-based differential positioning (FCDP) method which introduces fog computing technology to BDS. Compared with the original data center-based differential positioning (DCDP) method, the simulation results demonstrate that the FCDP method decreases the latency of positioning, while assuring the positioning accuracy.

1. Introduction

BeiDou satellite navigation system (BDS) is a self-developed and independently operated global satellite navigation and positioning system in China [1–4]. In recent years, the industrialization of BeiDou civil application and its promotion work have achieved remarkable results with far-reaching effects [4]. Since 2012, when the system officially serves the Asia Pacific region, the BDS has been applied in various fields, including transportation, marine fisheries, hydrological monitoring, and weather forecasting [5]. As BDS plays an important role in many fields, it is very important to provide fast and accurate positioning algorithm to improve the system competitiveness and promote the development of the system.

Due to eliminating some common errors in the system, the differential positioning technology has been commonly used in positioning tasks and applications with higher accuracy requirements [6–8]. With this technology, all the differential data of base stations are sent to the data center and then return to mobile station after calculation. Due to the existence of the data center, the differential positioning method provides multilevel authentication for users and improves the computing ability [9, 10]. Despite the fact that the data center-based differential positioning (DCDP) method provides many benefits, there are still some challenges such as increases in delay, the waste of bandwidth, and privacy concerns, which is not good for quick positioning and the development of BDS [11]. In addition, with the development of the BDS, the construction of the base station will surely increase construction efforts to keep pace with the needs of the business, and the scalability of the DCDP system will be limited by the performance of the data center. Once the data center is attacked, the overall system performance will be greatly reduced.

To reduce the positioning time and solve all of the above problems caused by the data center, many scholars have conducted studies on algorithms or methods [12–15]. The paper [12] proposed an algorithm based on congestion control, i.e., Random Early Detection (RED) gateways, which could avoid unnecessary queue delay, while the general reduction in delay does not offer the best results. The paper [13] proposed the cutting payload (CP) mechanism to process timeouts but the CP mechanism is limited to hardware conditions. The

paper [14] examined the effectiveness of alternative TCP and Ethernet-level strategies in mitigating the TCP throughput collapse but at the cost of seriously increasing the input costs. The paper [15] proposed a global data transfer scheduler at the application layer but it will destroy synchronization.

The above approaches are for congestion, queuing, and other issues caused by the data center in order to provide improvement. However, the centralized mode is still adopted and does not fundamentally solve the potential problem caused by bidirectional transmission, especially the transmission delay caused by the uplink and downlink. In other words, due to the constraint of triangle inequalities, the link delay through the intermediate node forwarding is greater than the direct link delay between two points. To resolve the disadvantages of the data center, fog computing has recently emerged. Fog computing is considered as a cloud server operating at the edge of the network, offering special services that require network context information, location awareness, and ultra-low latency [16].

Since the features of fog computing could satisfy the demands of BDS exactly, the paper integrates fog computing to BDS for quicker and even real-time positioning processing and proposes a fog computing-based differential positioning (FCDP) method to offload computing tasks and decrease delay. Meanwhile, the original high-precision feature of the DCDP is not affected.

The reminder of this paper is organized as follows. Section 2 introduces the current differential positioning model and DCDP method of BDS. Section 3 describes the FCDP proposed by this paper and formulates the calculation progress. Section 4 presents the experimental results and analyses. Finally, Section 5 concludes the paper.

2. The Carrier Phase Differential Positioning Principle with the Data Center

According to the information used, differential positioning algorithm can be divided into four types: position differential positioning, pseudo-range differential positioning, phase smooth pseudo-range differential positioning, and carrier phase differential positioning [17]. Among them, carrier phase measurements of carrier phase differential positioning allow the measurement of short baselines with an inaccuracy as low as a few centimeters [18]. Therefore, the paper adopts the carrier phase difference as an example to introduce the differential positioning principle.

There are two types of receivers: base station receiver and mobile receiver. The base station is used as a benchmark to correct the common error between mobile receiver and base receiver. The mobile receiver is usually used as a target to locate devices such as mobile phone or the user's other device.

The base station and mobile receiver simultaneously observe the ephemeris information of BeiDou satellites. The observation equations of satellite j are as follows [19]:

$$\begin{aligned}\rho_b^j &= \lambda\left(N_b^j + \varphi_b^j\right) - c\left(d\tau_b - d\tau^j\right) - d\rho_b^j + dI_b^j - dT_b^j \\ \rho_u^j &= \lambda\left(N_u^j + \varphi_u^j\right) - c\left(d\tau_u - d\tau^j\right) - d\rho_u^j + dI_u^j - dT_u^j\end{aligned} \tag{1}$$

Here, ρ_*^j represents the real distance between the satellite j and the receiver; λ is the wavelength of carrier; N_*^j denotes the number of circumference between receiver and satellite j; φ_*^j is the period that is less than one circumference; c is the speed of the electromagnetic wave; $d\tau_*$ is the clock deviation of the receiver; $d\tau^j$ is the deviation of satellite j; $d\rho_*^j$ is the error caused by satellite calendar; dI_*^j is ionosphere error; and dT_*^j is tropospheric error.

When the distance between base station and the mobile station is of short or medium baseline (less than 100km), $d\rho_b^j \approx d\rho_u^j$, $dI_b^j \approx dI_u^j$, $dT_b^j \approx dT_u^j$ [20]. Therefore,

$$\rho_u^j - \rho_b^j = \lambda\left(\varphi_u^j - \varphi_b^j\right) - c\left(d\tau_u - d\tau_b\right) + \lambda\left(N_u^j - N_b^j\right) \tag{2}$$

$\Delta(\bullet)$ is ordered to replace the single difference between the base station and the mobile station, and then

$$\Delta\rho_{ub}^j = \lambda\Delta\varphi_{ub}^j - c\Delta d\tau_{ub} + \lambda\Delta N_{ub}^j \tag{3}$$

For the convenience of programming calculations, (3) is linearized [21].

$$H_u^j\Delta p_u + \widehat{\rho}_u^j - \rho_b^j = \lambda\Delta\varphi_{ub}^j - c\Delta d\tau_{ub} + \lambda\Delta N_{ub}^j \tag{4}$$

where H_u^j denotes the direction vector for the mobile station and satellite j; Δp_u is the coordinate correction of the mobile receiver; and $\widehat{\rho}_u^j$ represents the approximate distance between satellite j and mobile station.

The above formula is the single-difference model of BDS, while the positioning error is still influenced by clock deviation that can be eliminated through calculating the quadratic difference between different satellites [22].

For satellite k, the single-difference equation is [21]

$$H_u^k\Delta p_u + \widehat{\rho}_u^k - \rho_b^k = \lambda\Delta\varphi_{ub}^k - c\Delta d\tau_{ub} + \lambda\Delta N_{ub}^k \tag{5}$$

The second difference between (4) and (5) is

$$\begin{aligned}&\left(H_u^j - H_u^k\right)\Delta p_u + \left(\widehat{\rho}_u^j - \widehat{\rho}_u^k\right) - \left(\rho_b^j - \rho_b^k\right) \\ &\quad = \lambda\left(\Delta\varphi_{ub}^j - \Delta\varphi_{ub}^k\right) + \lambda\left(\Delta N_{ub}^j - \Delta N_{ub}^k\right)\end{aligned} \tag{6}$$

$\nabla\Delta(\bullet)$ is ordered as a double difference of different satellites, and (6) can be simplified as (7)

$$\begin{aligned}&\left(H_u^j - H_u^k\right)\Delta p_u + \left(\widehat{\rho}_u^j - \widehat{\rho}_u^k\right) - \left(\rho_b^j - \rho_b^k\right) \\ &\quad = \lambda\nabla\Delta\varphi_{ub}^{jk} + \lambda\nabla\Delta N_{ub}^{jk}\end{aligned} \tag{7}$$

where $\nabla\Delta\varphi_{ub}^{jk}$ is phase observation error and $\nabla\Delta N_{ub}^{jk}$ is double-difference integer ambiguity.

Let $L_{ub}^{jk} = \lambda\nabla\Delta\varphi_{ub}^{jk} + (\rho_b^j - \rho_b^k) - (\widehat{\rho}_u^j - \widehat{\rho}_u^k)$; the error equation form of (7) can be simplified as

$$\nabla\Delta v_{ub}^{jk} = \left(H_u^j - H_u^k\right)\Delta p_u - \lambda\nabla\Delta N_{ub}^{jk} - L_{ub}^{jk} \tag{8}$$

Here, L_{ub}^{jk} is the correction value to revise the common error.

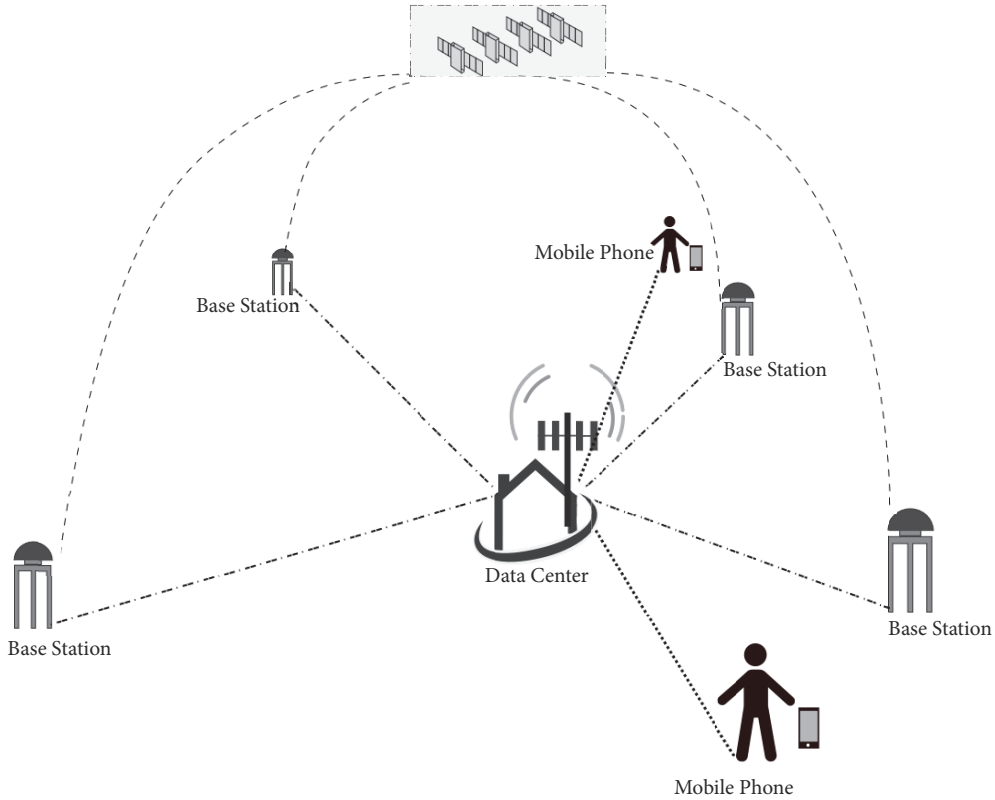


Figure 1: DCDP method model.

Within the positioning area, multiple base stations constitute the continuously operating reference stations (CORS) system [23]. All of the correction value of different base stations should be sent to data center to calculate the more suitable final value that corrects the mobile station, called the DCDP method. The data center calculates the final differential correction synthetically based on several principles such as weighted average method, partial derivative method, and minimum variance method. Then, the data center sends the final value, which could revise the common error to the mobile receiver. The model of the DCDP system is shown in Figure 1.

As is shown in Figure 1, there is no direct connection between the base station and mobile station, but the exchanged data and calculated final information by the data center causes transmission delay, data congestion, packet loss, and other issues. The paper solves these issues by introducing fog computing.

3. Fog Computing-Based Differential Positioning Method

3.1. Fog Computing. According to the definition of Cisco, the level of fog calculation is at the edge of the network, and the equipment used can be a router or base station [24]. These devices do not have many resources such as CPU, memory, hard disk, or bandwidth as they do in the data center, but they are closer to the user. The response delay is lower when a user requests the service, and some data communications are already set between the user and the data information. In other words, fog computing is a highly virtualized middle layer between end users and the cloud. Similar to cloud computing, fog computing provides user data, computing, storage, and network services [25]. It is not centralized but close to the end devices. By lowering data resources from users to the edge of the network, users do not need to obtain resources from a distant cloud center, which can effectively reduce the traffic and delay when obtaining services, finally effectively reducing the network burden.

The fog computing system consists of multiple mutually independent fog computing nodes and can independently provide localization services for mobile terminals. There are three layers including terminal layer, fog layer, and cloud layer [26]. The service model of fog computing is shown in Figure 2.

The cloud layer and the fog layer provide services for end users. The cloud layer generally serves network users, while the fog layer mainly serves intelligent mobile terminal users. The terminal layer buffers the resources from the fog layer node in advance according to a certain strategy. Then the fog layer computed node has a certain storage space and computing capacity to handle the task request of the terminal layer. Furthermore, the fog layer node can be automatically located and then, the nearest neighboring fog server is found, so that the terminal's data request can always obtain a response from the nearest fog server [26].

The fog computing architecture has three parts that match the service model including fog computing edge storage

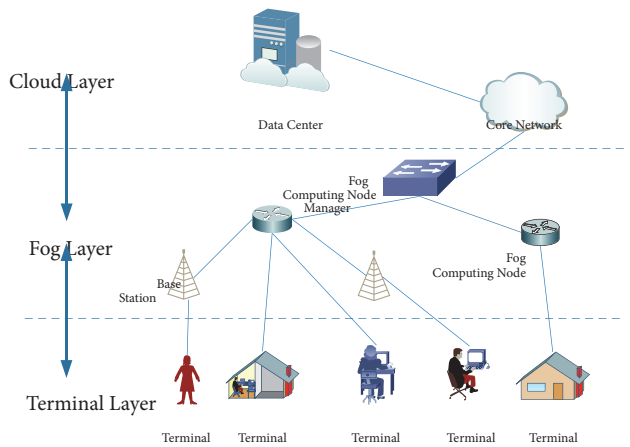


Figure 2: Service model of fog computing.

nodes (fog computing nodes), fog computing node manager, and fog computing network. The three parts correspond to three layers [26].

(1) Fog Computing Edge Storage Nodes. The fog computing edge storage node or server is a miniaturized hardware device that is disposed locally and provides localized services for mobile terminal users through wireless access. The device integrates high-density wireless network access technology, large data and low-delay channel transmission technology, and localized storage and computing technologies for cloud computing front-end hardware devices. The fog computing edge storage node is located at the edge of the network and is the closest component to the end user's fog computing system. Its data content comes from the cloud platform's data center, and it is updated intelligently according to the user's needs. The end user needs only one hop to be able to obtain the storage data service in the cloud-computing node. The fog computing nodes can be deployed on mobile vehicles, such as intercity buses, tourism operators, trains, and ships, to provide localization (without connecting to cloud data centers), high-speed, and low-cost services for users.

(2) Fog Computing Node Manager. The corresponding management program of the fog computing node is called the fog computing manager. For example, the data object storage table is used to record the data objects, access time, and access data logs that are stored in the fog computing node. Every time the mobile terminal requests the service of edge node, the manager first looks up whether there is a record stored in the table. If so, the data object of the local fog computing node is returned to the terminal user directly and then the related content of the data storage table is updated; if not, within the specified time, the data object is requested to other nodes at a minimum cost. In addition, if the other fog computing nodes have the data object in the cache, the data will be provided to the user; and if not, the request will be forwarded to the cloud to look for the data and return to the user. Correspondingly, the fog computing manager will also make the appropriate record updates and decides whether to store the data object.

(3) Fog Computing Network. Fog computing network consists of geographically dispersed nodes and storage nodes. These nodes can communicate with each other through the fog network and perform data transmission according to a set routing algorithm, data distribution strategy, etc.

If a fog computing node finds the required data stored in other fog computing nodes, the fog computing node can obtain the data needed by fog computing network. At the same time, the fog computing node can also connect to the cloud or the source server through the Internet to obtain data objects. Furthermore, the cloud can timely monitor the content information change and then update the record correspondingly.

Due to wide distribution and high density, fog computing can quickly respond to end-user requests, handle tasks in real time, and shorten access delays. Furthermore, the fog nodes are at the edge of the network and close to the end user, so the user needs only one hop to obtain the various data resources needed. Because fog computing nodes do not have as much storage and computing space as cloud computing, certain strategies must be adopted to filter the fog computing nodes that can provide maximum value for terminals in the service area, to maximize the quality of the service to users.

3.2. FCDP Method. Due to the problems of DCDP method caused by the data center such as high latency, congestion, and packet loss, this paper proposes to apply fog computing to BDS. In the FCDP method, the data center was removed, and the base station is set as the fog layer to calculate the correction information. Then all the base stations within the positioning area broadcast the correction data incessantly. The mobile receiver in the area receives the correction information from the base stations and performs a comprehensive calculation. The model of FCDP is shown in Figure 3.

In the positioning area, all base stations can participate in positioning. However, if there are too many base stations, the factors that affect the accuracy of mobile receivers will also increase. In consideration of comprehensive elimination of mobile station errors and the stability of the reference station network, four reference stations are selected for positioning in this paper. Therefore, a suitable base station selection strategy is needed to minimize network losses and balance the delay between stations.

The base station selection algorithm of this paper considers three main indicators: transmission delay (distance between base station and mobile station), balance of base station load, and performance of the base station itself. It can be seen from (1) that the pseudo-range value is related to tropospheric and ionospheric effects, which have a strong correlation with distance [27]. Although the atmospheric noise of the base station and mobile receiver can be equal by default in the case of medium or short baseline, shorter baselines have higher similarity [27]. Therefore, the weight of

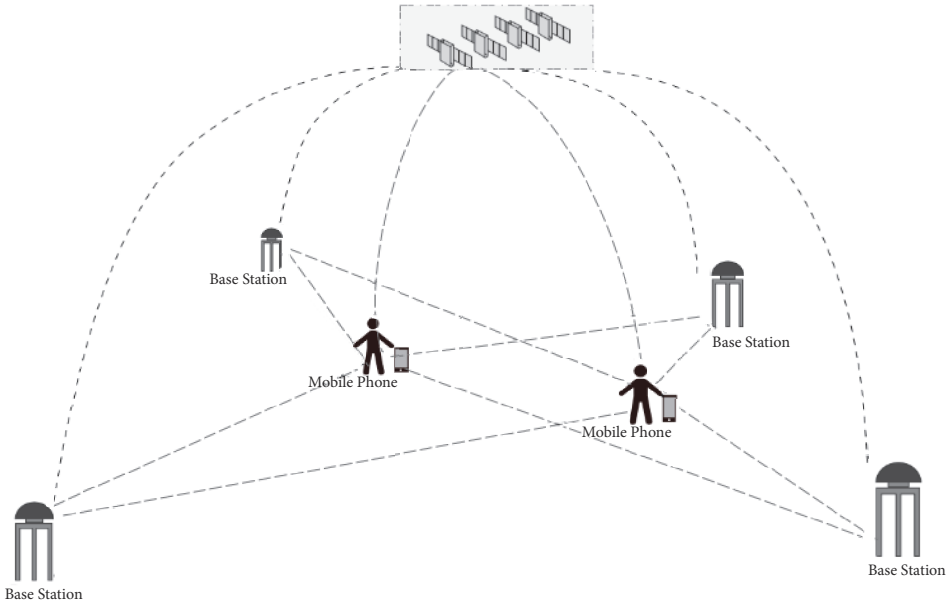


Figure 3: FCDP method model.

the base station closer to the mobile receiver should be higher. In addition, unbalanced node assignments will cause some base stations to be overloaded and increase the likelihood of node crashes. Finally, the performance of the base station itself also affects the trustworthiness of the parameters.

According to the above requirements, we need to find an optimal objective function to implement the selection strategy of the base station. The above purpose can be expressed as formula (9).

$$J(u,i) = penalty(t(u,i)) * t(u,i) * a(q,i) \tag{9}$$

where

$$penalty(t(u,i)) = \begin{cases} 1, & t(u,i) < \varepsilon \\ \infty, & t(u,i) \geq \varepsilon \end{cases} \tag{10}$$

$J(u,i)$ is the performance function of base station i to mobile receiver u; $t(u,i)$ is the time of transmitting $L_{ub_i}^{jk}$ to mobile receiver u; q is the selected rate of base station i, and $a(q,i)$ is the reliability of base station i which is positively related to the selected rate. $penalty(t(u,i))$ is the penalty function for transmission time. When the transmission time exceeds the set valueε, the penalty function value will be infinite. Then, this base station will not be selected.

Based on the size of the performance function value, the top four base stations that are not infinitely involved are selected for positioning. Therefore, if the penalty function value is infinite, this base station will not be selected.

When observing satellite j and k, the correction value of base station i is ordered to (11).

$$L_{ub_i}^{jk} = \lambda \nabla \Delta \varphi_{ub_i}^{jk} + \left(\rho_{b_i}^{j} - \rho_{b_i}^{k}\right) - \left(\widehat{\rho}_{u}^{j} - \widehat{\rho}_{u}^{k}\right) \tag{11}$$

Then the final correction value is

$$L_{f}^{jk} = \sum_{1}^{n} \frac{J(u,i)}{\sum_{1}^{n} J(u,i)} L_{ub_i}^{jk} \tag{12}$$

where n is the number of base stations participating in BDS positioning.

The weight of the correction number is inversely proportional to the distance between the base station and the mobile receiver, so the closer the distance is, the higher the credibility of the base station will be. Furthermore, the FCDP method has good antirisk ability. If the nearest base station fails to transmit information, or if any base station in the network fails, the remaining base stations can still form a network and calculate the final value based on the performance weighted.

After calculating the correction value, the error equation is

$$\nabla \Delta v_{ub}^{jk} = \left(H_{u}^{j} - H_{u}^{k}\right) \Delta p_{u} - \lambda \nabla \Delta N_{ub}^{jk} - L_{f}^{jk} \tag{13}$$

The matrix form of (13) is

$$V = AX - L \tag{14}$$

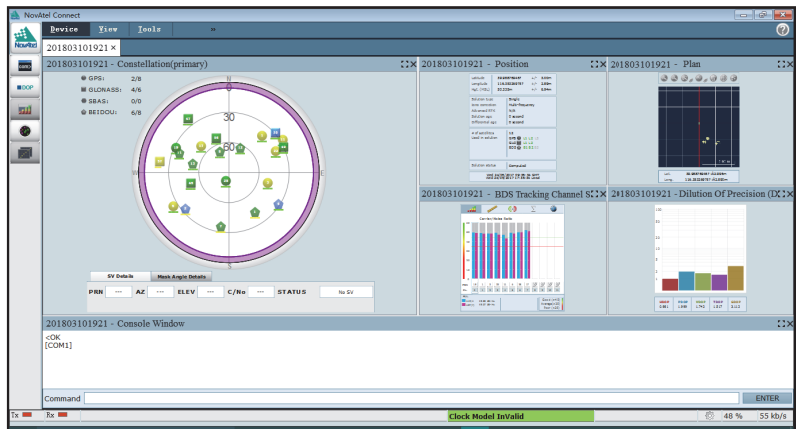


Figure 4: Interface of BeiDou satellites states.

Let m be the number of observation satellites; then

$$A = \begin{bmatrix} H_u^2 - H_u^1 & -\lambda & 0 & \cdots & 0 \\ H_u^3 - H_u^1 & 0 & -\lambda & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ H_u^n - H_u^1 & 0 & 0 & \cdots & -\lambda \end{bmatrix},$$
$$X = \begin{bmatrix} \Delta p_u \\ \nabla \Delta N_{ub}^{jk} \end{bmatrix}, \quad (15)$$
$$L = \left[L_f^{12}, L_f^{13}, \cdots, L_f^{1m} \right]^T$$

Then, when the number of observation satellites is greater than or equal to 4, the final receiver position can be calculated using the weighted least-squares algorithm. To weaken the influence of the correlation caused by the reference satellites, the chosen weight matrix usually uses the covariance inverse Q^{-1} of the carrier observations.

Setting $R = Q^{-1}$, then the solution value of weighted least-squares (WLS) algorithm is (16) [28].

$$X = \left(A^T R A \right)^{-1} A^T R L \quad (16)$$

Combined with the advantages of fog calculation, the FCDP method that removes the data center decreases the positioning delay.

4. Numerical Results and Analyses

To ensure the high reliability of the simulation experiment, this paper collected the satellite ephemeris information in the real environment as original positioning data through OEM6® Family Firmware which produced by NovAtel company, a subsidiary company affiliated with Sweden Hexagon Company. The satellite status information can be obtained using the host computer software. As is shown in Figure 4, the current satellite conditions information is shown intuitively by data collection. The acquired satellite ephemeris information converted into RINEX format can be used for positioning solution. Through the ephemeris parameters, we can obtain the coordinates of visible satellites used for positioning calculation.

4.1. Positioning Accuracy Comparison. When the coordinates of satellites in the CGCS2000 coordinate system are obtained, the positioning error of the DCDP and FCDP methods can be calculated using the weighted least square algorithm. As is shown in Figure 5, the absolute error of three axes and the root mean square error (RMSE) of these two methods are similar and all are below to 1 m, while the RMSE of FCDP is more stable. The results demonstrate that, due to the selection strategy of the base stations, the positioning accuracy is more stable and have no big ups and downs.

Figure 6 shows the PDF of the DCDP and FCDP methods. As we can see, the FCDP method has more concentrated distribution, while the DCDP method has a slightly longer tail. It means that the FCDP method has better stability, rarely undulating largely in the same long-term duration.

4.2. Delay. In the FCDP method, there is a single-hop communication delay between the user and base stations, whereas the DCDP system has a two-hop communication delay that includes transmitting data to the server and sending the final result to the mobile receiver. The known singular delay of transmitting data is 270 *ms*. Thus, the delay of the FCDP method with transmitting data is approximately 270 *ms*, while the delay of the DCDP method is approximately 540 *ms*. Therefore, the FCDP method has a shorter response time during transmission; theoretically, the delay is 270 *ms* shorter than that of the DCDP method.

The delay of the DCDP and FCDP methods is shown in Figure 7. Figure 7 illustrates that the delay of the FCDP method is less than 490 *ms* approximately and 260 *ms* shorter than the DCDP method. It means that the FCDP method can achieve faster positioning result, which has great significance especially in real-time positioning and mobile navigation fields.

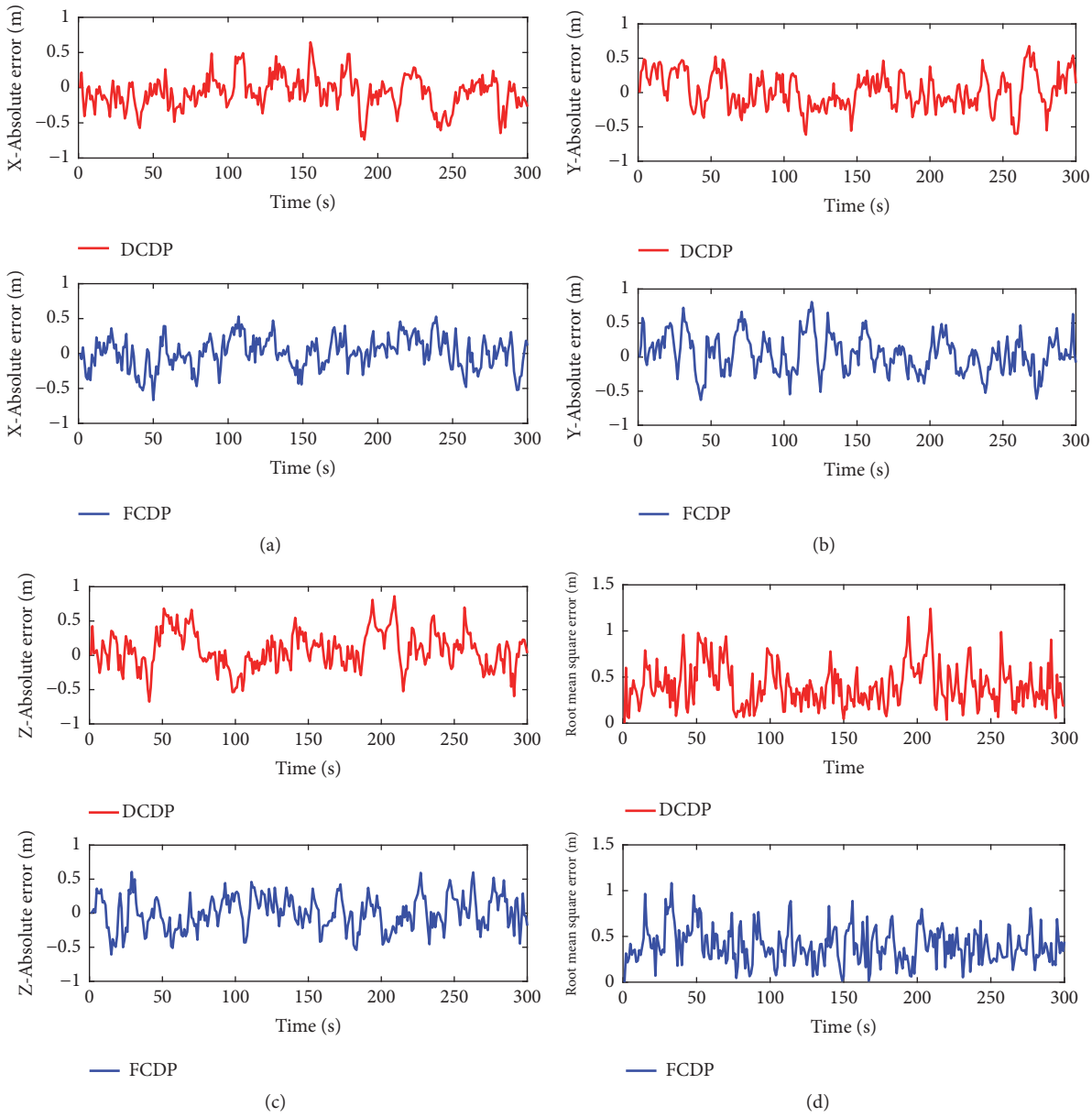


Figure 5: Error of DCDP and FCDP methods.

Consequently, the FCDP method is better than the DCDP method in terms of delay, has more stable accuracy range, and is beneficial for fast and accurate positioning of BDS. Furthermore, due to multiple fog computing nodes participating in the calculation, the data pressure is greatly shared and avoids server crashes, which are efficiently constructive to solve a large number of positioning requests and responses.

