

Wireless Communications and Mobile Computing

# Crowdsourcing for Mobile Networks and IoT

Lead Guest Editor: Xiping Hu

Guest Editors: Zhaolong Ning, Kuan Zhang, Edith Ngai, Kun Bai, and Fei Wang





---

# **Crowdsourcing for Mobile Networks and IoT**

Wireless Communications and Mobile Computing

---

## **Crowdsourcing for Mobile Networks and IoT**

Lead Guest Editor: Xiping Hu

Guest Editors: Zhaolong Ning, Kuan Zhang, Edith Ngai, Kun Bai,  
and Fei Wang



---

Copyright © 2018 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  
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  
Dajana Cassioli, Italy  
Luis Castedo, Spain  
Ioannis Chatzigiannakis, Greece  
Lin Chen, France  
Yu Chen, USA  
Hui Cheng, UK  
Luca Chiaraviglio, Italy  
Ernestina Cianca, Italy  
Riccardo Colella, Italy  
Mario Collotta, Italy  
Massimo Condoluci, Sweden  
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  
Alessandra De Paola, 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. Gomez-Pulido, Spain  
Ke Guan, China  
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 Nicolitidis, 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. Prys, Germany  
Junaid Qadir, Pakistan  
Sujan Rajbhandari, UK  
Rajib Rana, Australia  
Luca Reggiani, Italy  
Daniel G. Reina, Spain  
Abusayeed Saifullah, USA  
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

## **Crowdsourcing for Mobile Networks and IoT**

Xiping Hu , Zhaolong Ning, Kuan Zhang, Edith Ngai , Kun Bai, and Fei Wang  
Editorial (2 pages), Article ID 6231236, Volume 2018 (2018)

## **Intelligent Smoke Alarm System with Wireless Sensor Network Using ZigBee**

Qin Wu , Jiashuo Cao , Chuang Zhou , Ji Huang , Zhuo Li, Shin-Ming Cheng , Jun Cheng ,  
and Guanghui Pan  
Research Article (11 pages), Article ID 8235127, Volume 2018 (2018)

## **A Research on Fast Face Feature Points Detection on Smart Mobile Devices**

Xiaohe Li , Xingming Zhang, and Haoxiang Wang   
Research Article (8 pages), Article ID 9729014, Volume 2018 (2018)

## **Congestion Control and Traffic Scheduling for Collaborative Crowdsourcing in SDN Enabled Mobile Wireless Networks**

Dawei Shen, Wei Yan, Yuhuai Peng , Yanhua Fu, and Qingxu Deng   
Research Article (11 pages), Article ID 9821946, Volume 2018 (2018)

## **Channel Assignment Mechanism for Multiple APs Cochannel Deployment in High Density WLANs**

Jianjun Lei , Jianhua Jiang, and Fengjun Shang   
Research Article (11 pages), Article ID 6931765, Volume 2018 (2018)

## **IoT-B&B: Edge-Based NFV for IoT Devices with CPE Crowdsourcing**

He Zhu  and Changcheng Huang  
Research Article (15 pages), Article ID 3027269, Volume 2018 (2018)

## **An SAT-Based Method to Multithreaded Program Verification for Mobile Crowdsourcing Networks**

Long Zhang , Wanxia Qu , Yinjia Huo, Yang Guo , and Sikun Li  
Research Article (8 pages), Article ID 3193974, Volume 2018 (2018)

## **HuAc: Human Activity Recognition Using Crowdsourced WiFi Signals and Skeleton Data**

Linlin Guo , Lei Wang , Jialin Liu, Wei Zhou, and Bingxian Lu   
Research Article (15 pages), Article ID 6163475, Volume 2018 (2018)

## **A Time and Location Correlation Incentive Scheme for Deep Data Gathering in Crowdsourcing Networks**

Fulong Ma, Xiao Liu, Anfeng Liu , Ming Zhao, Changqin Huang, and Tian Wang   
Research Article (22 pages), Article ID 8052620, Volume 2018 (2018)