5. Conclusions

In recent years, new technology and application requirements have emerged continuously. Researchers are required to constantly learn and promote these applications from a new perspective in order to maximize the potential of BDS. The emergence of fog computing technology in the form of "decentralization" can be a way to solve the delay, congestion, and packet loss caused by the "center" of BDS. The FCDP method proposed in this paper combines the emerging fog computing technology. Under the condition of guaranteeing the accuracy of the original algorithm, the positioning delay is reduced, which is good for quick positioning. In addition, the networking of the fog computing layer also enhances the stability of the algorithm. Even if some base stations fail to transmit data effectively, the remaining base stations can still form a network without affecting the final data calculation. The simulation results demonstrate that the FCDP method has low latency and guaranteed accuracy which provide insights for improving response times.

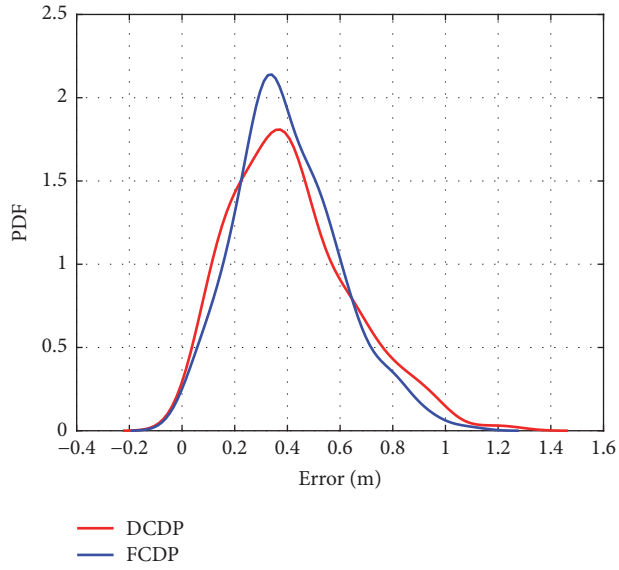


Figure 6: PDF of DCDP and FCDP methods.

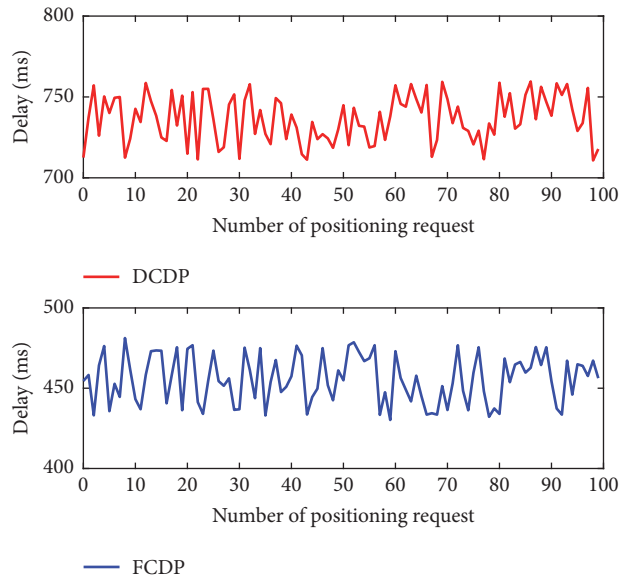


Figure 7: Delay of DCDP and FCDP methods.

Data Availability

No data were used to support this study.

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 under Grant no. 61701020, the University of Science and Technology Beijing Project under Grant no. 04130017, the Fundamental Research Funds for the Central Universities under Grant no. FRF-BD-17-015A, and the Foundation of Beijing Engineering and Technology Center for Convergence Networks and Ubiquitous Services.

References

[1] M. Li, L. Qu, Q. Zhao, J. Guo, X. Su, and X. Li, "Precise point positioning with the BeiDou navigation satellite system," *Sensors*, vol. 14, no. 1, pp. 927–943, 2014.

[2] C. Han, Y. Yang, and Z. Cai, "BeiDou navigation satellite system and its time scales," *Metrologia*, vol. 48, no. 4, pp. S213–S218, 2011.

[3] W. Tang, C. Deng, C. Shi, and J. Liu, "Triple-frequency carrier ambiguity resolution for Beidou navigation satellite system," *GPS Solutions*, vol. 18, no. 3, pp. 335–344, 2014.

[4] Z. zhang, K. Hua, and S. Sun, "The Application of the Combination of BeiDou and GPS in the Civil Aviation," *Automation and Instrumentation*, p. S1, 2005.

[5] J. Wen, D. Wang, Y. Meng et al., "Application of beidou navigation satellite system to geological survey," *Journal of Geomechanics*, vol. 3, article 003, 2012.

[6] S. Baselga and L. García-Asenjo, "GNSS differential positioning by robust estimation," *Journal of Surveying Engineering*, vol. 134, no. 1, pp. 21–25, 2008.

[7] J. K. Choi, S. H. Park, D. J. Cho et al., "Correction error generation algorithm for differential positioning performance analysis of navigation equipment," in *Proceedings of the International Conference on Control, Automation and Systems, ICCAS '08*, pp. 1099–1103, IEEE, 2008.

[8] S. G. Jin, R. Jin, and D. Li, "Assessment of BeiDou differential code bias variations from multi-GNSS network observations," *Annales Geophysicae*, vol. 34, no. 2, pp. 259–269, 2016.

[9] C. Rizos, "Alternatives to current GPS-RTK services and some implications for CORS infrastructure and operations," *GPS Solutions*, vol. 11, no. 3, pp. 151–158, 2007.

[10] U. Vollath, A. Buecherl, H. Landau et al., "Multi-base RTK positioning using virtual reference stations," in *Proceedings of the ION GPS*, vol. 95, pp. 123–131, 2000.

[11] Y. C. Hu, M. Patel, D. Sabella et al., "Mobile edge computing—A key technology towards 5G," *ETSI White Paper*, vol. 11, no. 11, pp. 1–16, 2015.

[12] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397–413, 1993.

[13] P. Cheng, F. Ren, R. Shu et al., "Catch the Whole Lot in an Action: Rapid Precise Packet Loss Notification in Data Center," in *Proceedings of the NSDI*, pp. 71-28, 2014.

[14] A. Phanishayee, E. Krevat, V. Vasudevan et al., "Measurement and Analysis of TCP Throughput Collapse in Cluster-based Storage Systems," in *Proceedings of the FAST*, vol. 8, pp. 1–14, 2008.

[15] E. Krevat, V. Vasudevan, A. Phanishayee et al., "On application-level approaches to avoiding TCP throughput collapse in cluster-based storage systems," in *Proceedings of the 2nd International Petascale Data Storage Workshop, PDSW '07, held in Conjunction with Supercomputing '07*, pp. 1–4, November 2007.

[16] F. Bonomi, R. Milito, P. Natarajan et al., "Fog computing: a platform for internet of things and analytics," in *Big Data and Internet of Things: A Roadmap for Smart Environments*, pp. 169–186, Springer, Cham, Switzerland, 2014.

[17] G. Hu, V. Khoo, P. Goh, and C. Law, "Performance of Singapore Integrated Multiple Reference Station Network (SIMRSN) for RTK Positioning," *GPS Solutions*, vol. 6, no. 1-2, pp. 65–71, 2002.

[18] P. V. W. Loomis, Carrier phase differential GPS corrections network: U.S. Patent 5,899,957 [P], 1999.

[19] B. Forssell, R. A. Harris, and M. Martin-Neira, "Carrier phase ambiguity resolution in GNSS-2," in *Proceedings of the Proceedings of Ion Gps. Institute of Navigation*, vol. 10, pp. 1727–1736, 1997.

[20] R. Odolinski, P. J. G. Teunissen, and D. Odijk, "Combined GPS+ BDS for short to long baseline RTK positioning," *Measurement Science and Technology*, vol. 26, no. 4, Article ID 045801, 2015.

[21] G. Lu, On-the-fly RTK positioning system with single frequency receiver: U.S. Patent 6,127,968, 2000.

[22] S. Zhao, Y. Chen, H. Zhang et al., "Differential GPS aided inertial navigation: a contemplative realtime approach," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 8959–8964, 2014.

[23] W. Tang, L. Jin, and K. Xu, "Performance analysis of ionosphere monitoring with beidou CORS observational data," *Journal of Navigation*, vol. 67, no. 3, pp. 511–522, 2014.

[24] S. Yi, C. Li, and Q. Li, "A survey of fog computing: concepts, applications and issues," in *Proceedings of the 2015 Workshop on Mobile Big Data*, pp. 37–42, ACM, Hangzhou, China, June 2015.

[25] T. H. Luan, L. Gao, Z. Li et al., "Fog computing: Focusing on mobile users at the edge," 2015, https://arxiv.org/abs/1502.01815.

[26] S. Park and Y. Yoo, "Network intelligence based on network state information for connected vehicles utilizing fog computing," *Mobile Information Systems*, vol. 2017, Article ID 7479267, 9 pages, 2017.

[27] T. Rieckh, R. Anthes, W. Randel, S.-P. Ho, and U. Foelsche, "Tropospheric dry layers in the tropical western pacific: comparisons of gps radio occultation with multiple data sets," *Atmospheric Measurement Techniques*, vol. 10, no. 3, pp. 1093–1110, 2017.

[28] Y. P. Sun, Y. S. Zhang, E. S. Wang et al., "Design and positioning algorithm of BD-2/GPS combined system," *Electronic Design Engineering*, vol. 23, article 027, 2011.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 8680758, 12 pages
https://doi.org/10.1155/2018/8680758

Research Article

Using NearestGraph QoS Prediction Method for Service Recommendation in the Cloud

Yiqi Fu, Ding Ding, and Seid Ahmed

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

Correspondence should be addressed to Ding Ding; dding@bjtu.edu.cn

Received 2 March 2018; Revised 2 May 2018; Accepted 29 May 2018; Published 27 June 2018

Academic Editor: Pierre-Martin Tardif

Copyright © 2018 Yiqi Fu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the advent of the mobile network, the fusion of cloud computing and fog computing is becoming feasible to promise lower latency and short-fat connection. However, there are a lot of redundant cloud-aware services with identical functionalities but a different quality of service (QoS) in the fog cloud environment. In fact, since QoS information is stored in distributed fog servers rather than remote cloud, it is hard for individuals to make recommendation and selection with sparse QoS information. Collaborative filtering is an important method for the sparsity problems and has been widely adopted on the prediction of missing QoS values. Focusing on the fact that existing researchers often ignore the QoS fluctuation in a wide range in the fog cloud environment, a novel neighbor-based QoS prediction method is proposed for service recommendation, in which a concept and calculation method is put forward to describe the stable status of services and users with quantifiable QoS values, and a NearestGraph algorithm is further designed to recognize stable or unstable candidate along with their popularity by a nearest neighbor graph structure which can help to make missing QoS values prediction in a certain order to improve final prediction accuracy. Experimental results confirm that the proposed method is effective in predicting unknown QoS values in terms of service recommendation accuracy and efficiency.

1. Introduction

As the development of mobile Internet and agility of distributed system services [1, 2], cloud computing is migrating to the fusion of cloud and fog computing since fog computing is able to better satisfy demands on lower latency and short-fat connection. At present, the composed distributed system [3] is becoming the main solution accepted by the majority [4]. However, a wide range of cloud-aware services is produced to cater for the fog cloud environment. In this situation, mobile users often feel confused to select proper cloud-aware services due to the appearance of redundant cloud-aware services with identical functionalities but a different quality of service (QoS) [5, 6]. Recommender systems are designed to address the suitable matching problem with mobile users and cloud-aware services under information overload.

The key to cloud-aware service selection and recommendation is QoS [7]. QoS is defined as a set of properties of specific cloud-aware services such as response-time, throughput, reputation, and the like, which is treated as an important criterion to distinguish among different functionally equivalent services [8]. In the fog cloud environment, QoS information is normally collected and stored in various fog servers, instead of being transferred to the remote cloud directly, due to the big volume of data and heavy transmission cost. In this situation, QoS information is always distributed but not centralized [9], which means QoS information is often sparse and unavailable for mobile users. Therefore, motivated by making an effective recommendation, it is a feasible way to complete missing QoS values by making predictions.

In fact, all roles in the fog cloud environment have the motivation to predict QoS before their assignments. A typical example of the fog cloud environment [10–12] is shown in Figure 1, which includes three roles, service user, service broker, and service designer. The same problem is happening to each role on how to manage cloud-aware services with high-quality performance. For example, service users expect more qualified services which respond more quickly while meeting basic functions. In general, it is necessary for the three roles in the fog cloud environment to predict QoS

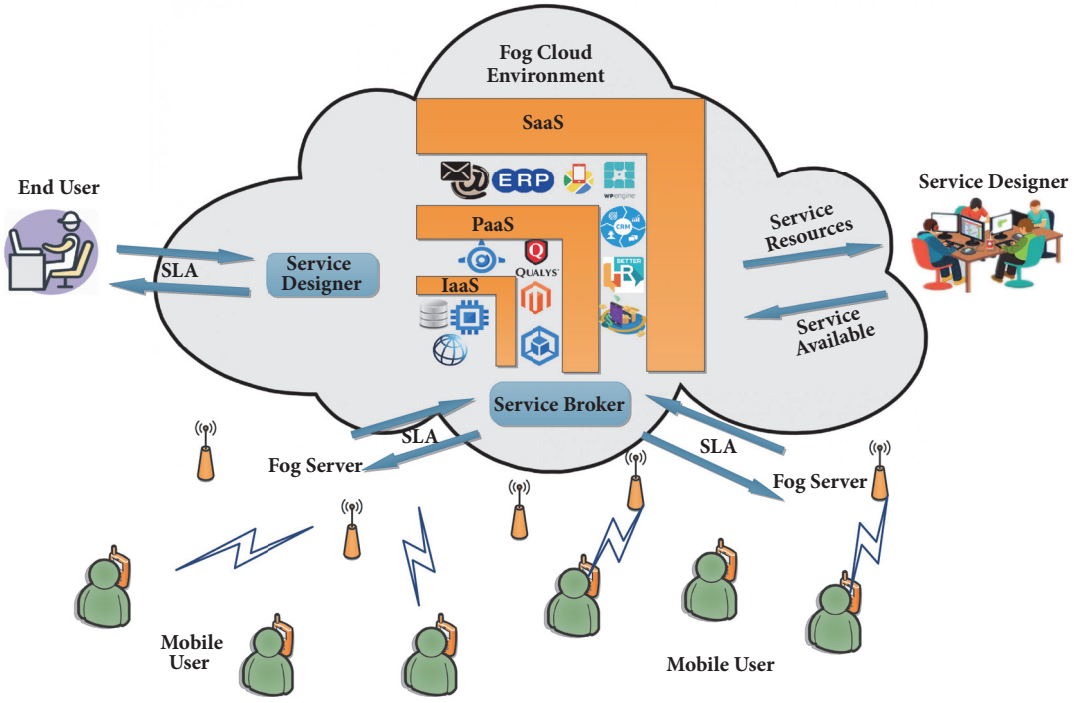


Figure 1: The architecture of fog cloud environment: each role wants to manage a cloud-aware service with "good" performance, especially QoS. However, QoS information is sparse and often varies among different roles. QoS prediction can achieve the goal of finding "good" performance through the analysis of historical QoS information.

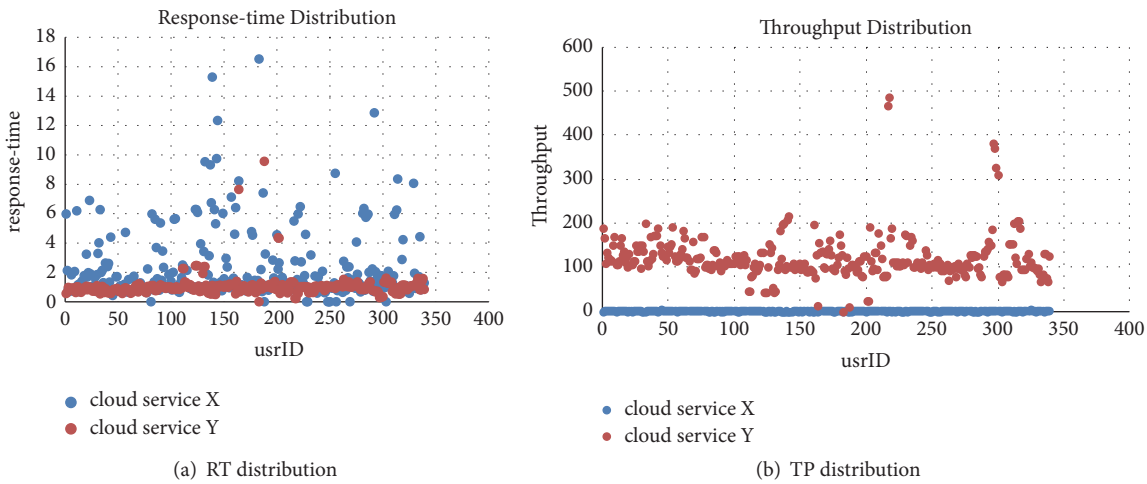
values due to the following reasons: (1) service user can only get a limited number of QoS values caused by time-and-money-consuming QoS invocation, which makes it difficult for cloud-aware service recommender to make a decision, (2) service broker always has a strong desire to manage cloud-aware services with good performances, and (3) service designer needs to deploy cloud-aware services that satisfy QoS constraints to avoid punishment. Therefore, QoS prediction is a critical issue for cloud-aware service deployment, selection, and recommendation.

At present, the studies on QoS prediction have made certain progress in recent years. Many scholars prefer to "fill" the unknown QoS values through historical QoS information and formulate it as a matrix completion problem [13]. Chen et al. [14] take user-service geographical location into account to improve prediction accuracy. Wang et al. [15] introduce more QoS values affecting aspects such as time and location. Wu et al. [16] answer this problem by considering the relationship between similarity and candidate's consistency. Some other researchers devote them to finding solutions on how to improve the poor credibility of a fog cloud environment. Tang et al. [17] apply the trust concept for cloud-aware service QoS prediction. Su et al. [18] make a prediction for missing QoS values based on the trust relationship.

However, to the best of our knowledge, there is still a lack of research efforts explicitly targeting on the fluctuation of QoS values related to mobile users' status and services' status. In a highly dynamic Internet environment, QoS values of cloud-aware services often fluctuate in a large range due to the variety of users' mobile networking environments and physical distance between mobile users and fog servers. There is a current situation that some services perform more "unstable" according to a study on the real world QoS[19]. We select two services in random which is invoked by 339 users and draw their QoS values distribution as shown in Figure 2. We all feel service Y in blue is more unstable compared with service X in orange intuitively in Figure 2. Therefore, we can conclude that a cloud-aware service with a wide QoS range performance is not of general applicability and should not be recommended to other users since it is difficult to make an accurate prediction when candidate services with "unstable" performance are employed.

In this paper, the problem of QoS prediction is formulated to leverage historical QoS information. Inspired by the fact of QoS fluctuation, we propose a novel neighbor-based QoS prediction algorithm under the assumption that QoS values have a close relation with services and users in the fog cloud environment. In our approach, a concept and quantization method is put forward to represent the stable status of services and users in the fog cloud environment. And a graph structure is adopted to recognize stable or unstable candidate and to expose their popularity at the same time. Based on this, a NearestGraph method is used to generate an optimal prediction order to get the higher prediction accuracy.

The remainder of this paper is organized as follows. Section 2 introduces related works of QoS prediction and existing methods. Section 3 presents our proposed QoS prediction method for cloud-aware service recommendation. Section 4 provides our experimental results and the details of our experimental implementation. Section 5 sets out our conclusion and looks forward to future works.



(a) RT distribution (b) TP distribution

Figure 2: Dataset visualization.

2. Related Works

At present, there are a lot of efforts and results devoted to tackling the issue of QoS prediction. Initially, scholars adopt static methods to make a QoS prediction. Static methods use the arithmetic average value for prediction, including average QoS value from global, user, and service, respectively. These methods are simple and easy to implement, disregarding the situation-aware factors of users and services. Moreover, these static methods cannot reflect the dynamic properties of QoS values, which are leading to greater prediction error between predicted and actual values according to our experimental results in Section 4.

Motivated by the success of traditional recommender systems, existing works on QoS prediction in the fog cloud environment is usually based on collaborative filtering. Collaborative filtering (CF) methods are widely used to rate prediction in recommender systems. It exploits the similarity between users' experiences to predict user preference on unknown items. The intuitive idea is to identify "similar" users with the active user and to predict the active user's preference based on these similar users' feedback. CF can be further divided into two main categories: model-based method and neighbor-based method.

Model-based method makes a prediction from the known QoS values by learning a predictive model [20]. Observed values are used to learn two matrices which are the basis to calculate similarities among users. However, model-based method suffers from the ignorance of the low-rank structure of real world user-service matrices [21]. The main idea of the model-based method is based on matrix completion, in which the key is to exploit the low-rank structure of the user-service matrix. Lee et al. [22] present an algorithm for nonnegative matrix factorization indicating that there is only a small number of factors influencing the service performance. Some scholars think QoS has strong relation with time and put forward an online prediction. Zhu et al. [23] propose a method for running cloud-aware service to predict its QoS value.

The neighbor-based method uses QoS values of similar users or services to make QoS predictions directly. Shao et al. [24]first introduce a collaborative filtering approach for similarity mining and inference based on historical QoS information in the user-service matrix. They perform positive and negative user similarity calculations separately and integrate them using a weighted mean equation. Zhu et al. [25]give a QoS prediction approach based on multidimension, which takes timing constraints, QoS, throughput, fairness, and load balancing into account. Zheng et al. [5] propose a novel QoS ranking prediction model with the consideration of different cloud users having different preferences for different QoS attributes values.

In this paper, we mainly focus on the neighbor-based collaborative filtering since it is simple to implement and the prediction results are often easy to explain. The prediction accuracy of the neighbor-based method is highly influenced by the available similar candidates. Similar candidates play an effective role in QoS prediction phase mainly since they come from similar computation and assign more or less importance to the target in the prediction. However, the sparsity of QoS information always degrades the accuracy of QoS prediction. In our proposed approach, we address this challenge by introducing graph structure to expose candidate's own popularity.

3. NearestGraph QoS Prediction

In this section, the problem of QoS prediction is described and formulated in Section 3.1. After that, both user-user similarity and service-service similarity are computed in Section 3.2 to select neighbors. Our NearestGraph algorithm is presented in Algorithms 1 and 2 to predict missing QoS values at last.

3.1. Problem Description. In this paper, the problem of QoS prediction is described as follows: considering a fog cloud environment with m users and n cloud-aware services, the QoS of n cloud-aware services rated by m users is represented as an $m \times n$ matrix R. The entry $r_{i,j} \in \mathbb{R}^k$ is a k-dimensional

Input: usrG: user nearest neighbor graph; R: user-service matrix
Output: $usrSet(t)$: prediction order for user-based CF
1: $usrSet(t)$=[ ];
2: **for** each $i \in$ [0, usrG.vertices] **do** //add property usrWeight, usrIndegree to a graph usrG=(usrID,usrEdge)
3: usrG←usrWeight given by Eq.(6);
4: usrG←usrIndegrees given by usrG.inDegrees;
5: **end for**
6: **repeat**
7: select usrID from usrG.vertices where usrG.vertices.Weight.max in {select ∗ from usrG.vertices where usrG.vertices.Indegree.min};
8: $usrSet(t).append$(usrID);
9: usrG←usrIndegrees given by usrG.inDegrees;
10: **until** usrG.vertices.count is 0

Algorithm 1: NearestGraph algorithm for $usrSet$.

Input: usrG: service nearest neighbor graph; R: user-service matrix
Output: $servSet(t)$: prediction order for service-based CF
1: $servSet(t)$=[];
2: **for** each $i \in$ [0, serG.vertices] **do** //add property serWeight, serIndegree to a graph serG=(serID,serEdge)
3: serG←serWeight given by Eq.(7);
4: serG←serIndegrees given by serG.inDegrees;
5: **end for**
6: **repeat**
7: select serID from serG.vertices where serG.vertices.Weight.max in {select ∗ from serG.vertices where serG.vertices.Indegree.min};
8: $serSet(t).append$(serID);
9: serG←serIndegrees given by serG.inDegrees;
10: **until** serG.vertices.count is 0

Algorithm 2: NearestGraph algorithm for $serSet$.

vector representing the QoS values of k^{th} criteria. Let $U = \{u_1, u_2, \ldots, u_i, \ldots, u_m\}$, $i \in \{1, 2, \ldots, m\}$ be the set of m users, let $S = \{s_1, s_2, \ldots, s_j, \ldots, s_n\}$, $j \in \{1, 2, \ldots, n\}$ be the set of n cloud-aware services, Ω is set of all tuples $\{i, j\}$, and Λ is set of all unknown tuples (i, j), $r_{i,j} = \varnothing$. Then the missing information $\{r_{i,j} \mid (i, j) \in \Lambda\}$ is filled based on the existing information $\{r_{i,j} \mid (i, j) \in \Omega - \Lambda\}$. The order of filling in the matrix R is expressed as $r^t_{i,j} \in \Lambda, t \in (1, |\Lambda|)$ and there is an optimal order t to satisfy the higher accuracy.

Figure 3 shows a matrix R formed by m users and n cloud-aware services. The shaded part of the matrix indicates the user has invoked the cloud-aware service and has rated the corresponding QoS value. The blank part indicates the user has not invoked the cloud-aware service and the QoS values need to be predicted. The objective of the missing QoS value prediction is to make the user-service matrix denser within certain iteration phases [26].

Due to analysis of real world QoS datasets, QoS values can vary widely and are highly skewed with large variances that degrade the accuracy of prediction. Without loss of generality, we apply the following function to QoS data in order to map QoS values onto the interval $(0, 1)$.

$$r_{i,j} = \frac{r_{i,j} - r_{min}}{r_{max} - r_{min}} \tag{1}$$

where r_{min} and r_{max} are the minimum and maximum QoS values, respectively.

3.2. Neighbors Selection. We can find the neighborhood similarities of users and services by employing Pearson Correlation Coefficient (PCC). PCC is widely used in neighbor-based recommendation systems for similarity computation and proved to have high accuracy. In this paper, we adopt an enhanced-PCC method proposed by Zheng for the neighborhood similarity computation on both sets of users and services. The similarity between two users a and b is defined by the following equation:

$$\begin{aligned} &\mathrm{sim}(a, b) \\ &= \frac{2 \times |S_a \cap S_b|}{|S_a| + |S_b|} \\ &\times \frac{\sum_{i \in (S_a \cap S_b)} (r_{a,i} - \bar{r}_a)(r_{b,i} - \bar{r}_b)}{\sqrt{\sum_{i \in (S_a \cap S_b)} (r_{a,i} - \bar{r}_a)^2} \sqrt{\sum_{i \in (S_a \cap S_b)} (r_{b,i} - \bar{r}_b)^2}} \end{aligned} \tag{2}$$

where $\mathrm{sim}(a, b)$ falls into the interval $[-1, 1]$, $|S_a \cap S_b|$ is the number of cloud-aware services that are invoked by the two users, and $|S_a|$ and $|S_b|$ are the number of cloud-aware services

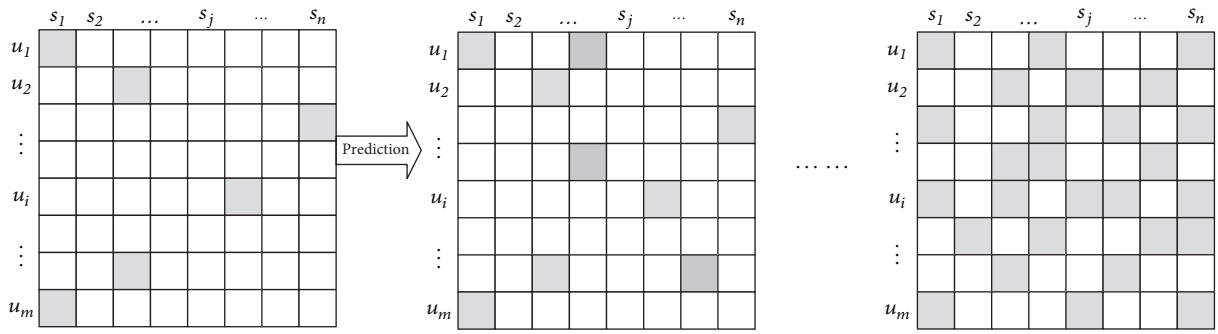


Figure 3: Prediction problem formulation.

invoked by user a and user b, respectively. $r_{a,i}$ is a vector of QoS values of cloud-aware service i observed by user a and $\bar{r}_a$ represents the average QoS values of different cloud-aware services observed by user a.

Similar to the user similarity computation, we also employ enhanced-PCC to compute the similarity between cloud-aware service x and y as follows:

$$\text{sim}(x,y) = \frac{2 \times \left|U_x \cap U_y\right|}{\left|U_x\right| + \left|U_y\right|} \times \frac{\sum_{u \in (U_x \cap U_y)} \left(r_{u,x} - \bar{r}_x\right)\left(r_{u,y} - \bar{r}_y\right)}{\sqrt{\sum_{u \in (U_x \cap U_y)} \left(r_{u,x} - \bar{r}_x\right)^2} \sqrt{\sum_{u \in (U_x \cap U_y)} \left(r_{u,y} - \bar{r}_y\right)^2}} \quad (3)$$

where $\text{sim}(x, y)$ falls into the interval $[-1, 1]$, $|U_x \cap U_y|$ is the number of users who invoked both cloud-aware services x and y previously, and $|U_x|$ and $|U_y|$ are the number of users who invoked cloud-aware services x and y, respectively. $r_{u,x}$ is a vector of QoS values of user u when he invokes cloud-aware service x and $\bar{r}_x$ represents average QoS values of different users when they invoke cloud-aware service x.

After the similarity computations, we can get the user similarity matrix and the service similarity matrix. At the same time, we can also identify their neighbors by similarity values in the ascending order. Traditional top-K algorithms select the top k most similar neighbors for making missing value prediction. In practice, some neighbors with negative similarity values could greatly decrease the prediction accuracy. In this paper, we exclude dissimilar neighbors with negative enhanced-PCC values. We employ the following equation to find a set of proper similar users for user i as Ψ_i:

$$\Psi_i = \left\{u_k \mid \text{sim}(i,k) > 0,\ \text{rank}_i(k) \leqslant K,\ k \neq i\right\} \quad (4)$$

where $\text{rank}_i(k)$ is the ranking position of user k in the similarity neighbors of user i and K indicates the lowest ranking position manually.

In the same way, we can get the set of proper similar cloud-aware services for cloud-aware service j as Φ_j:

$$\Phi_j = \left\{s_k \mid \text{sim}(j,k) > 0,\ \text{rank}_j(k) \leqslant K,\ k \neq j\right\} \quad (5)$$

where $\text{rank}_j(k)$ is the ranking position of cloud-aware service k in the similarity neighbors of cloud-aware service j and K indicates the lowest ranking position manually.

3.3. Predicting Missing QoS Values with NearestGraph. After user neighbors selection, we find an interesting fact that some users or services are relatively "popular" to others. For example, user E is on the top-1 similarity ranking position of user A. It also happens to user C when user E ranks top-1 in user C's similar neighbors. User B and User D may be confronted with the same situation. This is not an occasional case but happens for most similar neighbors. In order to expose this kind of popularity of users or services, we construct a directed graph by nearest neighbor graph as shown in Figure 4.

In Figure 4, a user is represented by `usrG=(usrID, usrEdge)`, in which `usrID` labels a user and `usrEdge` shows the relationship between the user and his most similar neighbor-a directed edge will line from a user to his most similar neighbor. Therefore, the indegree of a vertex in our nearest neighbor graph indicates the degree to which other vertices are in favor of this vertex. A vertex with larger indegree means it is very "popular" and will have a higher influence on other vertices. It can be understood as the relationship between celebrities and fans in social networking sites. A celebrity who has more fans means greater appeal, which reveals the greater influence at the same time. Similar method can be used to represent cloud-aware service by `servG=(servID, servEdge)`, in which `servID` labels a service and `servEdge` shows the relationship of a service and his most similar neighbor.

Furthermore, to reflect the stability of different users and cloud-aware services as shown in Figure 2, a concept of candidate stability is also proposed. We employ the following equation to describe the stability of user's status.

$$\text{stability}(a) = \frac{\sum_{x \in S_a} \left(r_{a,x} - \bar{r}_a\right)}{\left|S_a\right| \times \bar{r}_a} \quad (6)$$

Similarly, we can describe the stability of cloud-aware service's status as follows.

$$\text{stability}(y) = \frac{\sum_{b \in U_y} \left(r_{b,y} - \bar{r}_y\right)}{\left|U_y\right| \times \bar{r}_y} \quad (7)$$

where a smaller value of stability will indicate a more stable status.

In order to introduce the stability of users or services, we further extend above nearest neighbor graph

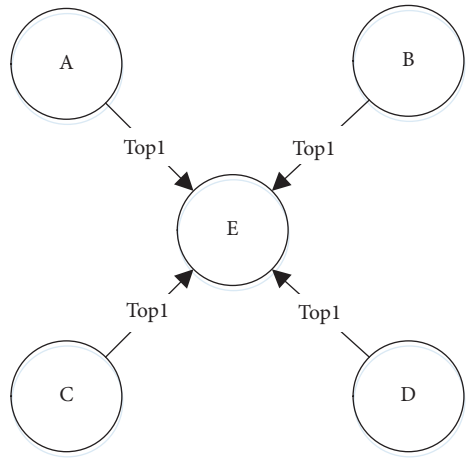


Figure 4: Nearest neighbor graph: we found that, in the sets of most similar neighbors for different users, some users tended to appear frequently.

to `usrG=(usrID, usrWeight, usrEdge)` and `servG=(servID, servWeight, servEdge)`, respectively, in which `usrWeigh` is the stability of user's status and `servWeigh` is the stability of service's status.

In this paper, we believe both popularity and stability will play an important role in QoS prediction and should be used to obtain better prediction accuracy. Therefore, we propose an algorithm called NearestGraph to achieve this goal, which can generate an optimal prediction order by nearest neighbor graph based on different popularity and stability. The main strategies of NearestGraph are the following three key points.

(1) Select the `usrID` with the minimum indegree.

(2) Select the `usrID` with the maximum weight if more than one vertex has the same indegree.

(3) Select the `usrID` with the minimum dictionary order if more than one vertex has the same indegree as well as weight.

The reason why we use those three rules comes from two facts: (1) those vertices with larger indegree, which means they are more popular, will affect more users and should be kept longer in our nearest neighbor graph to make full use of their important influence; (2) those vertices with larger weight, which means they are more stable, will have more positive impact on QoS prediction and should be kept longer in our nearest neighbor graph to make full use of their important influence. Now we take an example, shown in Figure 5, to illustrate the process of our NearestGraph algorithm.

Phase (a) is the initial state of nearest neighbor graph in which a property of vertex called `weight` is introduced to represent the status of stability and the directed edge is used to show the relationship between the user and his most similar neighbor. For example, vertex v_5 with weight 0.42 means it is more stable than vertex v_2 with weight 0.40, and vertex v_1 pointing to vertex v_2 expresses v_2 is the most similar neighbor of v_1. Then we will decide which vertex will be predicted according to the weight and indegree shown in nearest neighbor graph. According to No. 1 strategy of NearestGraph, v_1 will be predicted first since it has minimum indegree. In the next phase (b), there are two vertices with the same indegree after applying No. 1 strategy. Here No. 2 strategy can help us to make a decision in such a situation. v_5 should be predicted in phase (b) for its high stability. Then we loop through the three-key-point strategies to obtain a complete prediction order until there is only one vertex left in the graph structure. We can get a prediction order $usrSet$: $v_1 \Rightarrow v_5 \Rightarrow v_2 \Rightarrow v_4 \Rightarrow v_3 \Rightarrow v_7 \Rightarrow v_6$ in this example. The details of our NearestGraph algorithm for $usrSet(t)$ are as Algorithm 1.

Based on the prediction order generated by Algorithm 1, user-based method employs the values of entries to predict the missing entry $r_{i,j}$ in the user-service matrix as follows:

$$r^{U}{}_{i,j} = \overline{r}_i + \sum_{k \in \Psi_i} \frac{\operatorname{sim}(u_i, u_k)}{\sum_{a \in \Psi_i} \operatorname{sim}(u_i, u_a)} \times \left(r_{k,j} - \overline{r}_k\right), \quad (i, j) \in \{(\Omega - \Lambda) \cap usrSet\} \tag{8}$$

where $\overline{r}_i$ and $\overline{r}_k$ are the average existence QoS values of different cloud services rated by u_i and u_k, respectively.

We can also give the prediction order for $servSet(t)$ in a similar way as shown in Algorithm 2. And the values from service prediction order are correspondingly employed for prediction in service-based method as follows:

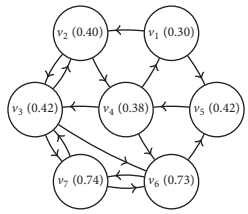
$$r^{S}{}_{i,j} = \overline{r}_j + \sum_{k \in \Phi_j} \frac{\operatorname{sim}(s_j, s_k)}{\sum_{a \in \Phi_j} \operatorname{sim}(s_j, s_a)} \times \left(r_{i,k} - \overline{r}_k\right), \quad (i, j) \in \{(\Omega - \Lambda) \cap servSet\} \tag{9}$$

where $\overline{r}_j$ and $\overline{r}_k$ are the average existence QoS values of s_j and s_k rated by different users, respectively.

In this paper, both user-based and service-based approaches are adopted as follows:

$$r^{*}{}_{i,j} = \lambda \times r^{S}{}_{i,j} + (1 - \lambda) \times r^{U}{}_{i,j} \tag{10}$$

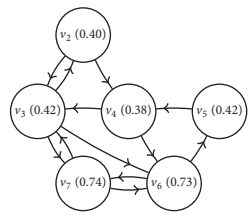
(a)



(1)

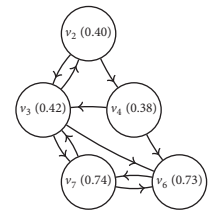
ID	weight	indegree
v_1	0.30	1
v_2	0.40	2
v_3	0.42	3
v_4	0.38	2
v_5	0.42	2
v_6	0.73	3
v_7	0.74	2

(b)



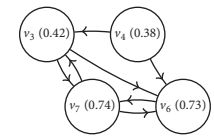
ID	weight	indegree
v_2	0.40	1
v_3	0.42	3
v_4	0.38	2
v_5	0.42	1
v_6	0.73	3
v_7	0.74	2

(c)



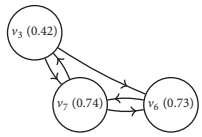
ID	weight	indegree
v_2	0.4	1
v_3	0.42	3
v_4	0.38	1
v_6	0.73	3
v_7	0.74	2

(d)



ID	weight	indegree
v_3	0.42	2
v_4	0.38	0
v_6	0.73	3
v_7	0.74	2

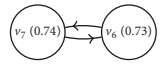
(e)



(2)

ID	weight	indegree
v_3	0.42	1
v_6	0.73	2
v_7	0.74	2

(f)



ID	weight	indegree
v_6	0.73	2
v_7	0.74	2

(g)

ID	weight	indegree
v_6	0.73	0

Prediction Order:

phase	ID
(a)	v_1
(b)	v_5
(c)	v_2
(d)	v_4
(e)	v_3
(f)	v_7
(g)	v_6

Figure 5: NearestGraph process.

The mixed approach can help us to get much more missing QoS values and therefore can improve the accuracy of prediction. The parameter λ controls how much fusion proportion of these two methods and can be trained on a sample dataset from the real world.

The complete QoS prediction algorithm is summarized in Algorithm 3.

4. Experiment

In this section, we evaluate the effectiveness of our proposed method on a distributed and parallel platform, Spark system. Section 4.1 introduces two typical metrics to assess the prediction accuracy. The comparison experiments on the prediction accuracy are conducted with different baseline algorithms in neighbor-based CF fields in Section 4.2 and three key parameters of NearestGraph on the prediction accuracy are further demonstrated in Sections 4.3, 4.4, and 4.5. All the experiments are conducted by using 4 PCs with i5-4460 CPU and 16G RAM as our hardware platform.

We evaluate the QoS prediction accuracy of our proposed method based on a real world QoS dataset which is widely used to evaluate the performance of QoS prediction. It contains response-time (response-time measures the time duration between user sending a request and receiving a response) and throughput (throughput stands for the data transmission rate of a user invoking a service) of 5828 services invoked by 339 distributed computers located in 30 countries from PlanetLab. According to statistics of this QoS dataset as shown in Table 1, the range of response-time and throughput are 0–20 s and 0–1000 kbps, respectively, and the means of response-time and throughput are 0.910 s and 47.386 kbps, respectively.

There are 100837 QoS records about response-time property and 143422 QoS records about throughput property in this QoS dataset. The corresponding user-service matrices on both these two QoS properties have some entries with the value of -1, which means the current QoS value cannot be obtained or the service is unreachable in the real world. Therefore, the entries with the value of -1 are where we need to predict in the matrix.

4.1. Metrics. To evaluate the performance of our proposed NearestGraph method, we compare its prediction accuracy with some neighbor-based CF methods by computing mean absolute error (MAE) and root-mean-square error (RMSE), which is to calculate the errors between predicted values and real values. The metric MAE is defined as

$$MAE = \frac{\sum_{i,j} \left| r_{i,j} - r^*_{i,j} \right|}{N} \tag{11}$$

Input: R: user-service matrix; K: lowest ranking position; λ: degree of fusion prediction results
Output: R^*
1: **for all** $(i, j) \in \Omega - \Lambda$ **do**
2: compute the similarity $\text{sim}(u_i, u_j)$ by Eq.(2)
3: compute the similarity $\text{sim}(s_i, s_j)$ by Eq.(3)
4: **end for**
5: **for all** $(i, j) \in \Omega - \Lambda$ **do**
6: similar user set Ψ_i by Eq.(4)
7: similar service set Φ_j by Eq.(5)
8: **end for**
9: Learn *usrSet* by applying Algorithm 1
10: Learn *servSet* by applying Algorithm 2
11: **for all** $(i, j) \in \Lambda$ **do**
12: compute $r^U_{i,j}$ by Eq.(8)
13: compute $r^S_{i,j}$ by Eq.(9)
14: fill by Eq.(10)
15: **end for**

Algorithm 3: QoS prediction algorithm.

Table 1: Statistics of QoS dataset.

Statistics	Response-Time(seconds)	Throughput(kbps)
Value Range	(0,20)	(0,1000)
Mean	0.910	47.386
Median	0.3320	11.07
Standard Variance	1.9320	107.4093
User Num	339	339
Service Num	5828	5828
Records Num	1974675	1974675

and RMSE is defined as

$$RMSE = \sqrt{\frac{\sum_{i,j} \left(r_{i,j} - r^*_{i,j}\right)^2}{N}} \quad (12)$$

where $r_{i,j}$ is the QoS value of cloud-aware service s_j observed by user u_i, $r^*_{i,j}$ is QoS value of cloud-aware service s_j that would be observed by user u_i as predicted by a method, and N is the number of predicted QoS values. According to the definitions, the smaller value of metric indicates the higher accuracy of prediction.

4.2. Performance Comparison. In this part, we conduct an overall comparison experiment on our NearestGraph method and some baseline algorithms in neighbor-based CF fields on both MAE and RMSE. They are listed as follows:

UMean: mean QoS values obtained by a user are used to predict the missing QoS value which has not been obtained by this user.

IMean: mean QoS values obtained by all users are used to predict the missing QoS value which has not been obtained by some users.

UPCC: it is a user-based collaborative filtering method, which uses similar users calculated by Pearson Correlation Coefficient to make a prediction [24].

IPCC: it is an item-based collaborative filtering method, which uses similar items calculated by Pearson Correlation Coefficient to make a prediction [27].

WSRec: it is a hybrid collaborative filtering method that combines IPCC and UPCC and uses both similar users and similar services for QoS prediction [5].

In order to simulate the users' invocation of cloud-aware services in the real world, we remove some entries from user-service matrix in random and compare their values with predicted ones. For example, 10% represents that we randomly remove 90% entries and use the remaining 10% entries to predict the values of removed entries. The parameter settings of NearestGraph are $top - K = 10$ and $\lambda = 0.5$ in the experiments.

Experiment results are shown in Table 2. We highlight the best performance of all methods for each row in Table 2. We can easily see from Table 2 that NearestGraph always obtains the minimum MAE and RMSE of response-time and throughput almost for all different matrix densities, which means it can improve the prediction accuracy. Moreover, with the value of matrix density increasing from 10% to 30%, the MAE and RMSE of NearestGraph method become smaller and smaller since a denser matrix will provide more information for the missing QoS value prediction.

Comparing the MAE and RMSE of response-time and throughput in Table 2, we can also find that the MAE and

Table 2: Performance comparisons.

Matrix Density(%)	Metrics	Response-Time (seconds)					
		UMean	IMean	UPCC	IPCC	WSRec	NearestGraph
10	MAE	0.8785	0.7015	0.6761	0.6897	0.6679	**0.6643**
	RMSE	1.8591	1.5813	1.4078	1.4296	1.4053	**1.4027**
20	MAE	0.8768	0.6867	0.5517	0.5917	0.5431	**0.5104**
	RMSE	1.8548	1.5342	1.3151	1.3268	1.2986	**1.2785**
30	MAE	0.8747	0.6818	0.5159	0.5037	0.4927	**0.4723**
	RMSE	1.8557	1.5311	1.2680	1.2252	1.1973	**1.1246**
Matrix Density(%)	Metrics	Throughput(kbps)					
		UMean	IMean	UPCC	IPCC	WSRec	NearestGraph
10	MAE	54.0084	29.2651	26.1230	29.2651	24.3285	**24.3269**
	RMSE	110.2821	66.6334	63.9573	64.2285	64.1908	**63.5435**
20	MAE	53.6768	27.3393	24.2695	26.8318	22.7717	**21.7493**
	RMSE	110.2977	64.3986	54.4783	60.0825	54.3701	**52.8731**
30	MAE	53.8792	26.6239	23.7455	26.4319	21.3194	**19.6530**
	RMSE	110.1751	64.3986	54.4783	57.8593	51.7768	**50.5765**

RMSE of response-time are larger than those of throughput which means NearestGraph performs better on throughput property than on response-time property. Taking the matrix density of 30% as an example, we can calculate the MAE improvements of NearestGraph over WSRec, respectively. For the response-time property, the MAE improvement is $((1.1973 - 1.1246)/1.1973) \times 100\% = 4.14\%$, while for the throughput property, the MAE improvement is $((21.3194 - 19.653)/21.3194) \times 100\% = 7.82\%$. That confirms that our proposed method focuses on facts of the QoS fluctuation and can make a better performance in a wide range of QoS values (just like the range of throughput is 0-1000 kbps and the range of response-time is only 0-20 s).

4.3. Impact of Matrix Density. In order to explore the impact of matrix density, we compare the prediction accuracy of all the methods under different matrix densities and present the results in Figure 6. The density of the matrix increases from 10% to 30% with a step of 10%. The parameter settings in this experiment are $top - K = 10$ and λ=0.5.

The MAE and RMSE results of response-time are shown in Figures 6(a) and 6(b) and the MAE and RMSE results of throughput are shown in Figures 6(c) and 6(d). In these figures, the green line NearestGraph stands for is always below any other lines, which means our proposed NearestGraph method gets the smallest values of MAE and RMSE under different matrix densities. Moreover, we can observe that the performance of our NearestGraph method improves with the increase of matrix density, which indicates that collecting more QoS information will greatly enhance prediction accuracy when the matrix is sparse.

4.4. Impact of λ. The parameter λ here controls how much fusion proportion of user-based and service-based method. A larger value of λ means user-based approach will contribute more to the hybrid prediction. In Figure 7, we study the impact of parameter λ in the proposed NearestGraph method on prediction accuracy by varying the values of λ from 0 to 1 with a step of 0.1 under the condition of $top - K$=10.

Figures 7(a) and 7(b) show the MAE and RMSE results of response-time and throughput, respectively. The prediction accuracies increase when we increase the value of λ at first. But when λ surpasses a certain threshold, the prediction accuracy decreases with the further increase of λ. From Figure 7, we can also find that NearestGraph gets the best performance when $\lambda \in [0.4, 0.7]$.

4.5. Impact of Top − K. The parameter $top - K$ determines the size of candidates sets including similar users and similar services. In Figure 8, we study the impact of parameter $top - K$ in the proposed NearestGraph method on prediction accuracy by varying the values of $top - K$ from 2 to 20 with a step of 2 under the condition of λ=0.5.

Figures 8(a) and 8(b) represent the MAE and RMSE results of response-time and throughput, respectively. The experimental results show that our NearestGraph will achieve best prediction accuracy (minimum MAE and RMSE) when $top - K$ is set around 10. This is because too small $top - K$ value will exclude useful information from some similar candidates, while too large $top - K$ value will introduce noise from dissimilar candidates, which will impact the prediction accuracy.

5. Conclusion and Future Work

In the fog cloud environment, to reduce the data transmission cost from mobile users to the cloud, QoS information is often first handled by distributed fog servers instead of being sent to a remote cloud directly. However, such a cross-platform data distribution will lead to the sparsity of QoS information for service recommendation. Focusing on the fact that existing researches on missing QoS value prediction often ignore the QoS fluctuation in a wide range especially in the fog cloud environment, we propose a novel QoS prediction method by using NearestGraph algorithm for service recommendation.

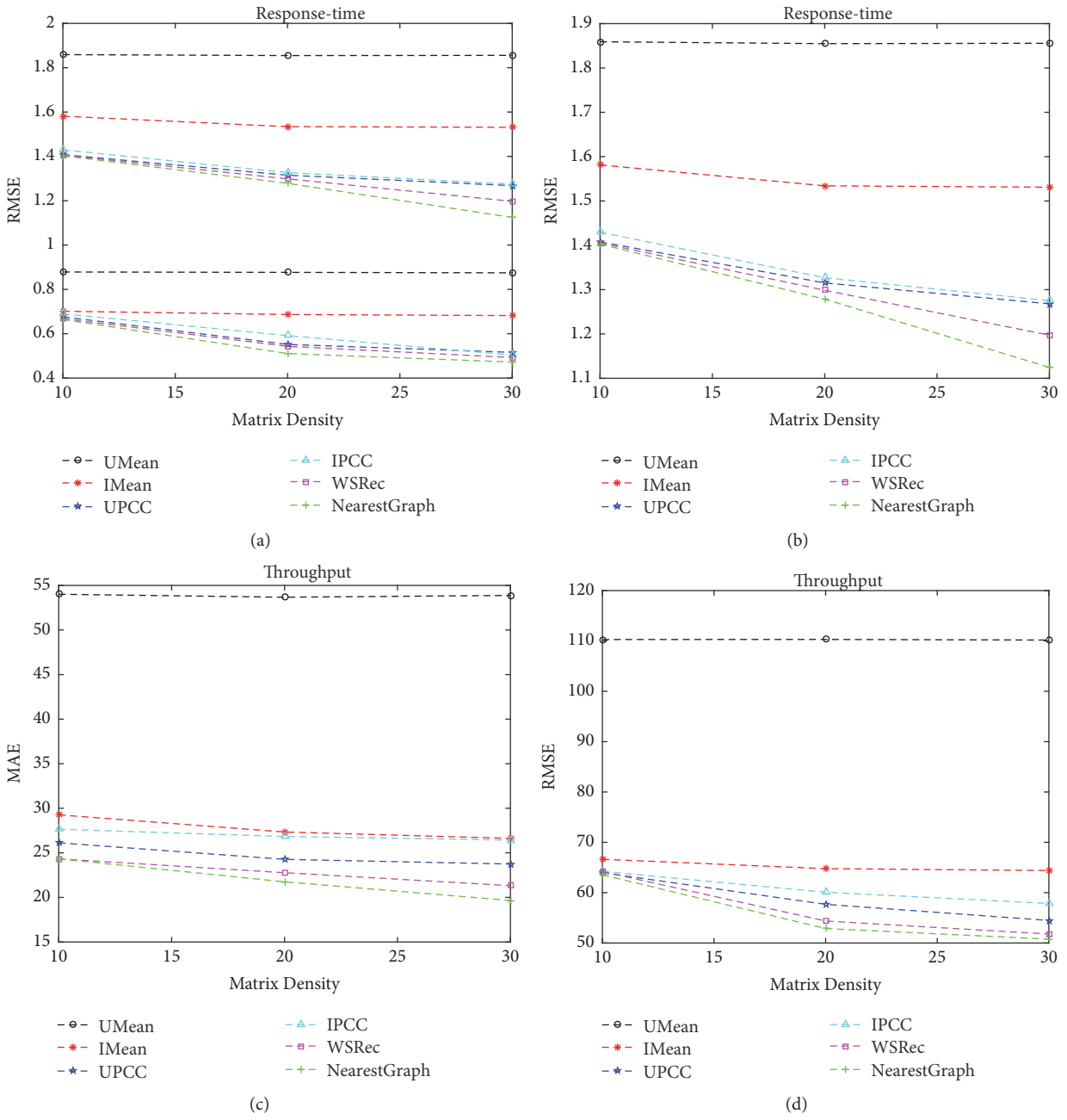


Figure 6: Impact of matrix density.

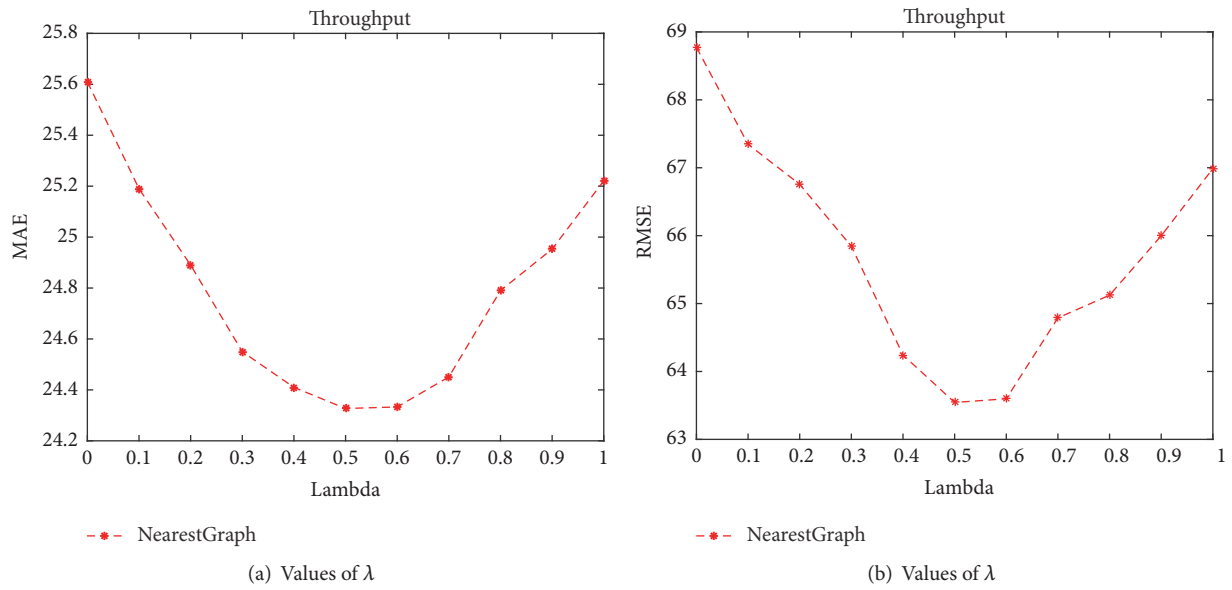
The key point of our approach proposed on the neighbor-based method is the construction of nearest neighbor graph which is designed to expose stable and popular candidates, and the choice of making prediction in a certain order, which applies priorities to different candidates instead of traversing candidates in random to promote the final accuracy. Through a set of experiments on a real world distributed service quality dataset *WS-DREAM* for stimulating the fog cloud environment, we validate the feasibility of our method in terms of service recommendation accuracy and confirm the motivation that NearestGraph can get a good performance in large fluctuation of QoS properties. In summary, the paper makes the following key contributions:

(1) We emphasize the fact of real world QoS values fluctuation in a wide range and take it into account to solve the inaccuracy of predicting missing values.

(2) We reveal the inner features of candidates behind neighbors and take their outer characteristic, stability, and popularity, in the fog cloud environment by constructing the nearest neighbor graph.

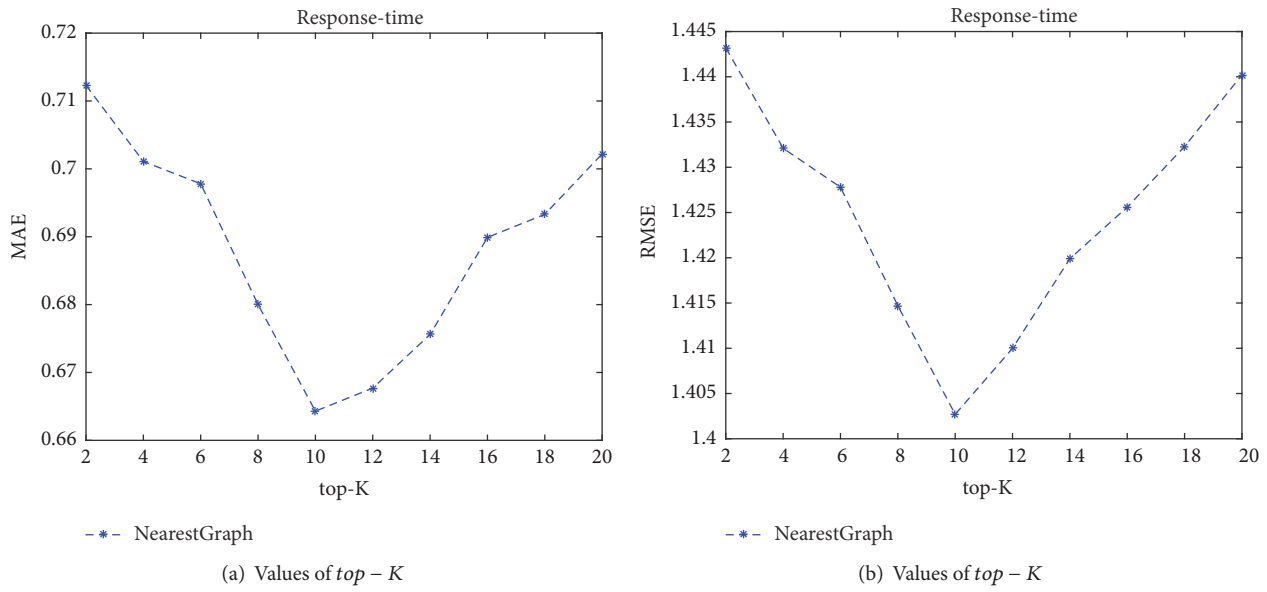
(3) Graph structure is employed to develop prediction order and enhance prediction accuracy.

Currently we predict the values of different QoS attributes separately. And we are going to investigate on the correlations



(a) Values of λ (b) Values of λ

FIGURE 7: Impact of λ.



(a) Values of $top - K$ (b) Values of $top - K$

FIGURE 8: Impact of $top - K$.

and combinations on the QoS attributes in the future. Furthermore, we will use time series analysis for prediction and extend NearestGraph to describe accurate user and service status in the fog cloud environment.

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

The authors acknowledge the support received from the National Natural Science Foundation of China (61300176, 61473031, 61472029, and 61672086) and Fundamental Research Funds for the Central Universities (2016JBZ005).

References

[1] T. H. Luan, L. Gao, Z. Li, Y. Xiang, G. Wei, and L. Sunm, "Fog computing: Focusing on mobile users at the edge," 2015, https://arxiv.org/abs/1502.01815.

[2] S. K. Datta, C. Bonnet, and J. Haerri, "Fog Computing architecture to enable consumer centric Internet of Things services," in

Proceedings of the IEEE International Symposium on Consumer Electronics, ISCE 2015, Spain, June 2015.

[3] D. U. Gamage, "QoS and trust prediction framework for composed distributed systems, 2016".

[4] A. Yousefpour, A. Patil, G. Ishigaki et al., "QoS-aware Dynamic Fog Service Provisioning," 2018, https://arxiv.org/abs/1804.01796.

[5] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "QoS-aware web service recommendation by collaborative filtering," *IEEE Transactions on Services Computing*, vol. 4, no. 2, pp. 140–152, 2011.