## **Network Traffic Prediction Based on Deep Belief Network and Spatiotemporal Compressive Sensing in Wireless Mesh Backbone Networks**

Laisen Nie, Xiaojie Wang , Liangtian Wan , Shui Yu, Houbing Song, and Dingde Jiang  
Research Article (10 pages), Article ID 1260860, Volume 2018 (2018)

**Channel Access and Power Control for Mobile Crowdsourcing in Device-to-Device Underlaid Cellular Networks**

Yue Ma , Li Zhou , Zhenghua Gu, Yang Song, and Bin Wang  
Research Article (13 pages), Article ID 7192840, Volume 2018 (2018)

**CPSFS: A Credible Personalized Spam Filtering Scheme by Crowdsourcing**

Xin Liu, Pingjun Zou, Weishan Zhang, Jiehan Zhou, Changying Dai, Feng Wang, and Xiaomiao Zhang  
Research Article (9 pages), Article ID 1457870, Volume 2017 (2018)

**Color Distribution Pattern Metric for Person Reidentification**

Yingsheng Ye, Xingming Zhang, and Wing W. Y. Ng  
Research Article (11 pages), Article ID 4089505, Volume 2017 (2018)

**Optimal Channel Selection Based on Online Decision and Offline Learning in Multichannel Wireless Sensor Networks**

Mu Qiao, Haitao Zhao, Shengchun Huang, Li Zhou, and Shan Wang  
Research Article (13 pages), Article ID 7902579, Volume 2017 (2018)

**A Crowdsensing-Based Real-Time System for Finger Interactions in Intelligent Transport System**

Chengqun Song, Jun Cheng, and Wei Feng  
Research Article (10 pages), Article ID 7385052, Volume 2017 (2018)

**Optimal Computing Resource Management Based on Utility Maximization in Mobile Crowdsourcing**

Haoyu Meng, Ying Zhu, and Ruilong Deng  
Research Article (13 pages), Article ID 1494851, Volume 2017 (2018)

## Editorial

# Crowdsourcing for Mobile Networks and IoT

**Xiping Hu** <sup>1</sup>, **Zhaolong Ning**,<sup>2</sup> **Kuan Zhang**,<sup>3</sup> **Edith Ngai** <sup>4</sup>, **Kun Bai**,<sup>5</sup> and **Fei Wang**<sup>6</sup>

<sup>1</sup>Chinese Academy of Sciences, Shenzhen, China

<sup>2</sup>Dalian University of Technology, Dalian, China

<sup>3</sup>University of Waterloo, Waterloo, ON, Canada

<sup>4</sup>Uppsala University, Uppsala, Sweden

<sup>5</sup>Tencent Company, Shenzhen, China

<sup>6</sup>Cornell University, Ithaca, NY, USA

Correspondence should be addressed to Xiping Hu; xp.hu@siat.ac.cn

Received 5 February 2018; Accepted 6 February 2018; Published 5 June 2018

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

As the deep integration of ubiquitous sensors, intelligent devices, and social networks, mobile networks and IoT are formed by the opportunity of virtual mobile communication networks and social communities between mobile carriers. People involved in a mobile network can easily interact and share information with each other anytime and anywhere through the popular use of intelligent devices. As a result, there is a remarkable trend to enable crowdsourcing for mobile networks and IoT to address various problems that involve real-time collection, processing, and collaborations among participants in highly mobile environments. Thus, crowdsourcing could be an efficient strategy to improve quality and user experiences of applications in mobile networks and IoT, which not only potentially brings enormous benefits for economics but also leads to revolution for our daily life. The embedded sensors including accelerometer, compass, gyroscope, GPS, microphone, and camera in mobile phones are leveraged to gather the required information to support location-based services, for example, environmental measurements, personal activity sharing, and online recommendation. In this special issue on crowdsourcing for mobile networks and IoT, we have invited some papers that address such issues.

L. Nie et al. introduce a network traffic prediction method in wireless mesh backbone networks based on deep learning and spatiotemporal compressive architecture. This method applies discrete wavelet transform to extract the low-pass component of network traffic. The performance of this method is verified by comparing with three widely used traffic prediction methods.

The paper entitled “CPSFS: A Credible Personalized Spam Filtering Scheme by Crowdsourcing” proposes a credible personalized spam filtering scheme and classifies spam into two categories, that is, complete-spam and semispam, before filtering them. According to the social trust and interest similarity, complete-spam can be filtered by the Bayesian filtering, and semispam can be estimated by crowdsourcing at the client side.

In order to maximize network utility, H. Meng et al. demonstrate an optimal real-time pricing strategy for computing resource management in mobile crowdsourcing. Furthermore, the existence of real-time prices is proved, which can align individual optimality with systematic optimality.

H. Zhu et al. allocate the sharing resource to users across the network edge. A novel architecture is proposed to share resource of physical customer-premised equipment nodes across the network edge and assign virtual customer-premised equipment instances to a cost-efficient node.

The paper entitled “An SAT-Based Method to Multi-threaded Program Verification for Mobile Crowdsourcing Networks” presents a novel IC3-based algorithm on the safety verification of the multithreaded programs for mobile crowdsourcing networks. The performance of the proposed algorithm is evaluated by the SAT-based model checking algorithms, focusing on memory consumption.

Y. Ye et al. introduce a color distribution pattern metric method, concentrating on reidentification in video searching for surveillance and forensic fields in crowdsourcing IoTs. Performance evaluations show that the presented method on different datasets can obtain higher network accuracy.

*Kuan Zhang  
Edith Ngai  
Kun Bai  
Fei Wang*

L. Guo et al. present a WiFi-based public activity framework by combining channel state information and crowd-sourced skeleton joints to improve the robustness and accuracy of activity recognition. The experiments show that the proposed method achieves high recognition accuracy in different datasets.

J. Lei et al. introduce a channel assignment algorithm based on point coordination function for cochannel deployment of access point in WLANs. Furthermore, the experiments are carried on networks with different densities. The results show the presented method can obtain high throughput, low packet loss rate, and bounded access delay compared with the existing methods.

The work from Y. Ma et al. focuses on the channel access and power control problem in device-to-device underlaid cellular networks. A novel semidistributed network-assisted power and channel access control scheme for D2D user equipment is proposed, and the achieved performances by cellular and D2D links are both evaluated.

The paper entitled “A Crowdsensing-Based Real-Time System for Finger Interactions in Intelligent Transport System” demonstrates a real-time projector-camera finger system based on crowdsensing, in which users can interact with a computer by bare hand touching on arbitrary surface. This designed system can be applied as an intelligent device in transport systems.

M. Qiao et al. introduce a channel selection strategy with hybrid architectures, combining the centralized and the distributed methods, which can reduce the overhead of access point and provide more flexibility in network deployment. Based on the self-decision algorithm and offline self-learning algorithm, the presented strategy is investigated in multi-channel wireless sensor networks. The theoretical analysis and performance evaluation depict the effectiveness of the studied algorithm.

The paper entitled “A Time and Location Correlation Incentive Scheme for Deep Data Gathering in Crowdsourcing Networks” addresses the problem of optimizing the quality of sensor data and collection enough sensor data in sensor data gathering system. Based on the greedy strategy, two effective algorithms are proposed for motivating smartphone users to participate in the smartphone-based crowdsourcing sensing.

Q. Wu et al. study a low precision and weak timeliness required smoke system. The implementation of intelligent smoke alarm system, containing sensor network, classification algorithm, and visual interface, is presented to monitor the fire in the weak electromagnetic environment.