[6] H. Yan and L. Zhi-Zhong, "Research on dynamic prediction method of QoS of cloud service," *Journal of Software Engineering*, vol. 11, pp. 1–11, 2017.

[7] D. Roongpiboonsopit, "Navigation Recommender: Real-Time iGNSS QoS Prediction for Navigation Services, 2011".

[8] D. A. Menascé, "QoS issues in web services," *IEEE Internet Computing*, vol. 6, no. 6, pp. 72–75, 2002.

[9] X. Wu, "Context-aware cloud service selection model for mobile cloud computing environments," *Wireless Communications and Mobile Computing*, vol. 2018, pp. 1–14, 2018.

[10] Y. Zhao, Z. Li, and X. Chu, *QoS Prediction for the Cloud Service Marketplace: A Grassmann Manifold Approach*, IEEE, 2015.

[11] F. Liu, J. Tong, J. Mao et al., "NIST cloud computing reference architecture," National Institute of Standards and Technology NIST SP 500-292, 2011.

[12] 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*, pp. 13–16, August 2012.

[13] M. B. Senturk, "in Mission Critical Communication Networks," *in Mission Critical Communication Networks*, vol. 76, 2014.

[14] X. Chen, Z. Zheng, Q. Yu, and M. R. Lyu, "Web service recommendation via exploiting location and QoS information," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 7, pp. 1913–1924, 2014.

[15] X. Wang, J. Zhu, Z. Zheng, W. Song, Y. Shen, and M. R. Lyu, "A spatial-temporal qoS prediction approach for time-aware web service recommendation," *ACM Transactions on the Web (TWEB)*, vol. 10, no. 1, article 7, pp. 1–25, 2016.

[16] X. Wu, B. Cheng, and J. Chen, "Collaborative Filtering Service Recommendation Based on a Novel Similarity Computation Method," *IEEE Transactions on Services Computing*, vol. 10, no. 3, pp. 352–365, 2017.

[17] M. Tang, X. Dai, J. Liu, and J. Chen, "Towards a trust evaluation middleware for cloud service selection," *Future Generation Computer Systems*, vol. 74, pp. 302–312, 2017.

[18] K. Su, B. Xiao, B. Liu, H. Zhang, and Z. Zhang, "TAP: A personalized trust-aware QoS prediction approach for web service recommendation," *Knowledge-Based Systems*, vol. 115, pp. 55–65, 2017.

[19] G. White, A. Palade, C. Cabrera, and S. Clarke, "Quantitative Evaluation of QoS Prediction in IoT," in *Proceedings of the 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops, DSN-W 2017*, pp. 61–66, usa, June 2017.

[20] Q. Yu, "CloudRec: a framework for personalized service Recommendation in the Cloud," *Knowledge and Information Systems*, vol. 43, no. 2, pp. 417–443, 2015.

[21] C. Bauckhage, "k-Means Clustering Is Matrix Factorization," 2015, https://arxiv.org/abs/1802.07891.

[22] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proceedings of the 13th International Conference on Neural Information Processing Systems (NIPS' 00)*, pp. 535–541, MIT Press, Denver, Colo, USA, 2000.

[23] J. Zhu, P. He, Z. Zheng, and M. R. Lyu, "Online QoS Prediction for Runtime Service Adaptation via Adaptive Matrix Factorization," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 10, pp. 2911–2924, 2017.

[24] L. Shao, J. Zhang, Y. Wei, J. Zhao, B. Xie, and H. Mei, "Personalized QoS prediction for web services via collaborative filtering," in *Proceedings of the IEEE International Conference on Web Services (ICWS '07)*, pp. 439–446, IEEE, Salt Lake City, Utah, USA, July 2007.

[25] X. Zhu and P. Lu, "A Multi-Dimensional scheduling scheme for QoS-Aware Real-Time Applications on heterogeneous clusters," in *Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications, HPCC 2008*, pp. 205–212, chn, September 2008.

[26] Y. Zhang and M. R. Lyu, *QoS Prediction in Cloud and Service Computing*, SpringerBriefs in Computer Science, Singapore, Singapore, 2017.

[27] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the 10th International Conference on World Wide Web (WWW '01)*, pp. 285–295, 2001.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 8539307, 12 pages
https://doi.org/10.1155/2018/8539307

Research Article

Reliability Analysis for Multipath Communications in Mobile Cloud Computing Architectures

Shiyong Li, Wei Sun, Yaming Zhang, and Haiou Liu

School of Economics and Management, Yanshan University, Qinhuangdao 066004, China

Correspondence should be addressed to Wei Sun; wsun@ysu.edu.cn

Received 4 April 2018; Accepted 20 May 2018; Published 19 June 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Shiyong Li 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.

Mobile cloud computing (MCC) has gained much attention from both academia and industry in recent years. It can support new types of services, such as m-commerce, m-learning, and mobile healthcare, and enrich mobile users' experience and satisfaction by taking full advantage of cloud computing. In MCC architectures multipath communications can be achieved with multihomed mobile devices, so as to utilize multiple paths for data transmission in parallel. They can achieve better utilization of bandwidth resource, split traffic for load balancing, and enhance reliability, fault tolerance, and robustness for applications. However, little attention has been paid to model the reliability of multipath communications in case of path failure. In this paper we investigate the reliability of concurrent multipath communications in MCC architectures and propose two reliability models when paths are failure. One is for static path failure where the failed paths cannot recover for communication in some delay time. The other is for dynamic path failure where the failed paths can recover in some delay time. Finally, numerical results are given to illustrate the reliability of multipath communications.

1. Introduction

With the wide popularity of mobile devices and the explosion of mobile applications, e.g., business, health, games, entertainment, social networking, travel, and news, mobile cloud computing (MCC) is arising and developed as an integration of cloud computing (CC) into the mobile environment. MCC can support new kinds of services such as mobile commerce, m-government, m-learning, and mobile healthcare and promote mobile users to take full advantages of cloud computing. It has become a profitable business option for enterprise since it can reduce the development and implement of mobile applications. And it is used as a new technology for mobile users to achieve rich experience of many mobile services at low cost. Finally it provides a promising solution to achieve green IT for entrepreneurs, engineers, and researchers [1, 2].

MCC has been attracting the attentions from both academia and industry in recent years and some significant surveys are provided, e.g., [3–6]. These surveys offer an extensive summary and review of mobile cloud computing research and highlight the specific concerns in mobile cloud computing. They also present a taxonomy based on the key issues in this area and discuss the different approaches taken to tackle these issues. Furthermore, they give a critical analysis of challenges which have not yet been fully met and highlight some directions for future work. Recently a new computing paradigm, known as fog computing and further mobile fog computing, has been proposed as an improvement to the cloud computing. Fog computing expands the cloud services to the edge of cloud networks and makes computation, communication, and storage closer to edge devices and end-users [7]. In the research surveys [8–10], the authors overview and survey fog computing model architectures, key technologies, and applications. They also provide some challenges and open issues which are worth indepth study and research in further.

In MCC systems or fog computing for mobile applications (e.g., [11]), each mobile device can be multihomed, so as to improve its throughput by allocating the application data over several paths simultaneously, which is known as *multipath communications* enabled by the promising multipath transmission technologies [12]. Generally, multipath

communications can be classified into two types. The first one is *dynamic path communication* where a primary path is used for transmission and alternate paths are adopted in case of traffic saturation or link breakage on the primary path. The second one is *concurrent path communication* where traffic is split and distributed over multiple paths that are node-disjoint in parallel. It is obvious that both types of multipath communications are preferred to single-path cases in many applications as the former could achieve robustness and load balancing and improve reliability. However, fewer researchers pay attention to analyze the reliability of multipath communications and to model the relationship between the reliability and the number of paths. In this paper we consider concurrent multipath communications in MCC and evaluate the communication reliability when some independent paths for a source-destination pair fail during data transmission. We present two reliability models where the failed paths cannot and can recover after some delay time, respectively, and deduce the probability of successful communication for an application in each model.

We end this section with a short overview of the rest of this paper. Section 2 reviews related work on multipath communications technologies. Section 3 discusses concurrent multipath communications in MCC. Section 4 introduces the reliability models for concurrent multipath communications in MCC. Section 5 presents the numerical results to illustrate the reliability analysis. Finally, conclusions are summarized in Section 6.

2. Related Work

Recent years have seen the increasing attention in the field of multipath communications that utilize multiple paths in parallel and split traffic for load balancing [13–16] and for avoiding DDoS attack in MCC [12]. It seems obvious that using multipath communications could generally increase the available bandwidth for applications [17]. More importantly, they can bring enhancements to the connection persistence, reliability, and fault tolerance, e.g., [18, 19]. They have been found to be useful in many scenario such as communication security in MCC [12], fault-aware resource allocation [20], high-availability virtual communication [21], connection management for identifier-based network [22], and edge computing for vehicular networks [23].

Many multipath protocols have been proposed and applied into wired networks and wireless networks. In wired networks, Multipath TCP protocols, e.g., pTCP [24], mTCP [25], MPTCP [26], and energy efficient congestion control for Multipath TCP [27], are a set of extensions of regular TCP that allow one TCP connection to be spread across multiple paths between each pair of source and destination. By striping one flow's packets across multiple paths, they can enhance user experience through improved resilience to network failure and higher throughput. Stream Control Transmission Protocol (SCTP) [28] standardized by the Internet Engineering Task Force (IETF) is a transport protocol that introduces support for transmission over multiple paths. In an SCTP multihomed association, each endpoint can include more than one IP address. Then, at the initialization time, endpoints exchange the lists of their IP addresses. After the destination is multihomed, one of its multiple destination addresses is chosen as the primary path and the others as secondary paths. During data transmission, if the primary path fails, the source will choose an alternative path to resume sending its packets. Its variants which use SCTP multihoming, e.g., Westwood SCTP with partial reliability (W-PR-SCTP) [17], concurrent multipath transfer [29], load-sharing SCTP (LS-SCTP) [30], independent per-path congestion control SCTP (IPCC-SCTP) [31], and application-layer multipath transport control [32], allow the protocols to distribute traffic over more than one path and use multiple end-to-end paths to carry packets from the same connection and with the same source-destination endpoints. Recently, Coudron [33] reviewed the latest developments in multipath communication technologies and presented some novel approaches for multipath communications. Obviously, using multipath communications improves the performance of well-known bandwidth-hungry applications and enhances reliability, robustness, and fault tolerance for applications. Furthermore, multipath protocols have also been applied to file download and resource allocation in peer-to-peer networks, e.g., BitTorrent, eDonkey, and Gnutella. For example, multipath communication schemes have been presented to realize reasonable resource allocation in [34].

In wireless ad hoc networks, ad hoc on-demand distance vector backup routing (AODV-BR) [35] is a protocol that uses backup nodes to provide fault tolerance. The protocol allows multiple paths between a source and its destination per one route discovery, without additional network load. Another protocols, e.g., ad hoc on-demand multipath distance vector (AOMDV) protocol [36] and optimized AOMDV routing protocol [37], also have a hop-by-hop approach to compute the primary route and multiple backup routes in each route discovery. Moreover, some other multipath protocols based on AODV were also proposed, such as ad hoc on-demand distance vector multipath (AODVM) protocol [38] and node-disjoint multipath routing (NDMR) protocol [39]. Meanwhile, other examples of backup route technique were also presented and multiple paths can be maintained between two nodes, e.g., [40, 41], which are derived from the dynamic MANET on-demand (DYMO) protocol. Using these protocols, the built routes and backup routes for each pair of source and destination can be link-disjoint or node-disjoint, which helps the construction of highly robust end-to-end communication for applications. Recently, in order to build reliable routing, some trust routing schemes were proposed in ad hoc networks, e.g., [42–45], which are the extensions of popular on-demand routing protocols such as the dynamic source routing (DSR) [46]. These proposed routing schemes can behave well in warding off black hole and changing behavior attacks.

Computer networks are known to be fundamental to communications systems. Therefore, it is very important to develop the principles of reliability and availability analysis for computer networks. Shooman [47] developed reliability and availability prediction and optimization methods and applied these techniques to a selection of fault-tolerant systems. Later, Abd-El-Barr [48] introduced the design and

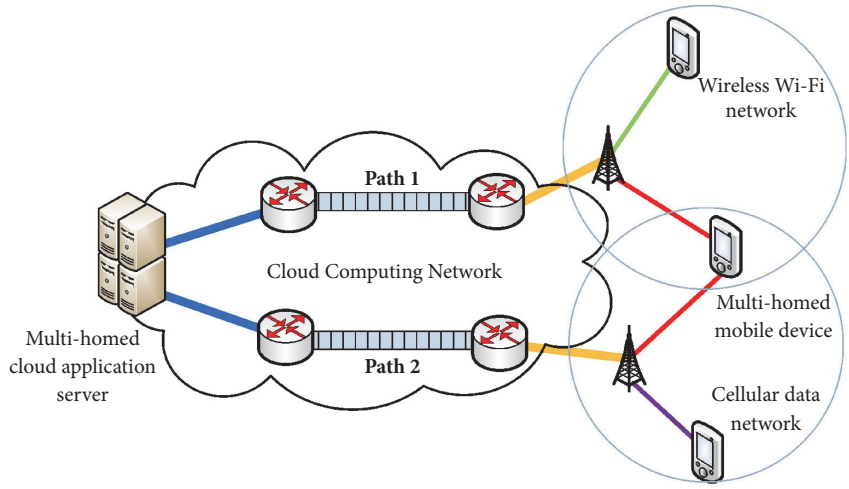


Figure 1: An example of multipath communications in a mobile cloud network (a copy from [12]).

analysis of reliable and fault-tolerant computer systems and discussed the main issues related to redundancy, including hardware, software, time, and information redundancies. Lin et al. [49] investigated an evaluation method for network reliability in ad hoc networks and gave some numerical examples to illustrate the performance. In this paper we consider reliability for concurrent multipath communications in MCC architectures and present two kinds of reliability models: one is static for path failure and the other is dynamic for path failure and recovery. We also give some numerical examples to illustrate the performance.

3. Multipath Communications in MCC

Multipath communications have become very attractive since they can achieve better robustness and reliability than single-path cases. Among these schemes, concurrent multipath routing schemes where paths for a source-destination pair do not share any common links gain a lot of attention. As shown in Figure 1 (a copy from [12]), in MCC systems each mobile device is multihomed through the promising multipath technologies, such as MPTCP and SCTP. There are multiple paths between each pair of mobile user and the server. This improves the user's throughput by allocating the application data over several paths simultaneously. More importantly, this bring enhancements to the connection persistence, security, reliability, and fault tolerance.

Further, we consider the different types of multipath communications. In the work [50] a multipath routing scheme modified from single-path AODV was proposed for MANET so as to reduce the effect of frequent communication failures. The proposed scheme is basically proposed for highly dynamic ad hoc networks where communication failures occur frequently and designed to compute not only node-disjoint paths but also fail-safe paths between each source-destination pair [51]. In this work different types of multiple paths are reviewed, which are shown in Figure 2 (a copy from [50]).

For a source-destination pair, node-disjoint paths do not share any nodes in common, except the source and destination, while link-disjoint paths do not have any links in common; however, they may share some intermediate nodes on the paths. Unlike node-disjoint and link-disjoint paths, fail-safe path between the source-destination pair bypasses at least one intermediate node on the primary path, which is the shortest path between the source and destination [50]. Thus, fail-safe paths can share both nodes and links in common, just as shown in Figure 2.

Among the three kinds of multipath scenarios for a source-destination pair, paths are independent from each other in the first case and do not own any common nodes or links. The paths are regarded to be *concurrent*. Thus, communication failure on one of them has no influence on others. However, in the second case, two paths share some common nodes which are regarded as *hot-nodes* for forwarding data packet. Failure on one of the shared nodes can result in communication failures on multiple paths and even failure of the communication progress between the source and destination. The third case is the most complicated one. Paths share nodes and links in common, which can enhance the robustness and reliability of communication. When some shared nodes are failure, the source can continue to communicate the destination by bypassing them. Reliability analysis of communication in link-disjoint or fail-safe case is more complicated than that in node-disjoint case since they share common nodes. Thus, as an attempt to analyze the reliability of multipath communications theoretically, we concentrate on concurrent multipath communications and present two reliability models for them.

In this paper, we consider a network consisting of a set of sources and destinations, whereby each source can send data to its destination over multiple independent paths. Thus an application can be completed as long as there exists at least an available path between the pair of source and its destination. Here, the paths for a source-destination pair are all assumed to be concurrent and node-disjoint; that is, they

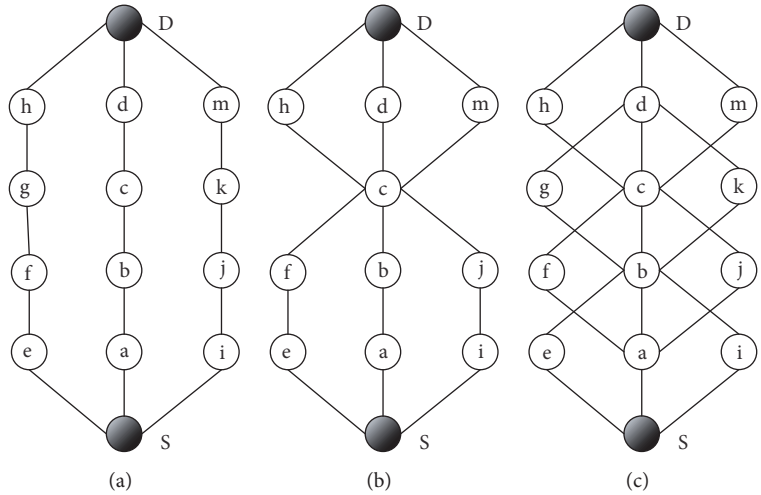


Figure 2: Different types of multiple paths: (a) node-disjoint, (b) link-disjoint, and (c) fail-safe.

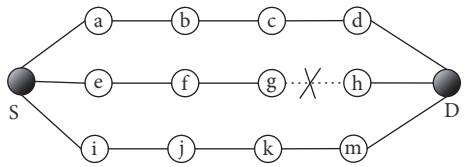


Figure 3: Static failure.

do not share any common node or link with each other, as shown in Figure 2(a).

4. Reliability Models

In this section we develop analytical models to evaluate the reliability of concurrent multipath communications in MCC in the face of path failures which may be caused by attacks on nodes or heavy congestion on bottleneck links on the paths. We assume that the communication for an application between two nodes can be completed successfully so long as there is at least one available path which is not failed. Our analysis begins by considering the simple case: the paths for a source-destination pair have static failures.

4.1. Static Failure. Firstly, we consider static failure where the failed paths cannot recover for communication after some delay time. As an example shown in Figure 3, there are three node-disjoint paths between the source S and destination D. At some time, the path e $\rightarrow$ f $\rightarrow$ g $\rightarrow$ h fails due to attacks on node g or heavy congestion on link g $\rightarrow$ h. Then the source will choose the remaining ones for communication and continue sending packets to the destination.

Let N be the number of all available node-disjoint paths for a source-destination pair, $M(\leq N)$ be the number of node-disjoint paths that the pair actually chooses and uses during data transmission, and $K(\leq N)$ be the number of failed paths for the pair. That is, among the available node-disjoint paths that the pair builds and maintains, the pair only chooses M of them for communication. Meanwhile, some of the paths may fail during the maintenance.

Let $P(n, m, k)$ be the probability that a set of k paths selected at random from n paths contains a specific subset of m paths. It is easy to obtain that

$$P(n, m, k) = \frac{C_k^m}{C_n^m}, \tag{1}$$

when $k \geq m$, and $P(n, m, k) = 0$ when $k < m$, where

$$C_a^b = \binom{a}{b} = \frac{a!}{b!\,(a-b)!}. \tag{2}$$

Thus, for a source-destination pair, the probability that an application can be completed successfully is

$$\mathbf{P} = 1 - P(N, M, K) = 1 - \frac{C_K^M}{C_N^M}. \tag{3}$$

Obviously,

$$\begin{aligned} \frac{P(N, M, K)}{P(N-1, M, K)} &= 1 - \frac{M}{N} \leq 1, \\ \frac{P(N, M+1, K)}{P(N, M, K)} &= \frac{1-(N-K)}{(N-M) \leq 1}, \end{aligned} \tag{4}$$

and

$$\frac{P(N, M, K)}{P(N, M, K-1)} = 1 + \frac{M}{(K-M)} > 1 \tag{5}$$

when $K \geq M$. Hence, an increase in the number of available paths N for the pair of source and destination or the number of paths M actually used by the source can increase the probability of successful communication for the application. Similarly, a decrease in the number of failure paths K can also increase the probability.

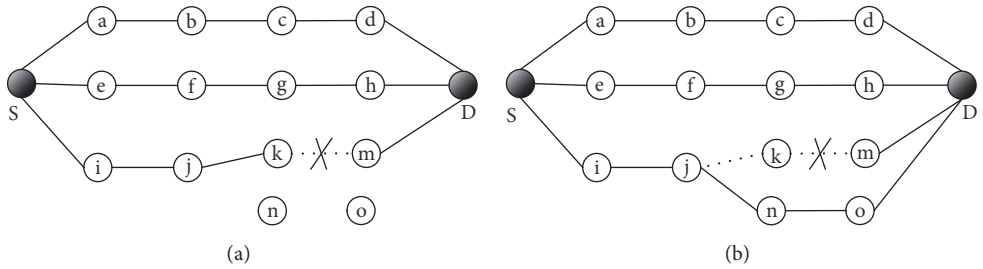


Figure 4: Dynamic failure: (a) before recovery; (b) after recovery.

4.2. Dynamic Failure and Recovery. Our previous model assumes that once the paths fail, they do not recover after some time. However, the nodes on the failed paths can take repairing and recovering actions so that new available paths can be established after some delay time. Hence, we extend the model to the case where the failed paths can take some repairing actions and recover the ability of communication for applications.

As an example shown in Figure 4, among the three node-disjoint paths for the pair of source S and destination D, the path i $\rightarrow$ j $\rightarrow$ k $\rightarrow$ m fails because of attacks on node k or heavy congestion on link k $\rightarrow$ m; then after some time source S detects it and reestablishes a new available path, i.e., i $\rightarrow$ j $\rightarrow$ n $\rightarrow$ o.

We assume that for a source-destination pair there is a failure delay or an attack delay D_f which is the difference in time from when an available path is first established to the time when the path is failed because of attacks on nodes or heavy congestion on bottleneck links on the path. Also there is a recovery delay D_r that equals the difference in time between when the source discovers the failed path to the time when it reestablishes a new available path.

Since there are so many types of attacks in networks, e.g., black hole attacks, rushing attack, and worm attack [50], we do not yet have a detailed understanding of how the failure and recovery processes will perform. Therefore, we do not have models that accurately capture the distributions of D_f (failure delay) and D_r (recovery delay). However, we are interested in gaining preliminary insight into how the failure of paths affects the reliability of concurrent multipath communications. We realize this insight by modeling the framework as a closed queueing system with a finite customer population. In the queueing system customers arrive at the server(s), obtain service, and then, after a certain delay, return to get serviced again. Thus, the number of jobs active in the queueing system equals the number of paths under attack that are to be failed. The recovery process removes jobs from the system and attacks cause in the path filed, resulting in placing jobs back in the system.

As an interesting method to evaluate the denial of service (DoS) attacks in computer networks, a two-dimensional embedded Markov chain model is presented in [52] to characterize the network under DoS attacks. The arrivals of the regular request packets and the attack packets are both Poisson processes, and they are independent of each other. Thus, similar to the queueing analysis for attacks in [52], we suppose that both D_f and D_r are exponentially distributed variables with respective rates λ and μ.

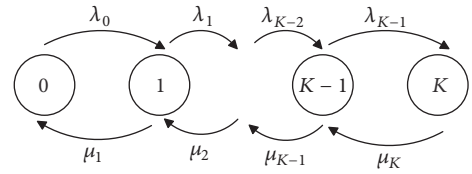


Figure 5: State transition diagram.

We are interested in two variants of modeling the failed path recovery process. In the first, the ability to detect and recover the failed paths is performed sequentially. This would occur when the detection and recovery of failed paths is made one after another by the source. We refer to this variant as the centralized recovery process. Alternatively, the other one is distributed recovery process, where recovery of failed paths can be performed in parallel. This would occur when each path can independently perform its recovery process. Similarly, the failure process can be centralized, where available paths for one source are failed one after another, or distributed, where all available paths would be failed in parallel.

For a source-destination pair, we define a random variable $F(t)$ to be the number of failure paths on which the nodes are under attacks or the links have heavy congestion at time t. Let K be the maximal number of failure paths; thus $F(t)$ is up to K, i.e., $0 \leq F(t) \leq K$.

Given that both the failure and the recovery process can be either centralized or distributed, there are four different scenarios. Each scenario is indeed a queueing model with $K + 1$ states where the process resides in state i when there are i paths that are failed because of attacks on nodes. The state transition diagram of each of the four models can be shown in Figure 5. When the paths failure is centralized, the transition rate from state i to $i + 1$ is λ; while the paths failure is distributed, the rate is $(K - i)\lambda$. When the paths recovery is centralized, the transition rate from state i to $i - 1$ is μ; while in the distributed case, the rate is $i\mu$.

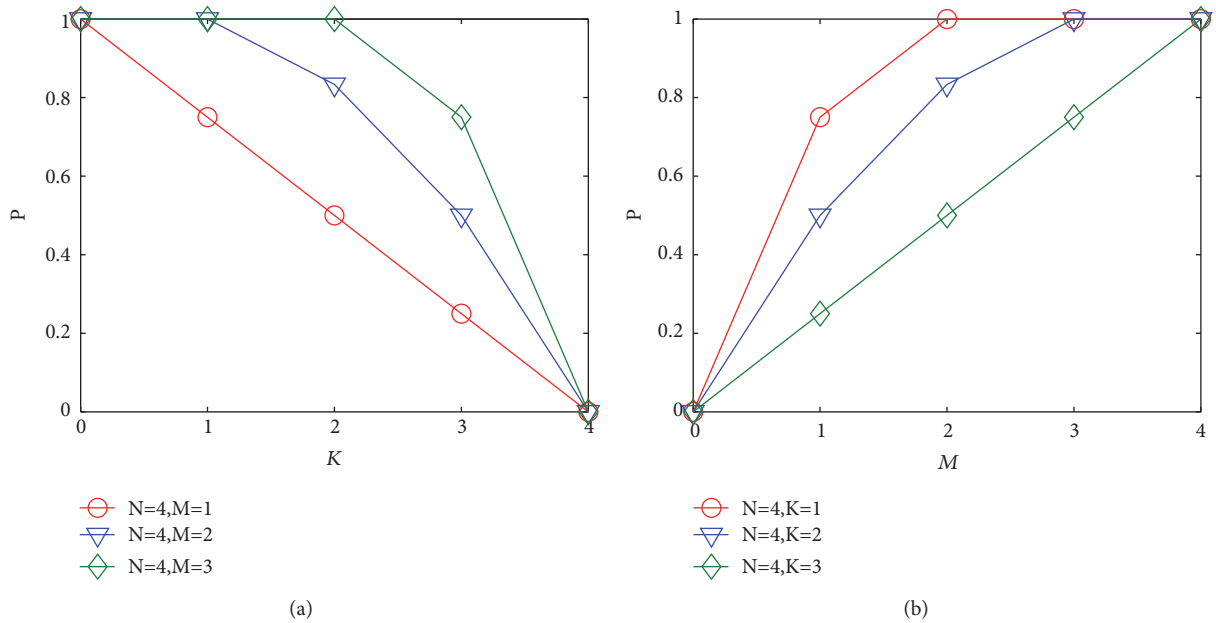


FIGURE 6: Reliability in the static failure case: (a) the relationship between **P** and K; (b) the relationship between **P** and M.

Let $\rho = \lambda/\mu$; then the probability $\pi_i = \Pr[F(t) = i]$ in the four scenarios can be summarized as follows:

(a) centralized failure, centralized recovery

$$\pi_i = \frac{1-\rho}{1-\rho^{K+1}}\rho^i, \tag{6}$$

(b) centralized failure, distributed recovery

$$\pi_i = \frac{\rho^i/i!}{\sum_{j=0}^{K}\left(\rho^j/j!\right)}, \tag{7}$$

(c) distributed failure, centralized recovery

$$\pi_i = \frac{\rho^i/(K-i)!}{\sum_{j=0}^{K}\left(\rho^j/(K-j)!\right)}, \tag{8}$$

(d) distributed failure, distributed recovery

$$\pi_i = \frac{\rho^i/i!\,(K-i)!}{\sum_{j=0}^{K}\left(\rho^j/j!\,(K-j)!\right)}. \tag{9}$$

Notice that in the four scenarios of failure and recovery process the probability π_i depends on both the failure rate ρ on paths and the maximal number of failure paths K.

Let N be the number of all *available* node-disjoint paths for the pair of source and destination and $M(\leq N)$ be the number of node-disjoint paths that the pair *actually chooses* for communication. Thus, the probability that an application between the source and destination can be completed successfully is

$$\mathbf{P} = \sum_{i=0}^{K}\pi_i\left(1 - P\left(N+i-K, M, i\right)\right), \tag{10}$$

where $P(N+i-K, M, i) = C_i^M/C_{N+i-K}^M$ when $i \geq M$ and 0 otherwise.

Intuitively, an increase in failure rate ρ decreases the successful probability **P**. For a fixed failure rate ρ, not surprisingly, increasing the number of available paths N or the number of actually used paths M or decreasing the maximal number of failure paths K can increase the successful probability **P**.

5. Numerical Examples and Analysis

In this section we consider a scenario of concurrent multipath communication as shown in Figure 2(a) and give some numerical examples to illustrate the reliability models for concurrent multipath communications. We also present analysis for the relationships between the successful probability **P** and the parameters N, M, K, and ρ.

5.1. Static Failure. Suppose that there are four available node-disjoint paths between each source and its destination. The communication for an application between the source and destination can be completed successfully even if there is only one available path which is not failed. Obviously in Figure 6, an increase in the maximal number of failure paths K can decrease the probability of successful communication for the application using concurrent multiple paths, while an increase in the number of actually used paths M can increase the probability. For example, the successful probability **P** is increased from 0.5 to 0.8333 when an increase in the number of actually used paths M from 1 to 2 for fixed number of available paths $N = 4$ and maximal number of failure paths $K = 2$.