The paper written by X. Li et al. proposes a supervised descent method. It leverages the performance of face feature points detection on mobile terminals and proves the optimization of the presented algorithm.

D. Shen et al. concentrate on the traffic scheduling and load balancing problem in software-defined mobile wireless networks and present a crowdsourcing-based routing forwarding scheme as well as a congestion control algorithm to solve the formulated problem.

*Xiping Hu  
Zhaolong Ning*

## Research Article

# Intelligent Smoke Alarm System with Wireless Sensor Network Using ZigBee

Qin Wu <sup>1,2</sup>, Jiashuo Cao <sup>2</sup>, Chuang Zhou <sup>2</sup>, Ji Huang <sup>2</sup>, Zhuo Li,<sup>2</sup>  
Shin-Ming Cheng <sup>1</sup>, Jun Cheng <sup>3,4</sup> and Guanghui Pan<sup>2</sup>

<sup>1</sup>Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan

<sup>2</sup>School of Computer Science, Chengdu University of Information Technology, Chengdu, China

<sup>3</sup>Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China

<sup>4</sup>The Chinese University of Hong Kong, Shatin, Hong Kong

Correspondence should be addressed to Shin-Ming Cheng; smcheng@mail.ntust.edu.tw and Jun Cheng; jun.cheng@siat.ac.cn

Received 24 September 2017; Revised 28 December 2017; Accepted 17 January 2018; Published 22 March 2018

Academic Editor: Kun Bai

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

The conflagration of fire is still a serious problem caused by humans, and houses are at a high risk of fire. Recently, people have used smoke alarms which only have one sensor to detect fire. Smoke is emitted in several forms in daily life. A single sensor is not a reliable way to detect fire. With the rapid advancement in Internet technology, people can monitor their houses remotely to determine the current condition of the house. This paper introduces an intelligent smoke alarm system that uses ZigBee transmission technology to build a wireless network, uses random forest to identify smoke, and uses E-charts for data visualization. By combining the real-time dynamic changes of various environmental factors, compared to the traditional smoke alarm, the accuracy and controllability of the fire warning are increased, and the visualization of the data enables users to monitor the room environment more intuitively. The proposed system consists of a smoke detection module, a wireless communication module, and intelligent identification and data visualization module. At present, the collected environmental data can be classified into four statuses, that is, normal air, water mist, kitchen cooking, and fire smoke. Reducing the frequency of miscalculations also means improving the safety of the person and property of the user.

## 1. Introduction

The wide deployment of sensing technologies in our daily living environments and the pervasive usage of mobile devices bring about great opportunities for the deployment of smoke alarms [1]. At present, most of the traditional smoke alarm systems consist of sensor modules, transmission lines, and monitors, each of which is connected to a sensor module [2]. Therefore, there is a defect in the accuracy and timeliness of the traditional alarm system. The misinformation of the fire alarm caused by water mist or oil smoke causes unnecessary losses to people. People spend a long time in vehicles on roads after homes and offices [3, 4]; they want to get the right fire alarm while they are in the distance. And there are two main reasons for these problems.

(1) *The Way to Judge Whether or Not Fire Is Too Simple.* The traditional smoke alarm system only detects a single environmental value. If the detection value exceeds the threshold, it is judged to be a fire. There are many uncertainties in the fire scene that will cause the smoke alarm system to be unable to accurately determine the fire, such as temperature, combustible gas concentration, smoke particle concentration, and barometric pressure, which can lead to false positives, omission, delay reporting, and other phenomena which occur frequently [5].

(2) *Inadequate Component Connection.* The traditional smoke alarm system use copper wires, insulated wires, or cables to connect detectors and alarms. This kind of connection has many disadvantages, such as copper wire's price, large

consumption, and weak anti-interface ability. On the other hand, the copper wire easily wears at high temperatures, resulting in the alarm system maintenance being very complex, reducing the reliability of the traditional smoke alarm system [6].

At present, some new smoke alarms use LoRa to achieve wireless communication, and they do not need to install wires. This solves the problem of difficult installation of traditional smoke alarms and enables people to view the status of the alarm remotely, so as to ensure the timeliness of the alarm. But there was no change in the way the fire was judged, and the accuracy of the alarm was not improved.

For the first question, we believe that using the machine learning classification algorithm and a variety of sensors to monitor the maximum real environment can restore the real scene of the environment approximately and greatly enhance the accuracy of fire warning, while reducing false, omission, and late alarms' frequency.

When designing the communication system, we want it to be able to support multipoint data transmission with low complexity, low cost, and high reliability. So, we choose to use ZigBee. In addition, we designed and implemented data visualization on the web to ensure that the users can monitor their homes remotely. Experiments show that the intelligent smoke alarm system has a high reliability in data transmission and fire alarm, can monitor multiple scenes at the same time, and has high practicability.

In this paper, the combustion process is analyzed, and a variety of WLAN technologies and ML algorithms are compared. The sensor type of the system is given, and the usability and reliability of the system are tested through simulation experiments. Finally, the conclusion is given and the improvement scheme is put forward.

## 2. Related Work and Background

*2.1. Changes in the Indicators in the Process of Conflagration.* The conflagration is a disaster that is not controlled by man and caused by combustion [7]. The three basic elements of the fire are combustible, comburent, and ignition source. The combustible material is in the form of gas, solid, and liquid; "comburent" mainly refers to oxygen. For combustible gas combustion, according to the mixture of combustible gas and air, it can be divided into two different ways. If it is in the combustion before the air has been mixed with the gas, it is called premixed combustion; if the air and combustible gas do not enter the combustion state at the same time but are mixed and burned, this is called diffusion combustion. Liquid and solid substances are condensed matter, which is difficult to mix well with air. The basic process of combustion is as follows: when it gets enough energy from the outside, the condensed matter evaporates into steam or decomposes, and the combustible gas molecules, ashes, and unburned matter particles are suspended in the air, called aerosols. Usually, aerosol molecules are relatively small. During the production of aerosols, large molecules of solid or liquid particles are produced at the same time, known as smoke. But when burning, heat will be generated, causing a temperature rise, while generating a lot of smoke; with the temperature,

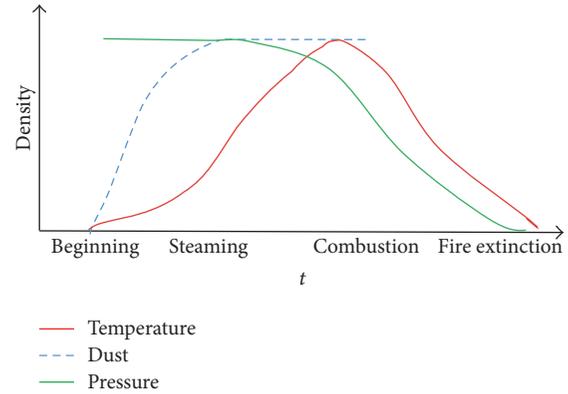


FIGURE 1: The change curve of each indicator in the process of fire [9].

pressure, and smoke dust parameters, it can be determined whether the fire occurred [8].

In general, combustibles while burning produce the following several forms of expression as shown in Figure 1: for liquid and solid combustible materials, the first produced is combustible gas, followed by smoke; in case of sufficient combustion, gas can only be fully burned, releasing a lot of heat to promote the current ambient temperature. In the process of fire, the initial stage produces a huge amount of smoke, but the temperature is not very high. If the detector at this stage begins to test, you can minimize the loss caused by the fire. After the fire starts, the fire will quickly spread and produce a lot of heat to the current environment, increasing the temperature and burning oxygen, so that the air pressure is reduced. If at this moment the current temperature and pressure can be effectively detected, the fire can be controlled.