5.2. Dynamic Failure and Recovery. In this part we further consider the concurrent multipath communication shown

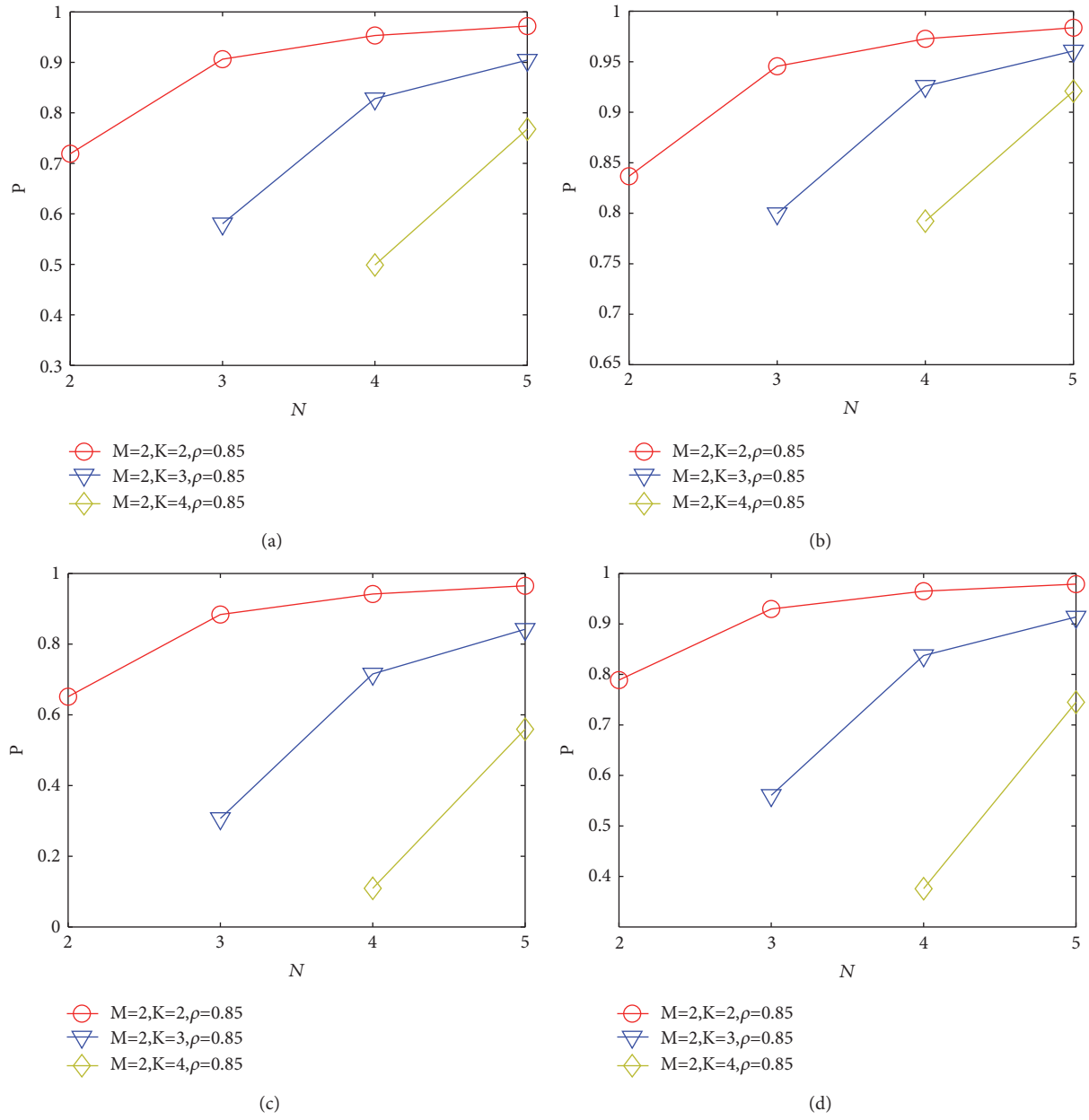


FIGURE 7: The relationship between **P** and N in the dynamic failure and recovery case: (a) centralized failure, centralized recovery; (b) centralized failure, distributed recovery; (c) distributed failure, centralized recovery; (d) distributed failure, distributed recovery.

in Figure 2(a) and investigate the relationships between **P** and N, M, K, and ρ when the failed paths can recover for communication after some delay time. In each example, the reliability performances in the four scenarios are listed in sequence as follows: (a) centralized failure, centralized recovery; (b) centralized failure, distributed recovery; (c) distributed failure, centralized recovery; (d) distributed failure, distributed recovery.

5.2.1. Relationship between **P** *and* N. Suppose for the pair of source and destination the number of actually used paths M is 2, the maximal number of failure paths K varies from 2 to 4, and the failure rate ρ is 0.85. As shown in Figure 7, in each scenario, for fixed M, K, and ρ, an increase in the number of available paths N for the pair of source and destination remarkably improves the successful probability **P**, since the likelihood for each source to select multiple available paths for communication increases. And the reliability performance is better when the failure process is centralized and the recovery process is distributed than that when the failure process is distributed and the recovery process is centralized. For example, when $N = 4, M = 2, K = 3, \rho = 0.85$, the successful probability **P** is 0.9258 in the former while it is only 0.7159 in the latter.

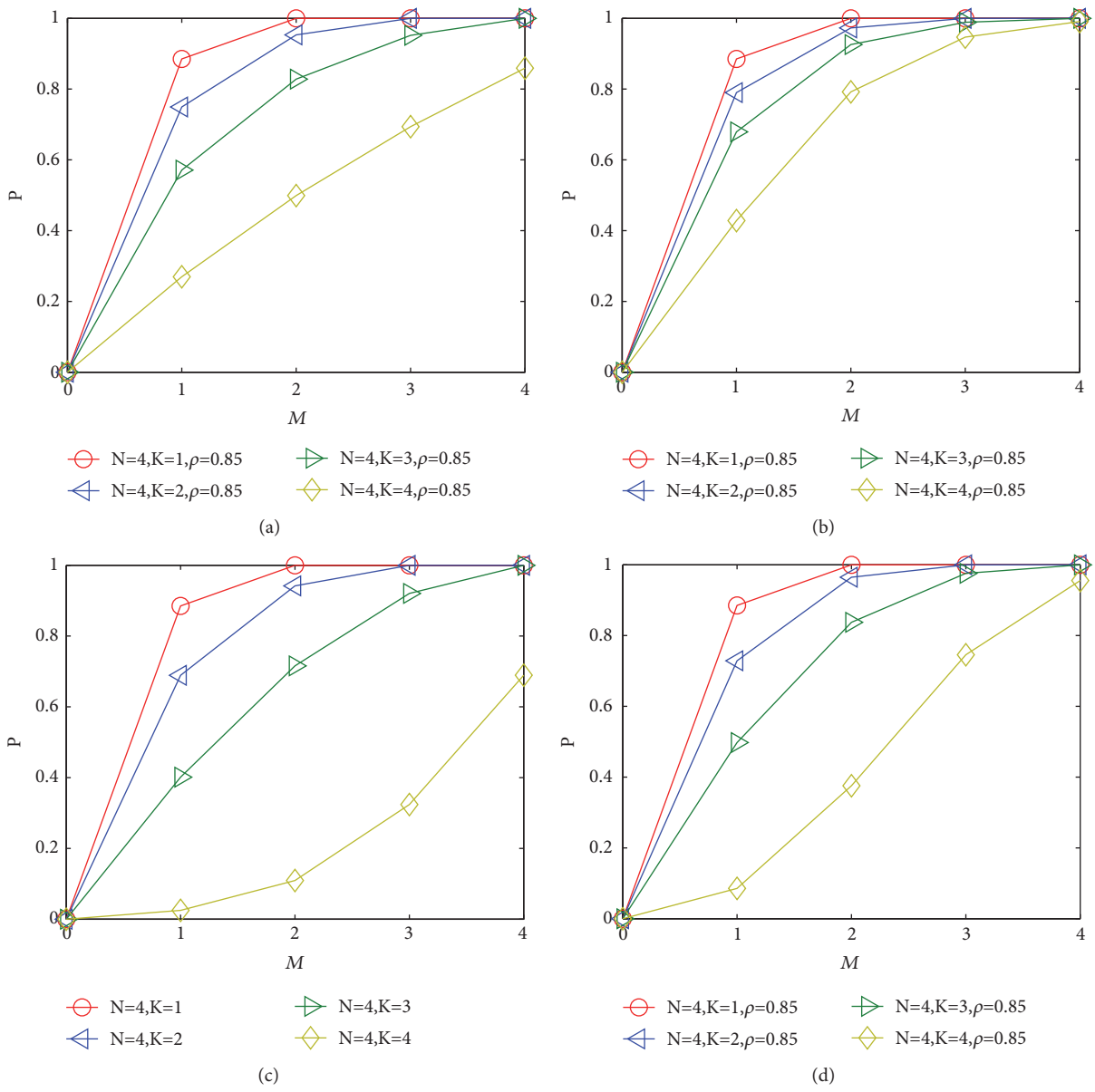


Figure 8: The relationship between **P** and M in the dynamic failure and recovery case.

5.2.2. Relationship between **P** *and* M. Suppose for the pair of source and destination the number of available paths N is 4, the maximal number of failure paths K varies from 1 to 4, and the failure rate ρ is 0.85. As shown in Figure 8, for fixed N, K, and ρ, the successful probability **P** significantly increases with an increase in the number of paths that the pair of source and destination actually chooses for communication. Obviously, applications are most likely to be completed when the failure process is centralized and the recovery process is distributed and they are least likely to be completed when the failure is distributed and the recovery is centralized. That is, the reliability performance of the former is better than that of the latter. For example, when $N = 4, M = 3, K = 3, \rho = 0.85$, the successful probability **P** is 0.9889 in the former while it is only 0.9204 in the latter. This result can be understood intuitively by comparing the respective birth-death processes of the system in the aforementioned two cases. In the former one, the upward transition rate is λ and the downward transition rate is $i\mu$, which is larger for state with larger i, whereas in the latter, the upward transition rate is $(K - i)\lambda$, which is larger for state with smaller i and the downward transition rate is μ.

5.2.3. Relationship between **P** *and* K. Suppose for the pair of source and destination the number of available paths N is 4, the number of actually used paths M varies from 1 to 4, and the failure rate ρ is 0.85. As shown in Figure 9, the probability **P** significantly decreases with an increase in the maximal number of failure paths K. Similarly, among the four different cases for path failure and recovery, the second one

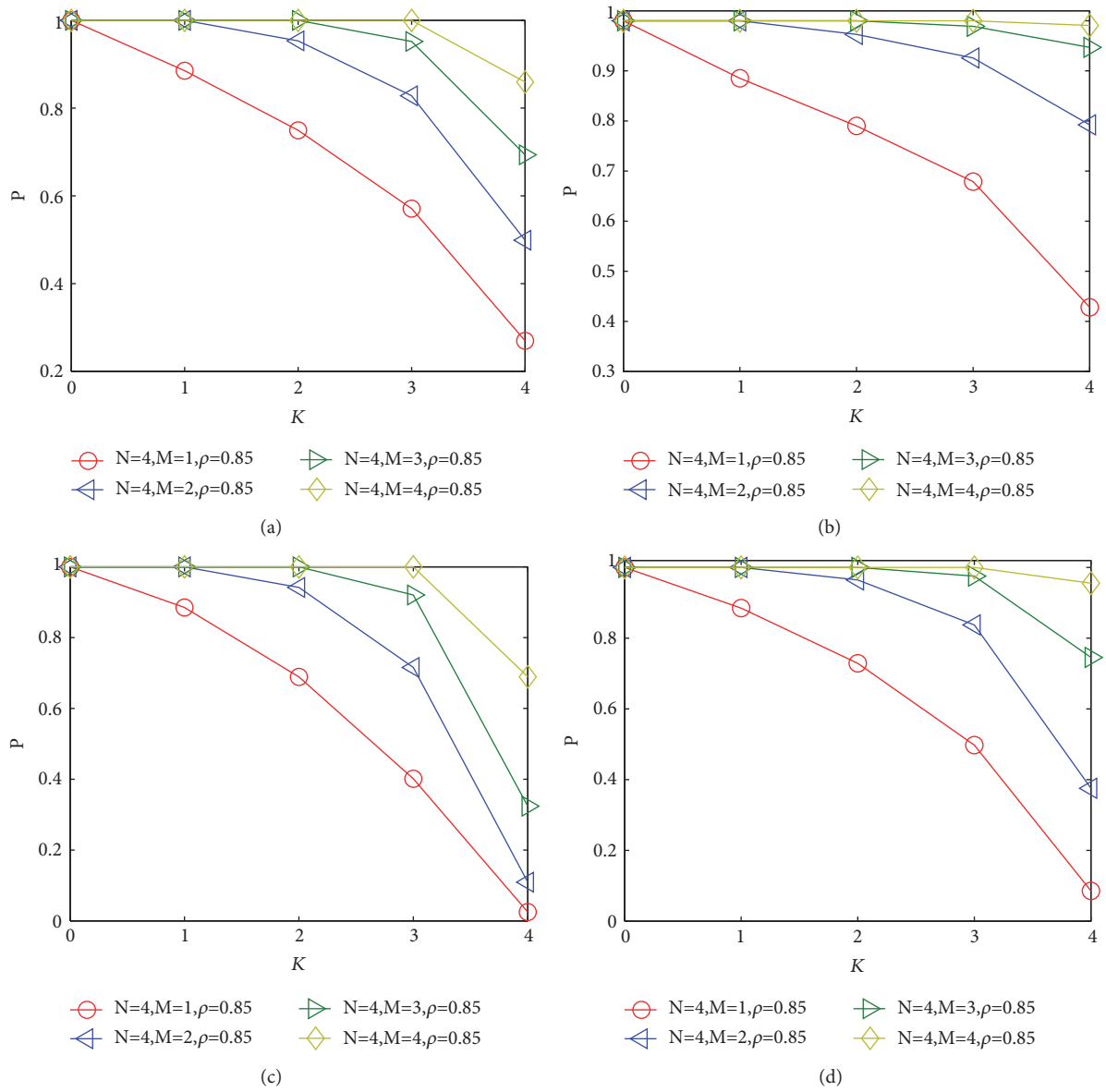


Figure 9: The relationship between **P** and K in the dynamic failure and recovery case.

is the best while the third one is the worst under the same network condition. Thus, in order to improve the likelihood of successful communication, distributed effective detection methods are highly suggested such that the failed paths can recover in a distributed way.

5.2.4. Relationship between **P** *and* ρ. Suppose for the pair of source and destination the number of available paths N is 4, the number of actually used paths M is 2, and the maximal number of failure paths K varies from 2 to 4. In Figure 10, we plot the probability **P** varying along with ρ. Obviously, **P** decreases as ρ grows. As ρ approximates to 1, **P** diminishes less when the failure process is centralized and the recovery process is distributed than that when the failure is distributed and the recovery is centralized or distributed. Thus, applications are most likely to be completed when the failure process is centralized and the recovery process is distributed.

From the results above in the four scenarios of failure and recovery process, we can obtain that reliability achieves much better when the recovery process is distributed; thus in order to improve the likelihood of successful communication between the source and destination, distributed effective detection methods are highly suggested such that the failed paths can recover in a distributed way.

6. Conclusions

MCC is regarded as an integration of cloud computing into the mobile environment. It provides a powerful tool to the user when and where it is needed irrespective of user movement, hence supporting new kinds of mobile applications

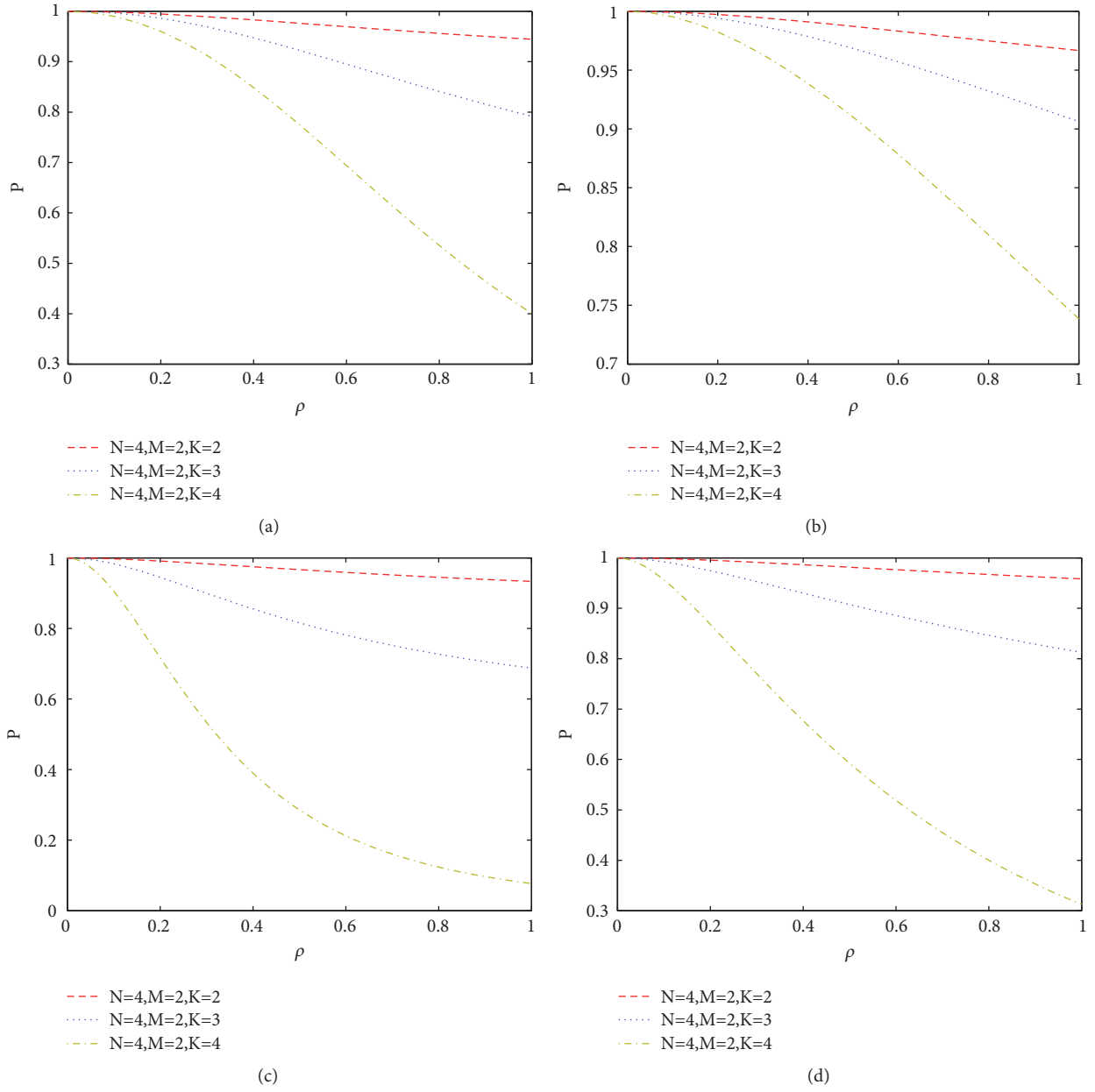


Figure 10: The relationship between **P** and ρ in the dynamic failure and recovery case.

such as m-commerce, mobile healthcare, and mobile social networking. In MCC systems each mobile device can be multihomed so that there are multiple paths between each pair of user and the server. It has been agreed that using concurrent multipath communications could improve the connection persistence, reliability, and fault tolerance between each pair of source and destination. Thus we consider concurrent multipath communications in MCC architectures and investigate the communications reliability when the paths are failed due to attacks. We obtain two kinds of reliability models when the failed paths cannot and can recover after some delay time, respectively. Our analysis demonstrates that using concurrent multipath communications generally improves the likelihood of successful communication for an application. Meanwhile, when communication paths are failed, distributed effective detection and quick recovery schemes should be highly guaranteed, so as to ensure high reliability requirements for communications of mobile applications.

Data Availability

The authors confirm that the data supporting the findings of this study are available within the article.

Conflicts of Interest

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

Acknowledgments

The authors would like to thank the support from the National Natural Science Foundation of China (nos. 71671159, 71301139, and 71271186), the Humanity and Social Science Foundation of the Ministry of Education of China (no. 16YJC630106), the project funded by Four Batch of Talents Program in Hebei Province, the Natural Science Foundation of Hebei Province (nos. G2018203302, G2016203236, G2016203220), the project funded by Hebei Education Department (nos. BJ2017029, BJ2016063), and Hebei Talents Program (no. A2017002108).

References

[1] M. Ali, "Green cloud on the horizon," in *Proceedings of the 1st International Conference on Cloud Computing (CloudCom '09)*, pp. 451–459, Manila, Philippines, 2009.

[2] L. Yang, J. Cao, Y. Yuan, T. Li, A. Han, and A. Chan, "A framework for partitioning and execution of data stream applications in mobile cloud computing," *Performance Evaluation Review*, vol. 40, no. 4, pp. 23–32, 2013.

[3] A. Abunaser and S. Alshattnawi, "Mobile cloud computing and other mobile technologies: Survey," *Journal of Mobile Multimedia*, vol. 8, no. 4, pp. 241–252, 2013.

[4] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," *Wireless Communications and Mobile Computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[5] A. Khan, M. Othman, S. A. Madani, and S. U. Khan, "A survey of mobile cloud computing application models," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 393–413, 2014.

[6] Y. Wang, I.-R. Chen, and D.-C. Wang, "A survey of mobile cloud computing applications: perspectives and challenges," *Wireless Personal Communications*, vol. 80, no. 4, pp. 1607–1623, 2015.

[7] F. Song, Z.-Y. Ai, J.-J. Li et al., "Smart collaborative caching for information-centric IoT in fog computing," *Sensors*, vol. 17, no. 11, p. 2512, 2017.

[8] P. Hu, S. Dhelim, H. Ning, and T. Qiu, "Survey on fog computing: architecture, key technologies, applications and open issues," *Journal of Network and Computer Applications*, vol. 98, pp. 27–42, 2017.

[9] O. Osanaiye, S. Chen, Z. Yan, R. Lu, K.-K. R. Choo, and M. Dlodlo, "From cloud to fog computing: a review and a conceptual live VM migration framework," *IEEE Access*, vol. 5, pp. 8284–8300, 2017.

[10] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, Fog et al.: a survey and analysis of security threats and challenges," *Future Generation Computer Systems*, vol. 78, pp. 680–698, 2018.

[11] L. F. Bittencourt, J. Diaz-Montes, R. Buyya, O. F. Rana, and M. Parashar, "Mobility-aware application scheduling in fog computing," *IEEE Cloud Computing*, vol. 4, no. 2, pp. 26–35, 2017.

[12] Y. Cao, F. Song, Q. Liu, M. Huang, H. Wang, and I. You, "A LDDoS-Aware energy-efficient multipathing scheme for mobile cloud computing systems," *IEEE Access*, vol. 5, pp. 21862–21872, 2017.

[13] S. Li, W. Sun, and C. Hua, "Fair resource allocation and stability for communication networks with multipath routing," *International Journal of Systems Science*, vol. 45, no. 11, pp. 2342–2353, 2014.

[14] S. Li, W. Sun, and N. Tian, "Resource allocation for multi-class services in multipath networks," *Performance Evaluation*, vol. 92, pp. 1–23, 2015.

[15] F. Song, Y. Zhang, Z. An, H. Zhou, and I. You, "The correlation study for parameters in four tuples," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 19, no. 1-2, pp. 38–49, 2015.

[16] S. Li, W. Sun, and C. Hua, "Optimal resource allocation for heterogeneous traffic in multipath networks," *International Journal of Communication Systems*, vol. 29, no. 1, pp. 84–98, 2016.

[17] M. Fiore, C. Casetti, and G. Galante, "Concurrent multipath communication for real-time traffic," *Computer Communications*, vol. 30, no. 17, pp. 3307–3320, 2007.

[18] F. Song, H. Zhou, S. Zhang, H. Zhang, and I. You, "The throughput critical condition study for reliable multipath transport," *Computer Science and Information Systems*, vol. 10, no. 2, pp. 567–587, 2013.

[19] F. Song, R. Li, and H. Zhou, "Feasibility and issues for establishing network-based carpooling scheme," *Pervasive and Mobile Computing*, vol. 24, pp. 4–15, 2015.

[20] X. Zhang, Q. Chen, Z. Shi, and J. Liang, "Fault-Aware resource allocation for heterogeneous data sources with multipath routing," *Scientific Programming*, vol. 2017, Article ID 9749581, 12 pages, 2017.

[21] S. Sirisutthidecha and K. Maichalernnukul, "High-availability virtual communication for cloud access," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 8, pp. 3455–3473, 2016.

[22] F. Song, Y.-T. Zhou, K. Kong, Q. Zheng, I. You, and H.-K. Zhang, "Smart collaborative connection management for identifier-based network," *IEEE Access*, vol. 5, pp. 7936–7949, 2017.

[23] K. Wang, H. Yin, W. Quan, and G. Min, "Enabling collaborative edge computing for software defined vehicular networks," *IEEE Network*, vol. 32, pp. 1–6, 2018.

[24] H.-Y. Hsieh and R. Sivakumar, "A transport layer approach for achieving aggregate bandwidths on multi-homed mobile hosts," in *Proceedings of the IEEE MOBICOM 2002*, Atlanta, Ga, USA, September 2002.

[25] M. Zhang, J. Lai, A. Krishnamurthy, L. Peterson, and R. Wang, "A transport layer approach for improving end-to-end performance and robustness using redundant paths," in *Proceedings of the USENIX*, Boston, Mass, USA, 2004.

[26] A. Ford, C. Raiciu, S. Barre, and J. Iyengar, Architectural Guidelines for Multipath TCP Development, draft-ietf-mptcp-architecture-00, 2010.

[27] W. Wang, X. Wang, and D. Wang, "Energy efficient congestion control for multipath TCP in heterogeneous networks," *IEEE Access*, vol. 6, pp. 2889–2898, 2017.

[28] R. Stewart, Q. Xie, K. Morneault, and C. Sharp, "Stream Control Transmission Protocol (SCTP)," RFC 2960, IETF, 2000.

[29] J. R. Iyengar, P. D. Amer, and R. R. Stewart, "Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths," *IEEE/ACM Transactions on Networking*, vol. 14, no. 5, pp. 951–964, 2006.

[30] A. Abd El Al, T. Saadawi, and M. Lee, "LS-SCTP: A bandwidth aggregation technique for stream control transmission protocol," *Computer Communications*, vol. 27, no. 10, pp. 1012–1024, 2004.

[31] G. Ye, T. N. Saadawi, and M. Lee, "IPCC-SCTP: an enhancement to the standard SCTP to support multi-homing efficiently," in *Proceedings of the IEEE International Conference on Performance, Computing, and Communications (ICPCC '04)*, pp. 523–530, Phoenix, Ariz, USA, 2004.

[32] W. Zhang, W. Lei, Y. Guan, G. Li, and L. Yang, "Considerations for application-layer multipath transport control," *International Journal of Communication Systems*, vol. 30, no. 17, Article ID e3343, 2017.

[33] M. Coudron, *Novel approaches for multipath communications [Thesis for Doctor of Dissertation]*, University Pierre and Marie CURIE, 2016.

[34] S. Li and W. Sun, "A mechanism for resource pricing and fairness in peer-to-peer networks," *Electronic Commerce Research*, vol. 16, no. 4, pp. 425–451, 2016.

[35] S.-J. Lee and M. Gerla, "AODV-BR: Backup routing in ad hoc networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '00)*, pp. 1311–1316, IEEE, Chicago, Ill, USA, September 2000.

[36] M. K. Marina and S. R. Das, "Ad hoc on-demand multipath distance vector routing," *Wireless Communications and Mobile Computing*, vol. 6, no. 7, pp. 969–988, 2006.

[37] Y. Yuan, H. Chen, and M. Jia, "An optimized ad-hoc on-demand multipath distance vector (AOMDV) routing protocol," in *Proceedings of the IEEE Asia-Pacific Conference on Communications*, pp. 569–573, Perth, Australia, October 2005.

[38] Z. Ye, S. V. Krishnamurthy, and S. K. Tripathi, "A routing framework for providing robustness to node failures in mobile ad hoc networks," *Ad Hoc Networks*, vol. 2, no. 1, pp. 87–107, 2004.

[39] X. Li and L. Cuthbert, "On-demand node-disjoint multipath routing in wireless ad hoc networks," in *Proceedings of the IEEE Conference on Local Computer Networks (LCN '04)*, pp. 419-420, Tampa, Fla, USA, November 2004.

[40] J. J. Gálvez and P. M. Ruiz, "Design and performance evaluation of multipath extensions for the DYMO protocol," in *Proceedings of the 32nd IEEE Conference on Local Computer Networks (LCN '07)*, pp. 885–892, October 2007.

[41] G. Koltsidas, F. Pavlidou, K. Kuladinithi, A. Timm-Giel, and C. Gorg, "Investigating the performance of multipath protocol for ad-hoc networks," in *Proceedings of the Annual IEEE International Symposuim on Personal, Indoor and Radio Communications (PIMRC '07)*, pp. 1–5, 2007.

[42] A. A. Pirzada, A. Datta, and C. McDonald, "Incorporating trust and reputation in the DSR protocol for dependable routing," *Computer Communications*, vol. 29, no. 15, pp. 2806–2821, 2006.

[43] S. Peng, W. Jia, G. Wang, J. Wu, and M. Guo, "Trusted routing based on dynamic trust mechanism in mobile ad-hoc networks," *IEICE Transaction on Information and Systems*, vol. E93-D, no. 3, pp. 510–517, 2010.

[44] J. Wang, Y. Liu, and Y. Jiao, "Building a trusted route in a mobile ad hoc network considering communication reliability and path length," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1138–1149, 2011.

[45] M. Sajwan, D. Gosain, and A. K. Sharma, "CAMP: cluster aided multi-path routing protocol for wireless sensor networks," *Wireless Networks*, 2018.

[46] R. Misra and C. R. Mandal, "Performance comparison of AODV/DSR on-demand routing protocols for ad hoc networks in constrained situation," in *Proceedings of the IEEE International Conference on Personal Wireless Communications (PWC '05)*, pp. 86–89, 2005.

[47] M. L. Shooman, *Reliability of Computer Systems and Networks: Fault Tolerance, Analysis, and Design*, John Wiley & Sons, Inc., New York, NY, USA, 2002.

[48] M. Abd-El-Barr, *Design and Analysis of Reliable and Fault-Tolerant Computer Systems*, World Scientific, Singapore, 2006.

[49] F. Lin, Y. Chen, J. An, and X. Zhou, "An evaluation method for network reliability in ad-hoc networks," in *Proceedings of the 4th International Conference on Multimedia Information Networking and Security (MINES '12)*, pp. 628–631, Nanjing, China, November 2012.

[50] B. Vaidya, S.-S. Yeo, D.-Y. Choi, and S. Han, "Robust and secure routing scheme for wireless multihop network," *Personal and Ubiquitous Computing*, vol. 13, no. 7, pp. 457–469, 2009.

[51] L. Reddeppa Reddy and S. V. Raghavan, "SMORT: Scalable multipath on-demand routing for mobile ad hoc networks," *Ad Hoc Networks*, vol. 5, no. 2, pp. 162–188, 2007.

[52] Y. Wang, C. Lin, Q.-L. Li, and Y. Fang, "A queueing analysis for the denial of service (DoS) attacks in computer networks," *Computer Networks*, vol. 51, no. 12, pp. 3564–3573, 2007.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 9787503, 8 pages
https://doi.org/10.1155/2018/9787503

Review Article

Overview on Fault Tolerance Strategies of Composite Service in Service Computing

Junna Zhang, Ao Zhou, Qibo Sun, Shangguang Wang, and Fangchun Yang

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Correspondence should be addressed to Ao Zhou; hellozhouao@gmail.com

Received 11 April 2018; Accepted 23 May 2018; Published 19 June 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Junna Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to build highly reliable composite service via Service Oriented Architecture (SOA) in the Mobile Fog Computing environment, various fault tolerance strategies have been widely studied and got notable achievements. In this paper, we provide a comprehensive overview of key fault tolerance strategies. Firstly, fault tolerance strategies are categorized into static and dynamic fault tolerance according to the phase of their adoption. Secondly, we review various static fault tolerance strategies. Then, dynamic fault tolerance implementation mechanisms are analyzed. Finally, main challenges confronted by fault tolerance for composite service are reviewed.

1. Introduction

With the rapid advance of SOA, there are greater numbers of self-contained, self-describing, loosely coupled, and modular component services in the Internet. To implement sophisticated business applications, one or more services are combined into value-added and coarse-grained service oriented system, that is, composite service. Nowadays, a growing number of enterprises employ composite services to shorten the software development cycle, reduce development costs, and ultimately implement their business processes [1].

However, faults are prone to happen during the execution of composite service. That is because a large proportion of component services are deployed in the best-effort and unreliable Internet, especially in the Mobile Fog Computing environment. Mobile Fog Computing is put forward to enable computing directly at the edge of the network, which can deliver new services for the future of the Internet. However, there are many resource-poor devices in the Mobile Fog Computing environment, for example, routers, switches, and base stations. Composite services are more prone to fault if component services are deployed on resource-poor devices [2]. Therefore, fault tolerant strategy has become a crucial necessity for building reliable composite service. In recent years, many scholars and organizations have engaged in fault tolerant strategies research and put forward various fault tolerant strategies. In this paper, an overview of key fault tolerant strategy for composite service is presented.

We categorize the fault tolerant strategies according to the phase of their adoption. When fault tolerance strategy is employed in the design phase of composite service, it is referred to as a static fault tolerant strategy. When it is adopted during the execution phase, the strategy is referred to as a dynamic fault tolerant strategy [3]. There are various implementation schemes for static and dynamic fault tolerance strategies, so an overview of main literature about them is presented in this paper.

The rest of this paper is organized as follows. The next section presents the category of fault tolerance. Static fault tolerance strategies are analyzed in Section 3. Dynamic fault tolerance strategies are discussed in Section 4. Brief conclusion about the challenge of fault tolerance strategies is given in Section 5. The last section concludes the paper.

2. Category for Fault Tolerance Strategy

To enhance the reliability and trustworthiness of composite service, various fault tolerance strategies have been put forward. The major fault tolerance strategies can be divided into static and dynamic fault tolerance strategy via the phase of their adoption. Static fault tolerance strategy is employed in the design phase of composite service, and it is usually

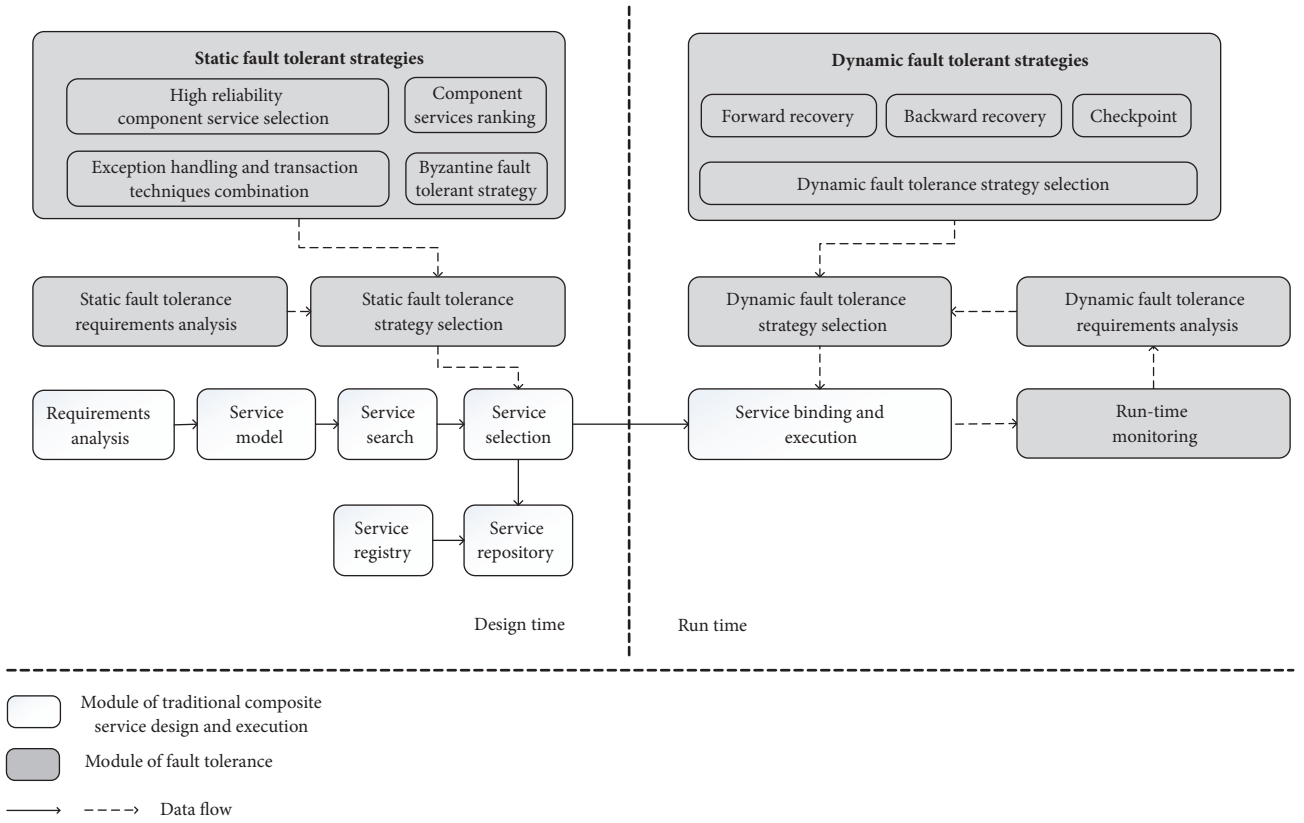


Figure 1: Fault tolerant composite service design and execution. The rectangle module presents the module of traditional composite service design and execution. The grey rectangle module presents the module that implements the fault tolerance function. Lined arrow and dashed arrow represent the data flow between the modules.

to implement the fault tolerance requirements of the user. Moreover, the designer considers the fault that is possible to occur during the execution stage and implements the coping strategy in the design stage. Dynamic fault tolerance strategy is usually adopted when the composite service really fails, and its purpose is to troubleshoot and resume execution of the composite service.

In order to make the category easier to understand, fault tolerance modules are inserted into traditional composite service design and execution modules [4]. All modules are illustrated in Figure 1.

In the design stage of the composite service, the composite service developers need to analyze fault tolerant requirements besides the functional requirements to implement complex tasks of the consumer. According to the results of the static fault tolerant requirements (which are obtained from the static fault tolerance requirements analysis module), the developer can select an appropriate strategy (which is obtained from the static fault tolerance selection module) and employ it in the service selection process. There are various traditional static fault tolerant strategies, for example, the high-certainty, high-trustworthiness, and high-reliability component services selection, exception handling and transaction techniques combination, and component services ranking. Besides, there is a kind of special fault during the execution of composite service, which can be referred to as Byzantine fault. To handle Byzantine fault, Byzantine fault tolerance strategy must be performed at the design time. All aforementioned static fault tolerance strategies will be analyzed in Section 3.

A fault may occur during the run-time of composite service. Therefore, the execution states of composite service should be collected by the run-time monitoring module. When a fault occurs, fault tolerant requirements are firstly analyzed by the dynamic fault tolerance requirements analysis module according to the fault state. Then an appropriate fault tolerant strategy is selected via the dynamic fault tolerance strategy selection module. Forward recovery, backward recovery, and checkpoint are main dynamic fault tolerance strategies. Finally, fault tolerance strategy recoveries the execution of composite service from the fault state. All aforementioned dynamic fault tolerance strategies will be discussed in Section 4.

3. Static Fault Tolerance Strategies

To construct a reliable and trustworthy composite service, static fault tolerant strategies are adopted at the stage of design. The purpose of static fault tolerance strategy is to select reliable and trustworthy component service for composite service. Static fault tolerance strategies are usually carried out during the service selection phase [5]. There are various static fault tolerant strategies, for example, the high-certainty component selection [6], high-trustworthiness

component selection [7], high-reliability component selection [8, 9], fault tolerance based on exception handling and transaction techniques [10], and component services ranking [11].

The above-mentioned strategies can only handle traditional fault of composite service, but they cannot handle Byzantine fault. A Byzantine fault poses a serious threat to the composite service via sending conflicting information to other component services. To mask this type of fault, Byzantine fault tolerance strategy must be adopted [12]. Hence, researchers keep exploring and working on this study.

3.1. Traditional Static Fault Tolerance Strategies. Besides functional requirements, nonfunctional requirements (or QoS constraints, e.g., total execution time should be less than 10 s) should be satisfied in a composite service design. However, component service providers only provide the average QoS values or even incorrect values to improve utilization, which would lead to the violation of QoS constraints. That is to say, there will be a fault. To avoid this situation, component service with high certainty and high reputation should be chosen in selection phase [13, 14].

To select the component services with the highest certainty for composite service, a reliable and efficient approach is put forward in [6]. Firstly, the approach adopts the probability theory and information theory to filter component services with low certainty. Then a reliable fitness function is devised via using 0-1 integer programming. Finally, the component services with the highest certainty are selected based on the fitness function.

According to the collaboration reputation, a service selection approach is proposed in [7] to select the trustworthy component service. The collaboration reputation is constructed on a component service collaboration network that includes two metrics. One metric is invoking reputation, which can be calculated via the recommendation of other component services. The other metric is invoked reputation, which can be calculated according to the interaction frequency among component services. Finally, a trustworthy component service selection algorithm is put forward based on collaboration reputation.

To improve the fault tolerance of the composite service, a novel service selection approach is proposed in [15]. The approach consists of two decision phases. In the first decision phase, the finding of reliable component service is defined as a multiple criteria decision-making problem. And a decision model is constructed to address this problem. In the second decision phase, service selection problem is formulated as an optimization problem based on QoS requirements, and a convex hull approach is presented to solve this optimization problem.

In [10], a fault tolerant framework that is referred to as FACTS is proposed for composite service. To design a fault tolerant mechanism that combines exception handling and transaction techniques, this paper identifies a set of high level exception handling strategies and presents a new taxonomy of transactional component services. Moreover, two modules (a specification module and a verification module) are also designed for assisting service designers in constructing fault handling logic conveniently and correctly.

Component service ranking is another approach for fault tolerance. In [11], FTCloud, a component service ranking framework, is put forward. Firstly, the framework employs two ranking algorithms. The first algorithm adopts invocation structures and frequencies of component service to make significant component ranking. The other ranking algorithm recognizes the significant component services from all composite services by fusing the system structure information and the designer's wisdom of application. After the component service ranking phase, a selection algorithm for optimal fault tolerance strategy is proposed, which can automatically supply optimal fault tolerance strategy for the significant components.

Traditional static fault tolerant strategies are usually employed in the design phase of composite service, so the key research issue of them is not the execution time reduction but the accuracy improvement [16]. Meanwhile, for aforementioned strategies that are only adopted in the design phase, their effectiveness during the execution is another key research issue. To our knowledge, there are few strategies that consider both accuracy and effectiveness during the execution.

3.2. Byzantine Fault Tolerance Strategies. During the execution of composite service, a failed component service may send conflicting information to another component service, which constitutes various threats to the consistency of composite service. This type of fault is known as Byzantine fault [17]. To mask Byzantine fault during the execution phase, the composite service must employ a fault tolerance strategy in the design phase [18]. In recent years, some scholars engage in studying Byzantine fault tolerance strategy.

To tolerate Byzantine faults of composite service, a framework, BFT-WS, is designed and used in [19, 20]. Firstly, BFT-WS adopts the standard technology of composite service (i.e., SOAP) to construct Byzantine fault tolerance service. Employing standard technology can ensure the interoperability of component services. BFT-WS is designed as a pluggable module. Therefore, the implementation of BFT-WS needs minimum change to the composite service. Finally, the key fault tolerance schemes employed in BFT-WS are designed based on the notable Castro and Liskov's Byzantine fault tolerance approach.

A practical algorithm, Perpetual, is proposed in [21]. Perpetual can tolerate Byzantine faults of deterministic n-tier composite service. Interaction between services with different number of replica is allowed in Perpetual. In addition, Perpetual supports not only long-running active threads of computation but also asynchronous invocation and processing. Therefore, Perpetual can improve performance and flexibility over other protocols.

To make the coordination of Web Services Business Activities (WS-BA) more trustworthy, a lightweight Byzantine fault tolerance algorithm is put forward in [22]. Depending on careful study of the threats of the WS-BA coordination services and comprehensive analysis of the state model, the algorithm is lightweight designed. In order to implement

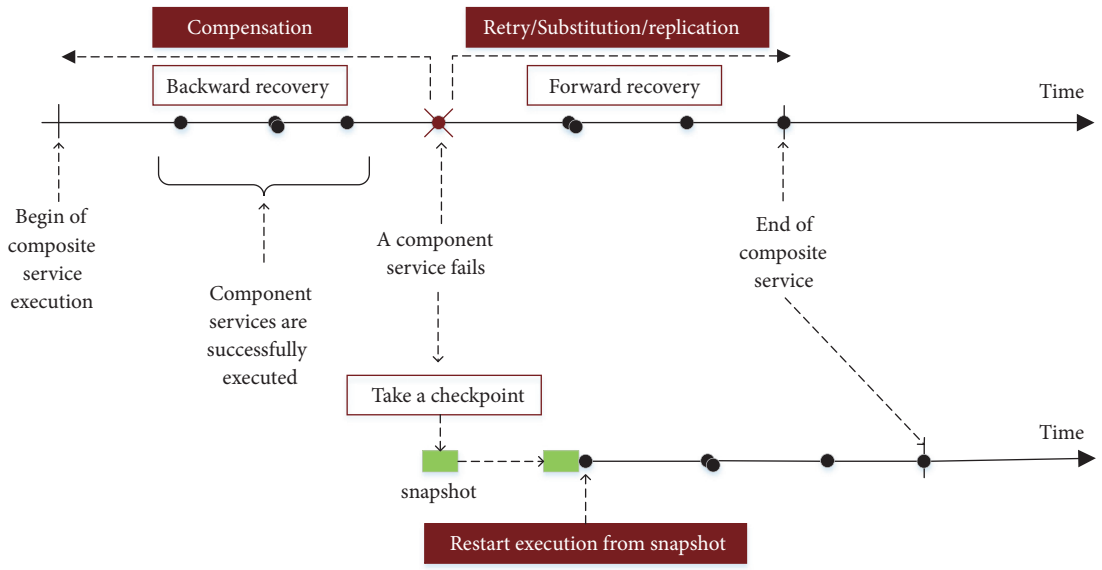


Figure 2: Dynamic fault tolerance strategies. During the execution of composite service, the fault tolerance strategy type can be selected according to the fault state and fault tolerance requirements when a fault occurs.

Byzantine fault tolerance and state machine replication of the WS-BA coordination services, the algorithm uses source ordering rather than total ordering.

To orchestrate delivery of reliable composite services, a hybrid asynchronous Byzantine fault tolerant protocol, GEMINI, is proposed in [23]. Firstly, GEMINI decomposes composite services' abstract workflows from its implementation because it sustains dynamic components provisioning. Then, GEMINI guarantees the reliability of service delivery modules via a lightweight Byzantine fault tolerant protocol. Moreover, GEMINI invokes multiple component services concurrently to realize component service redundancy. Finally, GEMINI employs a single leader Byzantine faults tolerance technology to optimize the current Byzantine fault tolerant protocol.

To handle Byzantine fault, group communication is obligatory among the component service replicas. However, if the traffic between different replicas of component service is very heavy, the response time of a component service may remarkable increase. That is because component services are usually distributed on the Internet. So a key research issue of the Byzantine fault tolerant is reducing the response time of component service. Meanwhile, component service replicas are usually provided by different service providers. Therefore, how to guarantee seamless communication between replicas is another key research issue.

4. Dynamic Fault Tolerance Strategies

A component service may fail during the execution of composite service. The fault must be repaired via dynamic fault tolerance strategies; otherwise, it will lead to the failure of composite service. The current dynamic fault tolerance strategies include forward recovery, backward recovery, and checkpoint, which are illustrated in Figure 2. To ensure the whole composite service in a consistent state even suffering from fault, it is necessary to provide component services with transactional property (all or nothing (every component service of composite service must either be executed successfully or have no effect whatsoever)). Backward recovery and forward recovery are two basic fault tolerance strategies supported by component service's transactional properties. If the faulty component service can be retried [24], replicated [25], or substituted [26], forward recovery is allowed. If the effects produced by the faulty component service need to be compensated [27], backward recovery is allowed [28]. However, users need to wait a long time to get the desired response when forward recovery is adopted, and users are unable to get the desired answer to their queries when backward recovery is adopted [29]. Taking checkpoint is another dynamic fault tolerance strategy. Current execution state and partial results are taken as a snapshot, which is returned to the user when a fault occurs. The checkpointed composite service can be restarted from the latest saved state, and the aggregated transactional attributes are not affected [28]. The recent researches of the dynamic fault tolerance are discussed in the following sections.

Different dynamic fault tolerance strategies need to be adopted for the different faults that occur during the execution of the composite service, and some scholars have specifically studied dynamic fault tolerance strategies selection [7]. Therefore, the main literature about it is presented in Section 4.4.

4.1. Forward Recovery. For forward recovery, the composite service tries to fix the fault without stopping execution. Retry, replication, and substitution can be used for forward recovery [30].

A solution based on forward recovery is proposed in [31] to provide reliable composite service. The solution has

no impact on the autonomy of the component services while exploiting their possible support for fault tolerance. The key issue of this solution is to construct cooperative atomic actions that have a well-defined behavior. Firstly, the notion of Web Service Composition Action (WSCA) is defined according to the concept of coordinated atomic action. Then dependable actions are structured by WSCA, and fault tolerance can be gotten as an emergent property of aggregation of several potentially nondependable services [32].

Fault can be repaired by the substitution. A substitution policy is proposed in [33], which substitutes a subset of component services (includes failed component service) with another equivalent subset. When a fault occurs, all subsets containing the failed component service are identified. Then the subsets that are equivalent to the failed one are determined. Finally, the equivalent subsets are ranked, and the failed subset is substituted by the best equivalent subset.

Replication creates redundant component services (replicas) for composite service. When a request from the user is assigned to all replicas, the technology is called active replication. Otherwise, only one replica acts as the primary one that responds to the request, and the backup replica takes over only after the primary one fails. The technology is called passive replication [34].

WS-Replication, a framework for seamless replication of composite services, is proposed in [35]. To increase the service availability, the framework permits the deployment of a component service in a set of sites. One of the standout features of WS-Replication is that replication is done concerning component service autonomy and only SOAP is used to interact across sites. What is more, WS-Multicast (one of the major components of WS-Replication) can also be used as a self-governed component for reliable multicast in a component service setting [36].

In [37], a distributed replication strategy evaluation and selection framework for fault tolerant composite service is proposed. Based on the proposed framework, various replication strategies are compared by using the theoretical formula and experimental results. Moreover, a strategy selection algorithm based on both objective performance information and subjective requirements of users is proposed.

Each of the aforementioned strategies has its own advantages and disadvantages and is employed for specific fault tolerance scenarios. The composite service developer should first analyze the requirements of the user and the possible fault scenario and then select appropriate strategy [38].

4.2. Backward Recovery. When a fault occurs, backward recovery should be adopted if the effects need be compensated [39].

Some scholars employed exception handling strategies to realize the backward recovery. For example, Liu et al. [10] present a framework named FACTS for fault tolerance of transactional composite service. FACTS combines exception handling and transaction techniques to improve fault tolerance of composite services. Firstly, the framework identifies a set of high level exception handling strategies. Then, a specification module is designed to help service designers to construct correct fault-handling logic. Finally, a module is devised to automatically implement fault-handling logic in WS-BPEL.

An efficient framework for fault tolerance of transactional composite service is proposed in [40]. For recovery from fault, the framework realizes a backward recovery method based on unfolding processes of Coloured Petri-Nets. The framework can be realized in distributed/shared memory system.

According to the transactional properties of component service, a framework, called FaCETa, is proposed in [41]. FaCETa employs service replacement and Coloured Petri-Nets' unrolling processes to tolerate fault. Besides, experimental results show that FaCETa efficiently realizes fault tolerant strategies for the transactional composite service with small overhead.

An approach that dynamically calculates the composite service's reliability to improve the performance of backward recovery is proposed in [42]. Firstly, a model of reliability is presented according to the doubly stochastic model and renewal processes. Then, to help the calculation of complex composite services, a bounded set strategy is briefly presented. Finally, a fault tolerance model is constructed via backward recovery block techniques.

Guillaume et al. [39] focus on checking the correctness of compensation via invariant preservation. Therefore, a correct-by-construction approach, which uses the Event-B algorithm to deal with runtime compensation, is put forward based on refinement and proof. The approach can be used as a foundational module for the compensation of run-time composite service. Meanwhile, a formal model is defined for equivalent, degraded, and upgraded service compensations.

Backward recovery needs to go back to a consistent state to repair the fault correctly. Therefore, a key issue of it is how to save the execution state of the composite service. In addition, how to look for an alternative execution path from the consistent state is another key issue of backward recovery.

4.3. Checkpoint. Checkpoint refers to execution states of composite service gathered by orchestration in a certain time, and the composite service can return to a previous specific state for fault tolerance [43].

Marzouk et al. [44] propose a flexible approach for composite service's execution. The approach synchronizes all flow branches of the composite service. Then a recovery state that permits saving a consistent checkpoint is constructed. When a fault or a QoS violation occurs, the failed process or a subset of running instance may be migrated to another server and restarted according to the checkpoint image.

The traditional "all-or-nothing" is too restrictive for composite service. Checkpoint techniques can relax the atomicity based on the transactional properties of component service. Based on checkpoint and transactional properties, a model that measures the fuzzy atomicity of composite service is presented in [45]. "All-or-nothing" attribute is relaxed into a fuzzy "all-something-or-almost-nothing" attribute.

Based on Coloured Petri-Nets, a checkpoint approach is proposed in [46]. If a fault occurs, the approach relaxes

the all-or-nothing attribute by executing a transactional composite Web service as much as possible and taking a snapshot of faulted state. In other words, the approach returns partial answers to the user as soon as possible. According to the snapshot, the user can resume the composite service without dropping the work previously done.

In [29], the unfolding processes of the Coloured Petri-Nets that control the execution of a transactional composite Web service are checkpointed if a fault occurs. In such way, users can first get partial responses as soon as they are obtained, and the composite service can be restarted from an advanced point of execution.

4.4. Dynamic Fault Tolerance Strategy Selection. Different types of faults may happen during the execution of the composite service. Therefore, different fault tolerance strategies should be employed to recover them [47]. There are some literatures that study how to select the most appropriate fault tolerance strategy [48].

The fault tolerance strategy selection has a significant effect on the QoS of composite service [49]. Therefore, Zheng et al. [50] investigated the problem of selecting an optimal fault tolerance strategy for building reliable composite services. They formulated the user's requirements as local constraints and global constraints and modelled the fault tolerance strategy selection as an optimization problem. A heuristic algorithm is presented to efficiently solve the optimization problem.

In [51], a QoS-aware fault tolerant middleware is proposed to make the dependability of composite service. The middleware includes a user-collaborated QoS model, a set of fault tolerance strategies, and a context-aware algorithm that (dynamically and automatically) determines the optimal fault tolerance strategy for both stateful and stateless composite services.

To maintain the required QoS even in the presence of fault, a novel approach is proposed in [4]. This approach builds on the top of the execution system of composite service and carries out the QoS monitoring. The result of QoS monitoring determines the selection of the fault tolerance strategy in case of fault.

To select appropriate fault tolerance strategy, Shu et al. [52] considered that the reliability of composite services must be analyzed. They proposed a tree-based composition structure model called the Fault-Tolerant Composite Web Service Tree (FCWS-T). Firstly, nodes in FCWS-T are separated into two types, which are control nodes and service nodes. Then, a reliable simulation method is put forward based on FCWA-T, and it can efficiently analyze the reliability of a complex composite service. Finally, an appropriate fault tolerance strategy is selected according to the reliability.

Using priority selector and fault handler, an approach of fault tolerance for service oriented architecture is put forward in [53]. Firstly, the approach selects the first priority level scheme quickly when a fault has been detected. If the fault cannot be handled, the second priority level scheme is selected by a fault handler for average performance. Otherwise, the lowest priority level scheme is employed to handle the fault.

5. Discussion and Open Challenges

Fault tolerance strategy has achieved great development in recent decades and has been successfully applied for solving various faults during the execution of composite service. However, due to the special structure (i.e., based on SOA) and complex and unreliable execution environment of composite service, there are numerous challenges in the research of fault tolerance strategy.

(1) Regarding a compatible development platform for component service, to construct fault tolerant composite service, various fault tolerance strategies should be employed. Most strategies try to choose another component service to replace the faulty one. However, component services are usually developed by different organizations based on different development platform, which leads to some differences between them. These differences have negative impact on the effectiveness of fault tolerance strategy. But few literatures consider this issue now. The difference would not be eliminated unless there is a compatible development platform for component service. Therefore, one of the future researches of fault tolerance strategy is to develop a compatible development platform for component service.

(2) For effectiveness validation of fault tolerance strategy in a real network environment, in recent years, plenty of fault tolerant strategies are proposed for different fault of composite service. However, most of their effectiveness is only validated in a simulation environment. However, real network environment is complex and changeable, and all existing simulation platforms cannot simulate it. Therefore, how to validate the effectiveness of existing fault tolerance strategy in a real network environment needs further study.

6. Conclusion

Building a highly reliable composite service has become a key issue with the prevalence of component services in the Internet. Therefore, many fault tolerance strategies are proposed in recent years. In this paper, fault tolerance strategies are divided into static and dynamic fault tolerance strategies. For implementation of static fault tolerance strategy, there are the high-certainty, high-trustworthiness, and high-reliability component services selection, fault tolerant mechanism of combined exception handling and transaction techniques, and component services ranking. Besides, Byzantine fault tolerant strategy can mask a special kind of fault, that is, Byzantine fault. The overview of the main literature about them is discussed. For implementation of dynamic fault tolerance strategy, there are forward recovery, backward recovery, and checkpoint. The overview of main literature about them is analyzed. Moreover, some challenges in the research of fault tolerance strategy are also provided.

Conflicts of Interest

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

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61602054, 61472047, and 61571066) and Beijing Natural Science Foundation (no. 4174100).

References

[1] Y. Ma, S. Wang, P. C. Hung, C. H. Hsu, Q. Sun, and F. Yang, "A highly accurate prediction algorithm for unknown web service QoS value," *IEEE Transactions on Services Computing*, vol. 9, no. 4, pp. 511–523, 2016.

[2] S. Yi, C. Li, and Q. Li, "A survey of fog computing: concepts, applications and issues," in *Proceedings of the Workshop on Mobile Big Data (Mobidata '15)*, pp. 37–42, ACM, Hangzhou, China, June 2015.

[3] A. A. von Davier, "Service fault tolerance for highly reliable service-oriented systems: an overview," *Science China Information Sciences*, vol. 58, no. 1, pp. 1–12, 2015.

[4] A. Immonen and D. Pakkala, "A survey of methods and approaches for reliable dynamic service compositions," *Service Oriented Computing and Applications*, vol. 8, no. 2, pp. 129–158, 2014.

[5] N. Aljeri, K. Abrougui, M. Almulla, and A. Boukerche, "A reliable quality of service aware fault tolerant gateway discovery protocol for vehicular networks," *Wireless Communications and Mobile Computing*, vol. 15, no. 10, pp. 1485–1495, 2015.

[6] S. Wang, L. Huang, L. Sun, C.-H. Hsu, and F. Yang, "Efficient and reliable service selection for heterogeneous distributed software systems," *Future Generation Computer Systems*, vol. 74, pp. 158–167, 2016.

[7] S. Wang, L. Huang, C.-H. Hsu, and F. Yang, "Collaboration reputation for trustworthy Web service selection in social networks," *Journal of Computer and System Sciences*, vol. 82, no. 1, part B, pp. 130–143, 2016.

[8] C. Esposito, M. Ficco, F. Palmieri, and A. Castiglione, "Smart cloud storage service selection based on fuzzy logic, theory of evidence and game theory," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 65, no. 8, pp. 2348–2362, 2016.

[9] A. Zhou, S. Wang, Z. Zheng, C.-H. Hsu, M. R. Lyu, and F. Yang, "On cloud service reliability enhancement with optimal resource usage," *IEEE Transactions on Cloud Computing*, vol. 4, no. 4, pp. 452–466, 2016.

[10] A. Liu, Q. Li, L. Huang, and M. Xiao, "FACTS: A framework for fault-tolerant composition of transactional web services," *IEEE Transactions on Services Computing*, vol. 3, no. 1, pp. 46–59, 2010.

[11] Z. Zheng, T. C. Zhou, M. R. Lyu, and I. King, "Component ranking for fault-tolerant cloud applications," *IEEE Transactions on Services Computing*, vol. 5, no. 4, pp. 540–550, 2012.

[12] L. Chen and W. Zhou, "Byzantine Fault Tolerance with Window Mechanism for Replicated Services," in *Proceedings of the 2015 Fifth International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC)*, pp. 1255–1258, Qinhuangdao, China, September 2015.

[13] S. Wang, A. Zhou, W. Lei, Z. Yu, C.-H. Hsu, and F. Yang, "Enhanced user context-aware reputation measurement of multimedia service," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 12, no. 4s, 2016.

[14] S. Wang, Z. Zheng, Z. Wu, M. R. Lyu, and F. Yang, "Reputation measurement and malicious feedback rating prevention in web service recommendation systems," *IEEE Transactions on Services Computing*, vol. 8, no. 5, pp. 755–767, 2015.

[15] W. Wang, Z. Huang, and L. Wang, "ISAT: An intelligent Web service selection approach for improving reliability via two-phase decisions," *Information Sciences*, vol. 433-434, pp. 255–273, 2018.

[16] A. Zhou, S. Wang, B. Cheng et al., "Cloud Service Reliability Enhancement via Virtual Machine Placement Optimization," *IEEE Transactions on Services Computing*, vol. 10, no. 6, pp. 902–913, 2017.

[17] L. Lamport, R. Shostak, and M. Pease, "The Byzantine Generals Problem," *ACM Transactions on Programming Languages and Systems (TOPLAS)*, vol. 4, no. 3, pp. 382–401, 1982.

[18] S. Murugan and B. Muthukumar, "Detection and Elimination of Byzantine Faults Using SOAP Handlers in Web Environment," *Modern Applied Science (MAS)*, vol. 9, no. 9, 2015.

[19] W. Zhao, "BFT-WS: A Byzantine Fault Tolerance Framework for Web Services," in *Proceedings of the 2007 11th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOC Workshops)*, pp. 89–96, Annapolis, MD, USA, October 2007.

[20] W. Zhao, "Design and implementation of a Byzantine fault tolerance framework for Web services," *The Journal of Systems and Software*, vol. 82, no. 6, pp. 1004–1015, 2009.

[21] S. L. Pallemulle, H. D. Thorvaldsson, and K. J. Goldman, "Byzantine fault-tolerant Web services for n-tier and service oriented architectures," in *Proceedings of the 28th International Conference on Distributed Computing Systems, ICDCS 2008*, pp. 260–268, chn, July 2008.

[22] H. Chai, H. Zhang, W. Zhao, M.-S. Michael, and L. E. Moser, "Toward trustworthy coordination of web services business activities," *IEEE Transactions on Services Computing*, vol. 6, no. 2, pp. 276–288, 2013.

[23] I. Elgedawy, "GEMINI: A Hybrid Byzantine Fault Tolerant Protocol for Reliable Composite Web Services Orchestrated Delivery," *International Journal of Computer Theory and Engineering*, vol. 8, no. 5, pp. 355–361, 2016.

[24] A. Erradi, P. Maheshwari, and V. Tosic, "Recovery policies for enhancing Web services reliability," in *Proceedings of the ICWS 2006: 2006 IEEE International Conference on Web Services*, pp. 189–196, usa, September 2006.

[25] M. Vargas-Santiago, L. Morales-Rosales, S. Pomares-Hernandez, and K. Drira, "Autonomic Web Services Enhanced by Asynchronous Checkpointing," *IEEE Access*, 2017.

[26] K. Ali, D. Tarun, and D. Mohemmed, "Computation of QoS While Composing Web Services," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 3, 2017.

[27] S. Wang, T. Lei, L. Zhang, C.-H. Hsu, and F. Yang, "Offloading mobile data traffic for QoS-aware service provision in vehicular cyber-physical systems," *Future Generation Computer Systems*, vol. 61, pp. 118–127, 2016.

[28] R. Angarita, M. Rukoz, and Y. Cardinale, "Modeling dynamic recovery strategy for composite web services execution," *World Wide Web*, vol. 19, no. 1, pp. 89–109, 2016.

[29] M. Rukoz, Y. Cardinale, and R. Angarita, "FaCETa*: Checkpointing for transactional composite web service execution based on petri-Nets," in *Proceedings of the 3rd International Conference on Ambient Systems, Networks and Technologies, ANT 2012 and 9th International Conference on Mobile Web*

Information Systems, MobiWIS 2012, pp. 874–879, can, August 2012.

[30] N. Zhang, J. Wang, Y. Ma, K. He, Z. Li, and X. Liu, "Web service discovery based on goal-oriented query expansion," *The Journal of Systems and Software*, vol. 142, pp. 73–91, 2018.

[31] F. Tartanoglu, V. Issarny, A. Romanovsky, and N. Levy, "Coordinated forward error recovery for composite Web services," in *Proceedings of the 22nd International Symposium on Reliable Distributed Systems, SRDS 2003*, pp. 167–176, ita, October 2003.

[32] A. Zhou, S. Wang, C.-H. Hsu, M. H. Kim, and K.-S. Wong, "Virtual machine placement with (m, n)-fault tolerance in cloud data center," *Cluster Computing*, pp. 1–13, 2017.

[33] S. Gupta and P. Bhanodia, "A fault tolerant mechanism for composition of Web services using subset replacement," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, no. 8, pp. 3080–3085, 2013.

[34] M. Vargas-Santiago, S. E. Pomares-Hernandez, L. A. Morales Rosales, and H. Hadj-Kacem, "Survey on Web Services Fault Tolerance Approaches Based on Checkpointing Mechanisms," *Journal of Software*, pp. 1–19, 2017.

[35] J. Salas, F. Perez-Sorrosal, M. Patîo-Martínez, and R. Jiménez-Peris, "WS-replication: a framework for highly available web services," in *Proceedings of the 15th International Conference on World Wide Web*, pp. 357–366, Edinburgh, UK, May 2006.

[36] A. Zhou, Y. Li, and J. Li, "Efficient Request Assignment Algorithm in Mobile Cloud Computing environment," *International journal of Web and Grid Service*, pp. 1–17.

[37] Z. Zheng and M. R. Lyu, "A distributed replication strategy evaluation and selection framework for fault tolerant Web services," in *Proceedings of the IEEE International Conference on Web Services, ICWS 2008*, pp. 145–152, chn, September 2008.

[38] S. Wang, A. Zhou, C.-H. Hsu, X. Xiao, and F. Yang, "Provision of Data-Intensive Services Through Energy-and QoS-Aware Virtual Machine Placement in National Cloud Data Centers," *IEEE Transactions on Emerging Topics in Computing*, vol. 4, no. 2, pp. 290–300, 2016.

[39] G. Babin, Y. Aït-Ameur, and M. Pantel, "Web Service Compensation at Runtime: Formal Modeling and Verification Using the Event-B Refinement and Proof Based Formal Method," *IEEE Transactions on Services Computing*, vol. 10, no. 1, pp. 107–120, 2017.

[40] Y. Cardinale and M. Rukoz, "A framework for reliable execution of Transactional Composite Web Services," in *Proceedings of the International Conference on Management of Emergent Digital EcoSystems, MEDES'11*, pp. 129–136, usa, November 2011.

[41] R. Angarita, Y. Cardinale, and M. Rukoz, "FaCETa: Backward and Forward Recovery for Execution of Transactional Composite WS," in *The Semantic Web: ESWC 2012 Satellite Events*, vol. 7540 of *Lecture Notes in Computer Science*, pp. 343–357, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[42] H. E. Mansour and T. Dillon, "Dependability and rollback recovery for composite web services," *IEEE Transactions on Services Computing*, vol. 4, no. 4, pp. 328–339, 2011.

[43] LY. Chiu, S. Fan, Y. Liu, and M. Mei, "Providing a fault tolerant system in a loosely-coupled cluster environment using application checkpoints and logs," in *and Mei M. Providing a fault tolerant system in a loosely-coupled cluster environment using application checkpoints and logs*, p. B2, United States Patent: US, 2015.

[44] S. Marzouk, A. J. Maâlej, and M. Jmaiel, "Aspect-Oriented Checkpointing Approach of Composed Web Services," in *Proceedings of the International Conference on Web Engineering (ICWE'10)*, vol. 6385, pp. 301–312, 2010.

[45] Y. Cardinale, J. El Haddad, M. Manouvrier, and M. Rukoz, "Measuring Fuzzy Atomicity for Composite Service Execution," in *Proceedings of the 2016 2nd International Conference on Open and Big Data (OBD)*, pp. 62–71, Vienna, Austria, August 2016.

[46] Y. Cardinale, M. Rukoz, and R. Angarita, "Modeling Snapshot of Composite WS Execution by Colored Petri Nets," in *Resource Discovery*, vol. 8194 of *Lecture Notes in Computer Science*, pp. 23–44, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[47] R. Gupta, R. Kamal, and U. Suman, "A QoS-supported approach using fault detection and tolerance for achieving reliability in dynamic orchestration of web services," *International Journal of Information Technology*, vol. 10, no. 1, pp. 71–81, 2018.

[48] Y. Liu, Y. Fan, K. Huang, and W. Tan, "Failure analysis and tolerance strategies in web service ecosystems," *Concurrency Computation*, vol. 27, no. 5, pp. 1355–1374, 2015.

[49] J. Wang, P. Gao, Y. Ma, K. He, and P. C. K. Hung, "A Web Service Discovery Approach Based on Common Topic Groups Extraction," *IEEE Access*, vol. 5, pp. 10193–10208, 2017.

[50] Z. Zheng and M. R. Lyu, "Selecting an optimal fault tolerance strategy for reliable service-oriented systems with local and global constraints," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 64, no. 1, pp. 219–232, 2015.

[51] Z. Zheng and MR. Lyu, "A QoS-aware fault tolerant middleware for dependable service composition," in *Proceedings of the In Proceedings of the IEEE/IFIP International Conference on Dependable Systems Networks (DSN'09)*, pp. 239–248, 2009.

[52] Y. Shu, D. Zuo, H. Liu, Q. Z. Sheng, W. E. Zhang, and J. Yang, "A Tree-Based Reliability Analysis for Fault-Tolerant Web Services Composition," in *Service Oriented Computing and Applications*, vol. 10601 of *Lecture Notes in Computer Science*, pp. 481–489, Springer International Publishing, Cham, 2017.

[53] G. Prasad Bhandari and . Ratneshwer, "Fault Repairing Strategy Selector for ServiceOriented Architecture," *International Journal of Modern Education and Computer Science*, vol. 9, no. 6, pp. 32–39, 2017.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 8562904, 11 pages
https://doi.org/10.1155/2018/8562904

Research Article

A Security Scheme of 5G Ultradense Network Based on the Implicit Certificate

Zhonglin Chen,[1] **Shanzhi Chen**,[1,2] **Hui Xu**,[2] **and Bo Hu**[1]

[1]*State Key Lab of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2]*State Key Lab of Wireless Mobile Communication, China Academy of Telecommunications Technology, Beijing 100081, China*

Correspondence should be addressed to Zhonglin Chen; chenzl@263.net

Received 2 March 2018; Accepted 19 April 2018; Published 23 May 2018

Academic Editor: Fuhong Lin

Copyright © 2018 Zhonglin 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.

The ultradense network (UDN) is one of the most promising technologies in the fifth generation (5G) to address the network system capacity issue. It can enhance spatial reuse through the flexible, intensive deployment of small base stations. A universal 5G UDN architecture is necessary to realize the autonomous and dynamic deployment of small base stations. However, the security of the 5G UDN is still in its infancy, and the data communication security among the network entities is facing new challenges. In this paper, we proposed a new security based on implicit certificate (IC) scheme; the scheme solves the security problem among the access points (APs) in a dynamic APs group (APG) and between the AP and user equipment (UE). We present each phase regarding how two network entities obtain the Elliptic Curve Qu-Vanstone (ECQV) implicit certificate scheme, verify each other's identity, and share keys in an UDN. Finally, we extensively analyze our lightweight security communication model in terms of security and performance. The simulation on network bandwidth evaluation is also conducted to prove the efficiency of the solution.

1. Introduction

In the 5G, data traffic will experience explosive growth in the years to come. The use of wireless physical layer technologies (e.g., coding technology, modulation technology, and multiple access technology) can only increase spectrum efficiency by about 10 times and the wider bandwidth can only improve the transmission efficiency by dozens of times. This is far from meeting the 5G demand. However, through the deployment of dense base stations, the spectrum efficiency caused by reducing the cell coverage radius can be increased by more than 2700 times [1]. Obviously, the application of dense small base stations with the narrow coverage in the heterogeneous network can remarkably improve the system capacity. In order to enhance up the system capacity of regional hotspot hundreds of times, the small bases network deployment needs to be more flexible and the frequency reuse needs to be more efficient. Therefore, the ultradense network (UDN) is proposed and has attracted wide attention [2].

The UDN is considered to be one of the most effective solutions to improve wireless system capacity. It decreases the distance between the user equipment (UE) and the network entities and greatly improves the spectrum efficiency. Meanwhile, the UDN has been identified as a constituent of future 5G core technologies by the IMT-2020 expert group [3]. With various small base stations acting as access points (APs), the intersite distance (ISD) decreases as the network entities' density increases. The AP of 5G is different from the traditional macro station. Traditional macro stations are regularly deployed by operators, while AP deployment may be irregular or even deployed by users. Pseudo or malicious AP will threaten 5G system security. What is more, the APs are not just an air network link; they will cooperate with each other to serve user in UDN. In the air transmission of UDN, the unprotected data among the APs is easy to intercept. Therefore, the mutual authentication and the secure data communication among the dense APs, including the keys for the sessions, will face new challenges.

As the wireless access network of the 5G, the UDN adopts a different deployment plan that focuses on the new requirement of "network follows user" and supports higher date transmission rates and multiple services. This must fully

Table 1: Comparison of the traditional certificate and implicit certificate.

Security level	Public key length (bits)		Certificate length (bits)			ECQV/RSA
	ECC	RSA	ECQV	ECDSA	RSA	
80	192	1024	193	577	2048	9.42%
112	224	2048	225	673	4096	5.49%
128	256	3072	257	769	6144	4.18%
192	384	7680	385	1153	15360	2.51%
256	521	15360	522	1564	30720	1.70%

support the organization and access security of dense APs in a heterogeneous environment and also support the seamless connectivity of the user-to-AP, AP-to-AP, and machine-to-machine communications. Therefore, the UDN faces more extensive and complex security threats than traditional wireless systems. However, the security research of the 5G UDN is still in an initial stage, especially the data communication security among the network entities.

In this paper, we propose a new security scheme based on implicit certificate (IC) to solve the security issues among the dense deployment access points (APs) in a dynamic APs group (APG) and between the AP and user equipment (UE). As a new variant of the public key certificate, the novel IC is more efficient in computing and bandwidth allocation, and it requires no peer information before a secure data communication session [4]. The IC has been widely applied to the efficient authentication of resource-constrained Internet of Things (IoT) systems in the literature [5, 6]. Meanwhile, based on the IC, [7] proposed an effective public key infrastructure for the Vehicle-to-Grid Network. After in-depth research, we believe that the principle based on the IC is suitable for providing a security solution for the UDN.

A new lightweight security scheme for secure data communications is presented in this paper. We provide the specific implementation solutions for the security application scenes in the UDN. Meanwhile, the security scheme is analyzed, and the simulation of the network bandwidth evaluation is conducted to prove the efficiency of the solution. Specifically, the scheme focuses on solving the following three subissues:

(i) How to generate the IC and private key

(ii) How to implement the mutual authentication based on the IC among the network entities

(iii) How to implement the lightweight secure communication with a shared key based on the IC

The main contributions of our proposed scheme are summarized as follows:

(i) In our solution, the reconstructed private key that would be instantaneously generated based on the IC could solve the key security issues in actual operations.

(ii) We propose an innovative scheme to solve the security issues of data communications by using shared key encryption based on the IC.

(iii) Our innovative authentication and key agreement method based on the IC is lightweight, efficient, and less resource-consuming.

The rest of the paper is organized as follows. In Section 2, the security challenges in the 5G UDN architecture are analyzed. The implicit certificate and relevant background knowledge are presented in Section 3. The design of the security solution based on the IC and the implementation processes are described in Section 4. Then, the security analysis and performance evaluation are presented in Section 5. The final conclusions are drawn in Section 6.

2. Implicit Certificate and Related Work

Compared with the traditional digital certificate based on the public key infrastructure (PKI) [8, 9], the implicit certificate (IC) [10] has significant advantages.

The traditional digital certificate (the explicit certificate) is a fixed structure that binds the public key (expressed as P) with the identity (expressed as I) that has an attached signature (expressed as Sig) that can be expressed as a triple (I, P, and Sig) [11, 12]. Different from the traditional certificate, the IC is composed of an identity element (still expressed as I) and reconfigurable key data (also expressed as P). P can reconstruct the public key of the identity entity together with the public key of the certificate authority (CA) [13]. Then, the IC could be expressed as a two-tuple (identity element and reconstructed key data) such as (I, P). Traditional authentication uses the RSA (Rivest-Shamir-Adleman algorithm) [14], the ECDSA (Elliptic Curve Digital Signature Algorithm), and other solutions to conduct the signature process, while the typical implicit authentication adopts the ECQV (Elliptic Curve Qu-Vanstone) solution [15].

(A) Smaller Size and Less Bandwidth Occupation. The IC can reduce the bandwidth occupation in the transmission process. Therefore, it is quite suitable for mobile communications, the Internet of Things (IoT), and other resource-constrained environments. Table 1 shows that the traditional digital certificate (such as the RSA) requires more bandwidth than the IC that uses the lightweight Elliptic Curves Cryptography (ECC) cryptosystem [16, 17]. For example, when the security level in a practical application is 112 bits, the IC size is merely 225 bits, which is 43% of the ECDSA certificate (673 bits) and 5.5% of the RSA certificate (4096 bits).

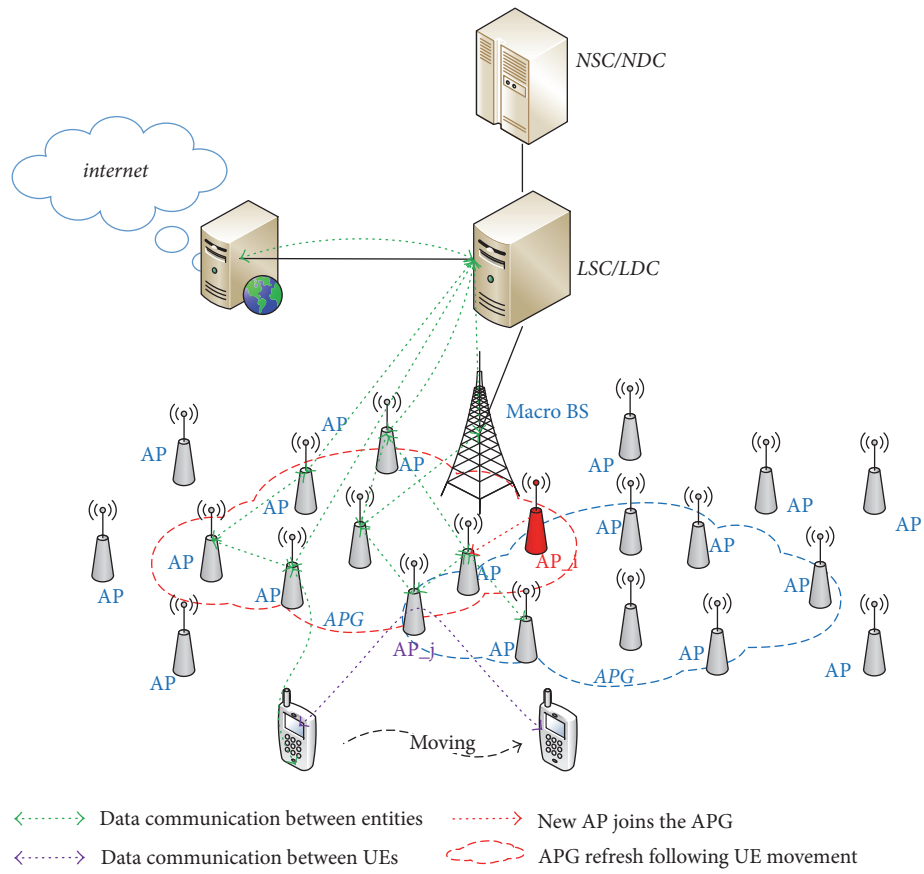


Figure 1: Typical UDN/UUDN architecture.

(B) Higher Speed and Less Resource Consumption. The IC takes the reconstructed public key as a substitute for the signature authentication process. It requires less computing resources compared to a traditional certificate. In addition, some handlers can be integrated into a parallel process with subsequent communication protocols. This may further reduce the computing time and improve efficiency.

In the implicit certificate, there is no signature process instead of the reconstructed public key. The computing work of the reconstructed public key is very small. Then, the implicit certificate is faster than the traditional certificate, consumes fewer resources, and also has better security [18]. For example, when a user's cell phone is stolen, then his private key will also be lost. In the traditional certificate, users need to apply for his certificate revocation to CA. The traditional certificate revocation is to publish the certificate revocation list (CRL) periodically. In large-scale network environment, the CRL is usually large. Because of the periodicity of CRL, the user certificate revocation will be delayed certainly. Thus, the user data security will be threatened. However, the implicit certificate is different. The user's private key corresponds to a short-term lightweight implicit certificate [19]. It does not need to be revoked. And it can be issued quickly and temporarily. In this way, the user's data communication will be more secure. Therefore, the implicit certificate shows stronger advantages than the traditional certificate. For example, ECQV based on implicit certificate has been successfully applied to the field of ZigBee wireless communication. It can be predicted that the implicit certificate can be more widely applied to mutual authentication and secure communication among the network entities in 5G system.

3. UDN Architecture and Security Challenges

3.1. Ultradense Network Architecture. The typical application scenarios of the UDN include office districts, intensive residential areas, high-density blocks, campuses, large gatherings, stadiums, subways, and apartments [20]. The above scenarios require the network to be deployed with adequate flexibility, efficiency, intelligence, and integration abilities. According to practical application demands, different UDN architectures are designed using various organizations of different base stations.

One type of UDN is based on the static virtual cell, which is composed of multiple access points in the area to form a "large" static cell and can provide users a similar coverage to that of the macro base station service experience with a unified identity and common services [21].

Another type of UDN is the user-centric UDN (UUDN) [22, 23], which has a local control center coordinated with the user, and the virtual adjoined cell is defined based on the unit of a single user. A typical UDN/UUDN architecture is shown in Figure 1. In the UUDN, the system organizes

a dynamic APs group (APG) that depends on each UE's situation. It provides unaware and seamless service to the user through dynamic refreshing of the APG as an invisible network coverage accompanying user movement.

3.2. Security Challenges in UDN. The 5G network confronts more extensive and complex security threats compared to current 3G and 4G networks. It includes the traditional security threats in the mobility of multiple UEs and the openness of the wireless channel. Moreover, it also includes new security threats from the enhanced functionality in multiple use patterns, the integration between diversified heterogeneous wireless networks, the open network infrastructure based on the IP framework, and the enriched business bearer with different trust-ratings [24].

The security problems of data communication in the UDN could be summarized as follows.

(C) Access Authentication Security for UE to UDN. To ensure access security, network access authentication is required for the UE to connect with the 5G network. Different from that of the 3G and 4G networks with traditional macro base station coverage, the security threat of UDN cannot be fully avoided by solely depending on traditional authentication and key agreements (AKAs) [25, 26]. For example, the network entities of the UDN in the flexible deployment environment (such as user self-deployed AP or an uncontrolled deployment environment) can be hijacked. Therefore, the security of UE authentication shall be strengthened.

The UE delivers an initial access request. Then, the AP receives the request and transmits it to the local network system. In accordance with the request's context, the local network system requests that the core network system provide the corresponding network layer authentication vector and response. Then, on the basis of the received network layer security parameter, the local network system initiates the network layer's mutual authentication process with the UE (similar to the 4G EPS-AKA process). When the network layer mutual authentication of the UE is finished, a static virtual cell or an APG is allocated to the UE by the UDN. At this time, the local network system generates network access authentication vectors to the UDN control layer based on the request parameter submitted from the UE. It conducts the access layer (a specific virtual cell or APG) mutual authentication process on the UE. When the mutual authentication processes toward both the network layer and the access layer are finished by the UE, the security access is accomplished.

(D) Communication Security among APs/APG. The UDN is composed of densely deployed APs. The APs are connected and organically organized depending on different technical framework demands, such as a unified static virtual cell or an APG. Regardless of the AP organization, the UDN must realize the access service of the UE while maintaining a high-quality user experience. Therefore, the influential factors should be eliminated in the UDN, such as the cochannel interference, shared spectrum interference, interference between multiple coverage layers and frequent network handover caused by the density increase, or the distance decrease between base stations [27].

To protect the APs from various security threats caused by other APs (e.g., illegal APs or malicious APs) and build a secure UDN environment, a solution for secure data communications between APs is necessary and very important. Furthermore, because of the limited capability and small coverage of APs different from the traditional macro base station in 3G and 4G networks, the security of data communication faces new challenges.

(E) Communication Security between UE and AP/APG. With the intelligent development of the UE, network data transportation is getting flatter. Several new trends have emerged in the 5G network architecture, including the localized flat, heterogeneous coordination of macro and small base stations, and submerging business functions. APs and APG are more than a network access. Depending on the difference in the APs and APG functional requirement, they can realize data transportation, data control, or both. The APs or APG becomes the key network entity when the UE accesses the 5G system and Internet. If the relevant registration data suffers security attacks, the security of user traffic also encounters risks. Therefore, the data communication security between users and APs (or APGs) is another security challenge for the 5G UDN.

Based on this analysis, the security requirements of data communication in the UDN include the following:

(i) Each network entity should be mutually authenticated, and the bilateral entities should use their respective private keys. The security mechanism should be applied to ensure that both sides can receive relevant information.

(ii) Each communication entity should be able to obtain the shared keys in data communications. The different communication sessions use different shared keys.

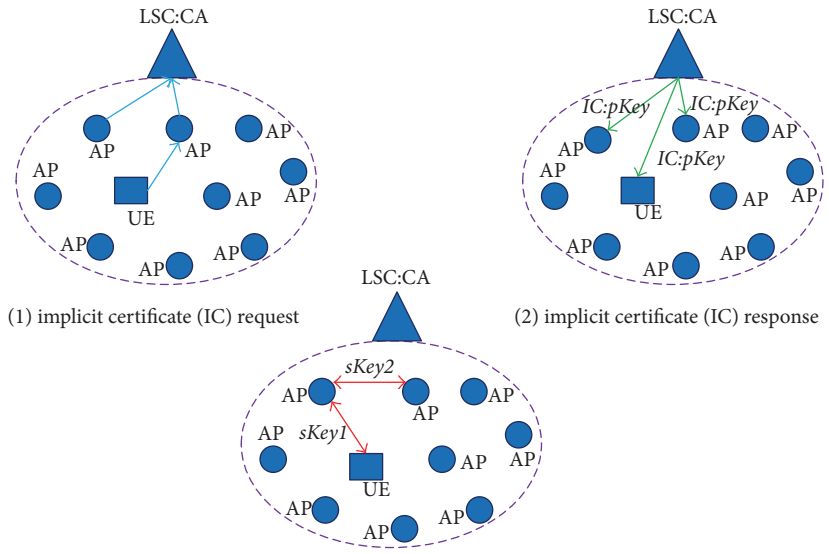
(iii) The security mechanisms for data encrypted based on shared keys should support the dynamic joining or leaving of communication entities.

(iv) All entities should receive unified management from the network operator. The generation of shared keys between communication entities should be in accordance with the relevant instructions.

(v) The security mechanism should support multiple logical channels between the same sources or destinations and avoid the duplication of the keystream.

(vi) The security mechanism should be efficient to ensure quick responses and adapt to the entity's performance and network bandwidth in different communication processes.

From the above requirements, a new certificate and key agreement mechanism are required to establish a secure connection for each pair of entities. Therefore, we propose a new security scheme based on the IC to implement the lightweight data communication between various entities.



(1) APs/UE: its own private key (pKey) can be reconstructed
(2) the shared key (sKey, between AP and AP or between AP and UE) is instant by the algorithm computing

Figure 2: Communication between entities in a UDN based on the IC.

4. UDN Security Solution

The data communication among network entities can be described as follows. It is based on the digital certificate and entity-to-entity security data communication model with the participation of the local service center. The entity can be AP or UE. The local service center (LSC) could run within CA functions.

4.1. Security Model and Notations. Data communication among the network entities is temporary and random, such as when an AP dynamically joins or leaves an APG and when the UE temporarily accesses the APG or other network entities. Therefore, using the IC in temporary key generation is a more convenient and efficient solution to achieve a secure data communication session than the prefixed key distribution. Based on the reconstructed public key and private key, trusted authentication management and shared key generation can be implemented through the implicit certificate from the CA. Then, secure data communication sessions can be implemented by using the shared key computed by the network entities' participants.

Under a CA domain, there are three phases among communication entities. In Phase 1, the CA issues an IC to the requesting entity, which is called the phase of IC generation. In Phase 2, entities conduct mutual certification based on the IC, which is called the phase of mutual identity authentication. In Phase 3, entities exchange data based on shared keys, which is called the phase of shared key generation and data communication. The security data communication model based on the IC is shown in Figure 2.

We assume that the basic configuration has been uniformly predeployed at the initialization phase, including the elliptic curve (EC) parameters, the authentication key K, the public key Q_{CA} of the CA, and the unique user identity label I. The CA can verify the identity and validity of the network entities in order to decide whether they belong to its CA domain. The IC is an ECQV certificate in our solution. The network entities can directly transmit data or forward them through other entities (via single hop or several hops). Any entity can destroy the public key or identity I (or put it on the blacklist) according to the control demand of the LSC.

The notations used in this paper are defined in "Notations." The EC parameters are denoted using q, a, b, G, and n. q is a prime defined on the finite field Fq. a and b are coefficients of the EC curve: $y^2 = x^3 + ax + b$, where $4a^3 + 27b^2 \neq 0$. Another prime G is the base point generator of the EC with order n.

4.2. Security Algorithm Solution. To establish a secure data communication session among the network entities, the security solution based on the IC can be implemented in four phases.

Phase 1 (implicit certificate generation). Before establishing the secure data communication, the entities should launch an IC request to the CA. For example, a new AP (denoted as Ent_U) attempts to join the APG, where another entity (denoted as Ent_V) is registered. The entity Ent_U must communicate with the entity Ent_V and exchange essential information. Then, Ent_U sends an IC request message.

The entity Ent_U with a unique identity ID_U generates a random number r_U and computes $R_U = r_U * G$. Concurrently, to avoid a replay attack, Ent_U produces a cryptographic random number N_U and computes HMAC$[K, R_U \,||\, N_U \,||\, ID_U]$. Then, Ent_$U$ sends R_U, N_U, and ID_U as well as the value of HMAC to the CA. The HMAC is a keyed-hash message authentication code algorithm in cryptography.

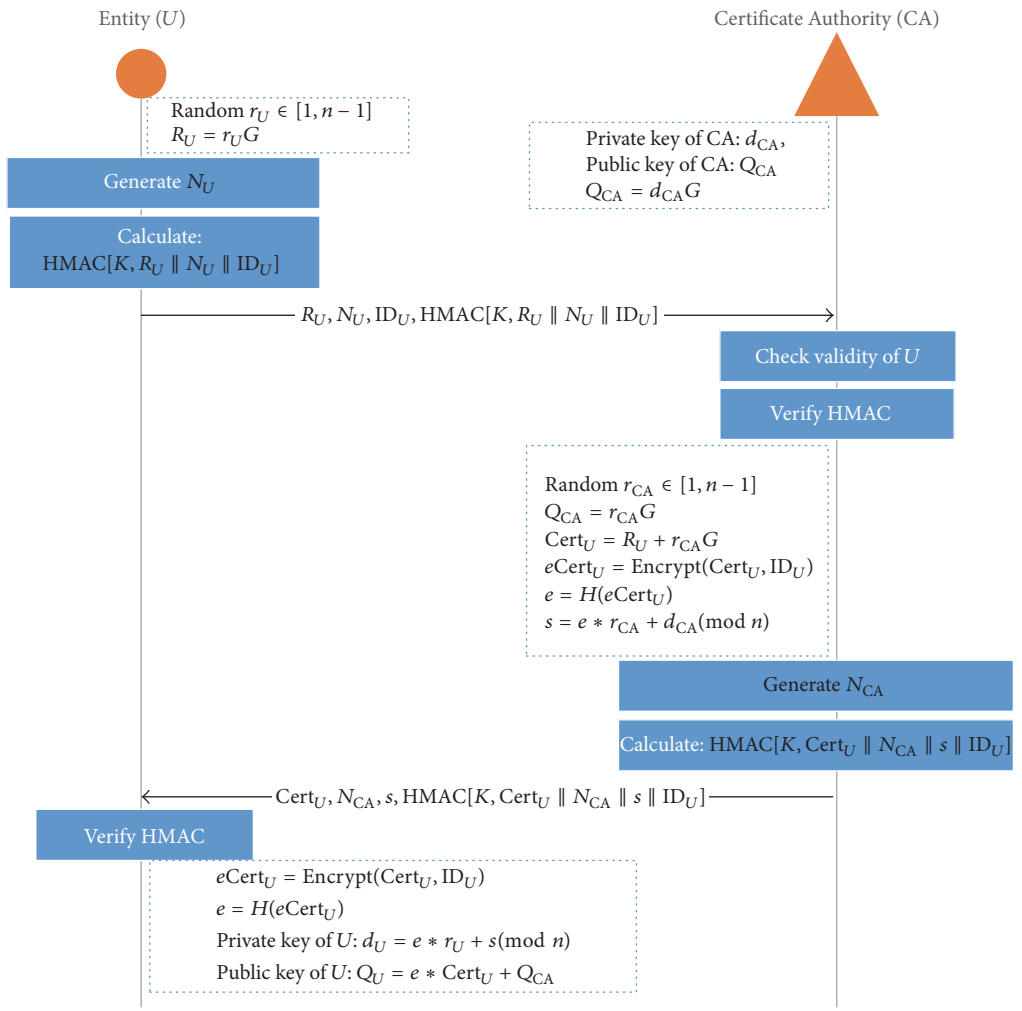


Figure 3: The generation process of the implicit certificate.

After the request is received, the CA (private key is d_{CA}, public key is Q_{CA}, and $Q_{CA} = d_{CA} * G$) verifies the identity ID_U and corresponding HMAC of Ent_U. If the validation is confirmed, a random number $r_{CA} \in [1, n-1]$ will be generated. The CA begins to compute the following:

(i) The reconstructed data of the public key: $\mathrm{Cert}_U = R_U + r_{CA} * G$.

(ii) The encrypted certificate with the entity's identity: $e\mathrm{Cert}_U = \mathrm{Encrypt}(\mathrm{Cert}_U, \mathrm{ID}_U)$, where ID_U is the entity's identity and Encrypt is an encoding function for the identity information protection.

(iii) The component data of the private key: $s = e * r_{CA} + d_{CA} (\mathrm{mod}\ n)$, where $e = H(e\mathrm{Cert}_U)$. H is a Secure Hash Algorithm (SHA) such as SHA-1.

(iv) Similarly, a sequence code N_{CA} is generated by the CA, and then the CA sends back to the requester Ent_U with Cert_U, N_{CA}, s, and HMAC$[K, \mathrm{Cert}_U \| N_{CA} \| s \| \mathrm{ID}_U]$.

Ent_U then verifies the message received from the CA. If the verification is confirmed, Ent_U computes the following keys using the reconstruction data:

$$\text{Ent\_}U \text{ private key (pKey): } d_U = e * r_U + s (\mathrm{mod}\ n)$$

$$\text{Ent\_}U \text{ public key (PKey): } Q_U = e * \mathrm{Cert}_U + Q_{CA}$$

At this point, entity Ent_U has its own public key and private key pair (d_U, Q_U) securely through the IC generation process. Similarly, other entities can apply for their respective ICs and the pairwise key, which is shown in Figure 3.

Phase 2 (mutual authentication between Ent_U and Ent_V). Similarly, the private key (pKey) $d_V = e * r_V + s (\mathrm{mod}\, n)$ and public key (PKey) $Q_V = e * \mathrm{Cert}_V + Q_{CA}$ of entity Ent_V can be easily obtained. Since the reconstructed data are publicly transmitted over the network in the UDN, other entities can be easily obtained.

Therefore, as long as the entity IC and its identity ID_U are known, it is easy to compute the entity's public key. Of course, the CA computes the IC and therefore obviously owns

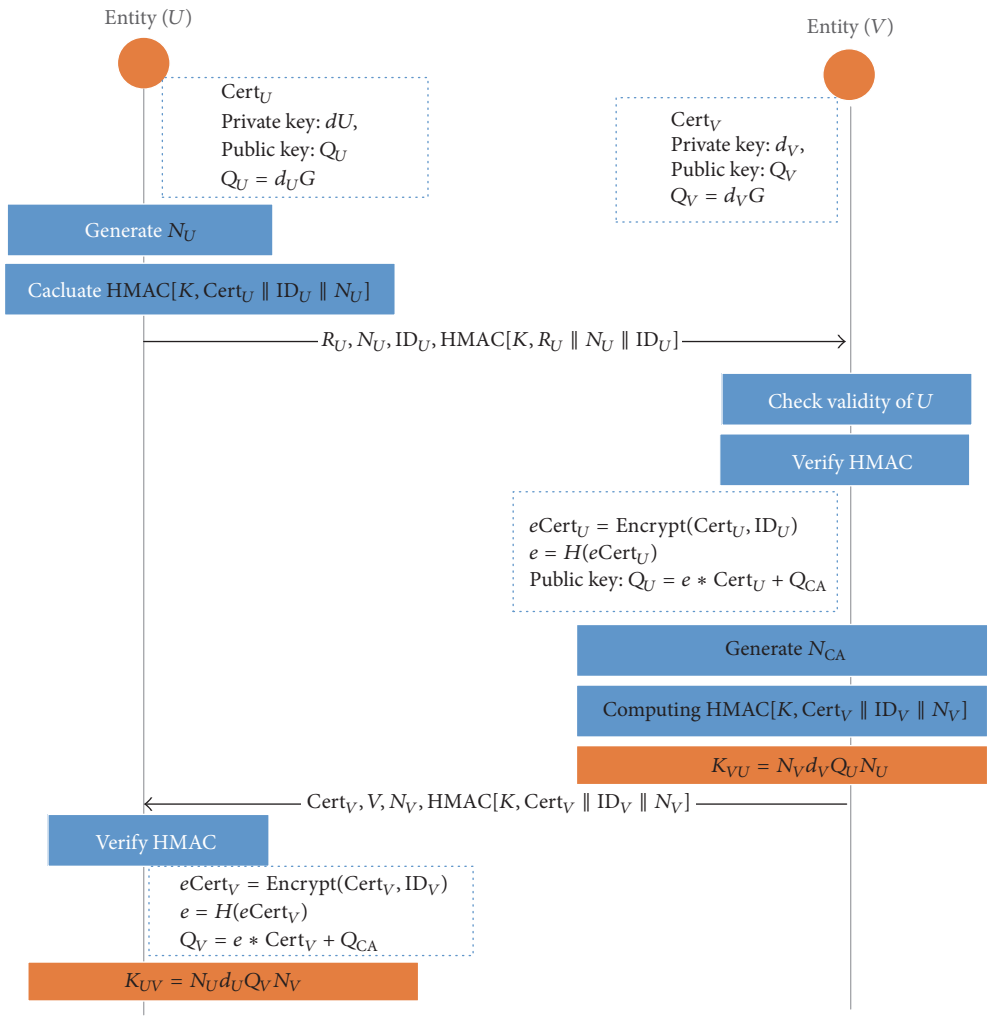


Figure 4: The AKA-IC process for the shared key.

the public keys of all network entities. Any network entity can obtain the other entity's PKey from the CA, but it cannot obtain the pKey owned and computed by others.

The PKey of the entity Ent_U generated in Phase 1 can be verified by the CA. In other words, the identity of entity Ent_U can be verified by the CA using the following formulas:

$$\begin{aligned} Q_U &= d_U * G = \left(e * r_U + s \,(\mathrm{mod}\ n)\right)(\mathrm{mod}\, n)) * G \\ &= \left(e * r_U + e * r_{\mathrm{CA}} + d_{\mathrm{CA}}\,(\mathrm{mod}\, n)\right) * G \\ &= e * \left(r_U + r_{\mathrm{CA}}\right) * G + d_{\mathrm{CA}} * G \\ &= e * \left(R_U + r_{\mathrm{CA}} * G\right) + Q_{\mathrm{CA}} = e * \mathrm{Cert}_U + Q_{\mathrm{CA}} \end{aligned} \tag{1}$$

The verification of Ent_V can also be similarly conducted.

In the UDN, a mutual challenge-response among the APs can be processed using the verification formula method.

Phase 3 (shared key generation between Ent_U and Ent_V). After the identities are confirmed, the entities can agree with the shared key for the communication session to guarantee the confidentiality of data transmission. The sender has to encrypt the data before transmission, while the receiver has to decrypt the data. Accordingly, both of the communication partners must have the same key, namely, the "shared" key (sKey), in this paper. However, since any data with the shared key can be intercepted and have high risk, it is impossible to transmit the key as plaintext in the network. Furthermore, each communication session is temporary and uncertain. The dynamic sessions require the key to be continuously refreshed and updated. It is difficult to preload different encryption keys for each communication session in the actual operator.

Fortunately, we discovered a lightweight scheme based on the IC to solve the issues mentioned above. The shared keys known only by both of the communication partners can be instantly generated through the authentication and key agreement protocol based on the IC (AKA-IC). Moreover, the shared key is locally generated and does not need to be transferred in the network. The new generation mechanism is shown in Figure 4.

From Phases 1 and 2, we know that the parameters of Cert_U, ID_U, N_U, and $\mathrm{HMAC}[K, \mathrm{Cert}_U \,||\, N_U \,||\, \mathrm{ID}_U]$ can be generated and sent to Ent_$V(d_V, Q_V)$ from Ent_$U(d_U, Q_U)$.

After entity Ent_V receives the message from Ent_U, it verifies the identity ID_U and HMAC. First, the public key Q_U of Ent_U can be computed by Ent_V. Then, Ent_V locally computes out the "shared" key sKey = K_{VU} using its private key pKey d_V:

$$\text{sKey} = K_{VU} = N_V d_V Q_U N_U \tag{2}$$

Similarly, the partner Ent_U locally computes the "shared" key sKey = K_{UV} at the same time:

$$\text{sKey} = K_{UV} = N_U d_U Q_V N_V \tag{3}$$

The equation can be derived as follows:

$$\begin{aligned} \text{sKey} = K_{UV} &= N_U d_U Q_V N_V = N_V d_U Q_V N_U \\ &= N_V d_U \left(d_V G\right) N_U = N_V d_V \left(d_U G\right) N_U \\ &= N_V d_V Q_U N_U = K_{VU} \end{aligned} \tag{4}$$

Proof is finished.

The above equation of $K_{VU} = K_{UV}$ = sKey shows that the keys temporarily generated by two entities separately are the same, and they can realize secure data communications using the "shared" key sKey.

Phase 4 (secure communication between Ent_U and Ent_V). When Ent_U and Ent_V have their own pairwise key, the two entities can generate the shared key for their communication sessions. Ent_U encrypts the data that need to be protected by the shared key and sends them to Ent_V. After the encrypted data are received, Ent_V securely decrypts them. Then, the two entities enter into a secure interaction phase until the session ends.

5. Security Analysis and Performance Evaluation

Focusing on the sensing characteristics of randomly deployed MSNs, we analyzed the coverage redundancy problem for the MSNs, where the sensing ranges satisfy the normal distribution.

5.1. Security Analysis

(A) Security of Key Generation. The core of asymmetric cryptography security is the public/private key pair, especially the user's private key. In our solution, the CA generates the user's private key data that can reconstruct the IC using its trusted private key. Then, the reconstructed key data can be locally recomputed. Then, the actual user's private key is generated. The user's private key is locally generated and is not plaintext transmitted in the network. Thus, the security of the user's private key generation is ensured.

(B) Data Confidentiality in Transmission. In our solution, when network entities need to transmit data, both of the communication entities use their private/public key pairs to generate a shared key at their respective locations. The sender encrypts the data using the shared key and sends them to the opposite side. The receiver uses the agreed algorithm to generate the same key to decrypt the data. Thus, the confidentiality of data transmission between the communication entities is guaranteed.

(C) Antireplay Attack. In the process of the shared key generation, the antireplay attack factor NUNV (or timestamp) is added during each computation. If the current interactive data are intercepted and returned to the receiver, the receiver will identify and refuse to receive them. Each communication session has a different encryption key. Moreover, in order to ensure the freshness of shared keys, the secret number increases in the process. It can effectively reduce the shared key's break probability and ensure that the shared key cannot be temporarily reused in the transmission.

(D) Mutual Authentication. In our algorithm, both sides of the communication network's entities have to pass the authentication before they interact with each other. Before the sender delivers the data (such as random numbers and identities), the data must be signed with a digital signature using the sender's private key. When the receiver obtains the signed data, it will use the sender's public key to verify the data. Furthermore, the sender's public key is computed based on the reconstructed public key data. If the validation is correct, then the sender's identity is legal. Similarly, when the receiver sends a reply message, the reply vector data including the identity will also be signed. The opposite side conducts the same legal validation to the vector data. If both sides pass the opposite verification, mutual authentication is complete.

(E) Nonrepudiation. In the algorithm process, both of the communication entities sign the messages using the sender's private key. The source of the data can be identified through the signature, since only the owner of the private key can generate the signature. The receiver simply uses the sender's public key to verify the source of the message. Since the sender's private key is only known by the sender himself/herself, it can effectively prevent the middleman attack and ensure that the sender cannot deny the delivered messages.

(F) Anti-Denial-of-Service Attack. In our scheme, the CA verifies the identity based on the inspection mechanism. According to the registration information in the database, the CA starts with identity check, including blacklists. The CA will directly reject the unregistered or blacklisted user's application for implicit certificates. Therefore, the Denial-of-Service attacks from some malicious network entities are resisted in the UDN.

5.2. Performance Evaluation.

In the practical application environment of the UDN, there are convenient deployment sites for small stations, such as large squares, and they may be limited by topography, such as blocks, stations, and other small stations that are irregularly deployed. Therefore, there are two deployment modes in our simulation: random deployment and regular deployment. To get closer to the

Table 2: Frequency of handover.

Simulation scene (320 ∗ 320 m^2)	UE's speed (km/h)	Handover (times/second/user)
Scene 1: APs randomly deployed, L2 centralized	3	0.731
	30	1.421
	60	2.018
Scene 2: APs randomly deployed, L1 centralized	3	0.771
	30	1.425
	60	2.034
Scene 3: APs regularly deployed, L2 centralized	3	0.579
	30	1.228
	60	1.796

practical application, the macro station is the center of the grid, where 256 APs are regularly deployed. The ISD is 20 m, which corresponds to the grid size of 320 ∗ 320 m^2. Similarly, the macro station is also the center of the grid, where 255 APs are randomly deployed, and the grid size is 320 ∗ 320 m^2.

Considering that the virtual cell is the direction of the future 5G network, the virtual cell is applied in the simulation scenario. The macro station functions as a control plane service entity, and the APs are the user-plane service entities. Since the service entity is dynamically selected when the UE moves among the APs, the best AP should be chosen by the UE to reduce the connection failure rate and improve the throughput. When the dynamic service AP is selected, for L2 (layer two), the service AP delay is changed to 5 ms. When L1 (layer one) is centralized (to similar RRH), the service AP delay is changed to 0 ms. In crowded scenes, users move relatively slowly. Therefore, we select three low-speed scenes: 3 km/h (on foot), 30 km/h (by bike), and 60 km/h (by car). The handover of the UE among the APs is simulated, as shown in Table 2.

In the simulated scene, when the moving UE accesses APs, the handover frequency is equivalent to the frequency of the communication session's establishment. All data communication sessions need different protection keys. The data protected with a traditional symmetric key method (such as LTE encryption algorithm 128-EEA3) and the key storage space requested for data communication can be calculated by formula (5):

$$\text{Sum (storage)} = \text{length (symmetric\_key)} * \text{times (handover)} \quad (5)$$

For instance, the UE continues moving for 30 minutes with the respective speeds of 3 km/h, 30 km/h, and 60 km/h according to Table 2. The required storage capacity can be calculated using the formula above. The results are shown as follows (assuming the SIM card capacity is 32 kB):

128 bit ∗ 30 ∗ 60 ∗ 0.731 = 168422 bits = 20 kB < 32 kB.

128 bit ∗ 30 ∗ 60 ∗ 1.421 = 327398 bits = 40 kB > 32 kB.

128 bit ∗ 30 ∗ 60 ∗ 2.018 = 464947 bits = 56 kB > 32 kB.

128 bit ∗ 30 ∗ 60 ∗ 0.771 = 177638 bits = 22 kB < 32 kB.

128 bit ∗ 30 ∗ 60 ∗ 1.425 = 328320 bits = 40 kB > 32 kB.

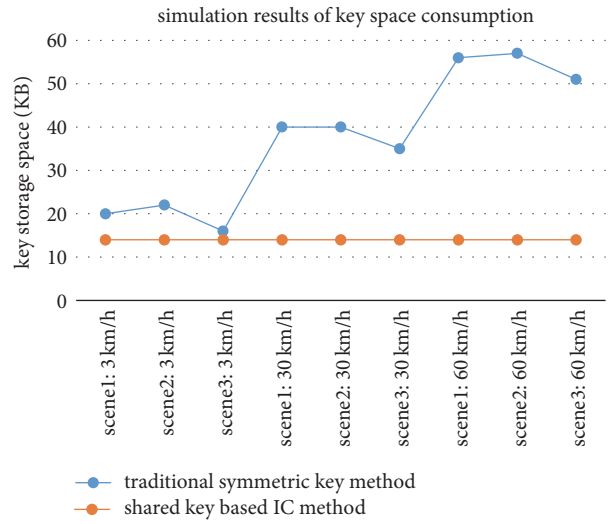


Figure 5: The result of key space consumption.

128 bit ∗ 30 ∗ 60 ∗ 2.034 = 468634 bits = 57 kB > 32 kB.

128 bit ∗ 30 ∗ 60 ∗ 0.579 = 133402 bits = 16 kB < 32 kB.

128 bit ∗ 30 ∗ 60 ∗ 1.228 = 282931 bits = 35 kB > 32 kB.

128 bit ∗ 30 ∗ 60 ∗ 1.796 = 413798 bits = 51 kB > 32 kB.

However, in our solution for secure data communication, the pairwise key that includes the public key and private key is a one-off generation using the restructured parameters. The pairwise key should be saved by the network entities, while the shared keys are instantaneously calculated. The shared keys can be generated many times and do not require storage. Therefore, the keys' storage capacity will be basically stable, and the keys' storage space can be calculated by formula (6):

$$\text{Sum (storage)} = \text{length (pairwise\_key)} * \text{quantity (APs)} \quad (6)$$

225 bit ∗ 2 ∗ 256 = 115200 bits = 14 kB.

The simulation results of the key storage capacity required are shown in Figure 5.

Figure 5, which is based on Table 2 and formulas (5) and (6), compares the key space consumption under three

kinds of UE's speed in traditional symmetric key method and the "shared" key based IC method. In our scenario, by means of the "shared" key based IC method, the key storage space is a constant value, 14 kb, but, with the way of traditional symmetric key method, the key storage space value is dynamic and incremental, which shows that when UE movement rate is greater than 30 km/h, the key storage space is generally greater than 32 KB.

From the above data analysis, we can draw the following conclusions:

(1) The key space consumed by the shared key method based on the IC is significantly less than the space consumption of the traditional symmetric key method.

(2) The key space consumed by the shared key method based on the IC is more stable. In contrast, when using the traditional symmetric key method, the number of protected keys generated by the UE increases with the increasing movement speed.

For 5G UDN, it is very important to have secure and efficient data communications in practical operations. We proposed a scheme where the AKA-IC solution can effectively guarantee the security authentication and data protection among the network entity communication and improve the computational efficiency with less bandwidth.

6. Conclusions

In the 5G, the UDN is an important solution to the explosive growth of network capacity and data traffic. UDN security will directly affect the security of the 5G system. However, there is little research on UDN security. In particular, the data communication security among the network entities of the UDN is still unclear.

In this paper, a new security scheme based on the implicit certificate is introduced based on the analysis of the security challenge of the UDN. We provide the solution that includes the IC and pairwise key generation, and the application process is based on the IC. Then, we analyze the performance of our security communication model. Moreover, an authentication and key agreement protocol based on the IC (AKA-IC) is proposed to solve the secure data communication issue. The AKA-IC algorithm is lightweight and efficient, and the result of the simulated evaluation shows that it is well adapted to various network entities of the UDN, among the APs of APG and between the AP and UE. The security solution based on the IC should be used as an important direction for data communication security for future 5G UDNs.

For future work, in addition to investigating the aforementioned security issues, we also identify other interesting research areas, such as the unified security authentication architecture, the user privacy protection mechanism, and the algorithm optimization of key generation. This will provide more security assurance for 5G systems.

Notations

K: Symmetric root key for initial authentication
r_U: Secret random integer generated by entity$U$
R_U: EC point for the IC request sent by entity$U$
Cert_U: The implicit certificate of the entity$U$
e: The result value from the hash computing of Cert_U
s: The value for the computing private key of the entities
d_U: The private key of entity$U$
Q_U: The public key of entity$U$
K_{UV}: The shared key between entity U and entity V
HMAC: The keyed-hash message authentication code 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.

Authors' Contributions

Zhonglin Chen, Shanzhi Chen, and Hui Xu contributed to the conception and algorithm design of the study. Zhonglin Chen and Hui Xu contributed to the acquisition of simulation. Zhonglin Chen, Hui Xu, and Bo Hu contributed to the analysis of simulation data and approved the final manuscript.

Acknowledgments

This work was supported by the National Science and Technology Major Projects for the New Generation of Broadband Wireless Communication Networks (Grant no. 2016ZX03001017) and the National Natural Science Foundation of China for Distinguished Young Scholars (Grant no. 61425012).

References

[1] S. Chen, "Analysis and Suggestion on Developing 5G," *Telecommunications Science*, vol. 7, pp. 1–10, 2016.

[2] IMT-2020(5G)PG, "WHITE PAPER ON 5G VISION AND REQUIREMENTS_V1.0 [EB/OL]," http://www.imt-2020.cn/zh/documents/1, 2014.

[3] IMT-2020(5G)PG, "WHITE PAPER ON 5G CONCEPT [EB/OL]," http://www.imt-2020.cn/zh/documents/1, 2015.

[4] S. Chen, F. Qin, B. Hu, X. Li, Z. Chen, and J. Liu, *User-Centric Ultra-Dense Networks for 5G*, Springer, Cham, Switzerland, 2017.

[5] Certicom, "Explaining Implicit Certificate," Certicom 2004, Certicom, Mississauga, Canada, 2004.

[6] D. A. Ha, K. T. Nguyen, and J. K. Zao, "Efficient authentication of resource-constrained IoT devices based on ECQV implicit certificates and datagram transport layer security protocol," in *SoICT '16: Proceedings of the Seventh Symposium on Information and Communication Technology*, pp. 173–179, ACM, New York, NY, USA, 2016.

[7] P. Porambage, C. Shmitt, P. Kumar, A. Gurtov, and M. Ylianttlila, "Two-phase Authentication Protocol for Wireless Sensor Networks in Distributed IoT Applications," in *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 2014–2728, IEEE, Istanbul, Turkey, 2014.

[8] A. P. Hansen, "Public Key Infrastructure (PKI) Interoperability: A Security Services Approach to Support Transfer of Trust," 1999.

[9] Z. F. Tian, "Research on security of public key infrastructure (PKI)," *China Safety Science Journal*, vol. 19, no. 2, pp. 116-117, 2009.

[10] "Wikipedia, the free encyclopedia. Implicit certificate [EB/OL]," https://en.wikipedia.org/wiki/Implicit_certificate, 2017.

[11] C. Gentry, "Certificate-based encryption and the certificate revocation problem," in *Advances in Cryptology - EUROCRYPT 2003*, E. Biham, Ed., vol. 2656 of *Lecture Notes in Computer Science*, pp. 272–293, Springer, Heidelberg, Germany, 2003.

[12] B. G. Kang, J. H. Park, and S. G. Hahn, "A Certificate-based Signature Scheme," in *CT-RSA 2004*, T. Okamoto, Ed., vol. 2964 of *Lecture Notes in Computer Science*, pp. 99–111, Springer, Heidelberg, Germany, 2004.

[13] C. Zouridaki, B. L. Mark, K. Gaj, and R. K. Thomas, "Distributed CA-based PKI for Mobile Ad Hoc Networks Using Elliptic Curve Cryptography," in *EuroPKI 2004: European Public Key Infrastructure Workshop*, vol. 3093 of *Lecture Notes in Computer Science*, pp. 232–245, Springer, Samos Island, Greece, 2004.

[14] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[15] SEC 4, *Standards for Efficient Cryptography: Elliptic Curve Qu-Vanstone Implicit Certificate Scheme (ECQV)*, Certicom, Mississauga, Canada, 1.2 edition, 2013.

[16] A. Sojka, K. Piotrowski, and P. Langendoerfer, "Short ECC a lightweight security approach for wireless sensor networks," in *Proceedings of the International Conference on Security and Cryptography, SECRYPT 2010*, pp. 304–308, grc, July 2010.

[17] B. Nair and C. Mala, "Analysis of ECC for application specific WSN security," in *Proceedings of the 6th IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2015*, ind, December 2015.

[18] C. Park, "A Secure and efficient ECQV implicit certificate issuance protocol for the internet of things applications," *IEEE Sensors Journal*, vol. 17, no. 7, pp. 2215–2223, 2017.

[19] N. M. Rabadi, "Improved anonymous group implicit certificate scheme," in *Proceedings of the 2011 IEEE Consumer Communications and Networking Conference, CCNC'2011*, pp. 308–312, usa, January 2011.

[20] B. Vaidya, D. Makrakis, and H. Mouftah, "Effective public key infrastructure for vehicle-to-grid network," in *Proceedings of the 4th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications, DIVANet 2014*, pp. 95–101, can, September 2014.

[21] Li. Yue and P. Mugen, "Layered heterogeneous wireless networking scheme based on virtual cell," *Telecommunications Science*, vol. 1, pp. 8–12, 2013.

[22] S. Chen, F. Qin, B. Hu, X. Li, and Z. Chen, "User-centric ultra-dense networks for 5G: Challenges, methodologies, and directions," *IEEE Wireless Communications Magazine*, vol. 23, no. 2, pp. 78–85, 2016.

[23] Z. Chen, S. Chen, H. Xu, and B. Hu, "Security architecture and scheme of user-centric ultra-dense network (UUDN)," *Transactions on Emerging Telecommunications Technologies*, vol. 28, no. 9, Article ID e3149, 2017.

[24] H. Kaizhi, J. Liang, and Z. Hua, "Research on 5G security threat and protection technologies," *Designing Techniques of Posts and Telecommunications*, vol. 6, pp. 8–12, 2015.

[25] H. Mun, K. Han, and K. Kim, "3G-WLAN interworking: Security analysis and new authentication and key agreement based on EAP-AKA," in *Proceedings of the 2009 Wireless Telecommunications Symposium, WTS 2009*, cze, April 2009.

[26] Y. Park and T. Park, "A Survey of Security Threats on 4G Networks," in *Proceedings of the 2007 IEEE Globecom Workshops*, pp. 1–6, Washington, DC, USA, November 2007.

[27] M. Peng, Y. Li, Z. Zhao, and C. Wang, "System architecture and key technologies for 5G heterogeneous cloud radio access networks," *IEEE Network*, vol. 29, no. 2, pp. 6–14, 2015.