*2.2. WLAN Technology Comparison.* We analyzed the most widely used technologies in the field of intelligent home wireless communication technologies, including Wi-Fi, Bluetooth, Z-Wave, and ZigBee.

Table 1 contains rough comparisons between different wireless technologies. It is conceptual in nature. However, it may be used as a quick tool for making a decision on which technology you will use in your application. It includes data on the range, data rate transmission, possibility of network building, and so on.

### 2.3. Analysis of ZigBee Network Architecture

*2.3.1. Star Network.* Star network structure control and synchronization are relatively simple, usually used for a small number of node occasions [10]. The disadvantage of this topology is that there is only one path for data routing between nodes. The coordinator may become the bottleneck of the entire network. Figure 2 shows the star network structure.

*2.3.2. Cluster-Tree Network.* The cluster-tree network structure can accommodate more nodes than the star network structure and can also expand the network as needed [10].

TABLE 1: Comparison of WLAN technologies.

	Wi-Fi (802.11) [20]	Bluetooth (BLE) [21]	ZigBee [22]	Z-Wave [23]
Transfer rate	11 and 54 Mbits/s	1, 3, and 24 Mbits/s	20, 40, and 250 Kbits/s	9.6 Kbits/s
Power consumption	10–50 mA	0.6 mA	1.2 $\mu$ .	
Effective distance	100 meters	10 meters	100 meters	100 meters
Frequency	2.4 GHz	2.4 GHz	868,900~928 MHz and 2.4 GHz	868~908.42 MHz
Networking topology	Point to hub	Ad hoc, small networks	Ad hoc, peer-to-peer, star, tree, mesh	Ad hoc, peer-to-peer, star, tree
Security	WEP, WPA (low)	64- and 128-bit encryption (medium)	128 AES plus application layer security (high)	128 AES (high)
Typical applications	Wireless LAN connectivity, Internet access	Wireless PAN connected with devices	Industrial control and monitoring, sensor networks, home control and automation	Sensor networks, building automation

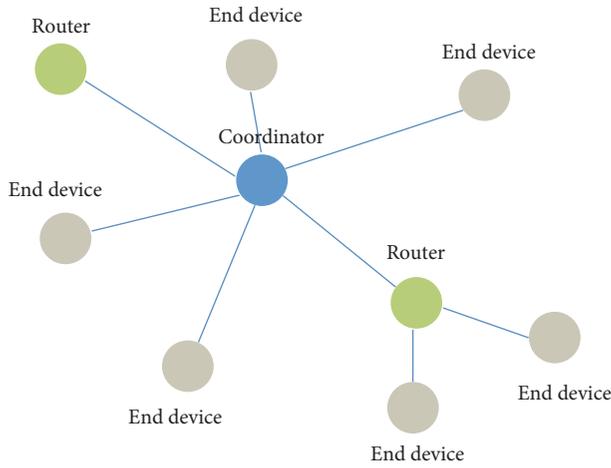


FIGURE 2: Star topology.

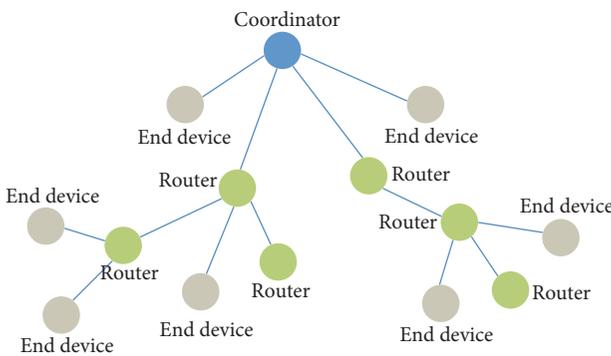


FIGURE 3: Cluster-tree topology.

The disadvantage of this topology is that the information is only a unique routing channel; once the routing node is paralyzed, the subnetwork below it will be completely paralyzed. Figure 3 shows the cluster-tree network structure.

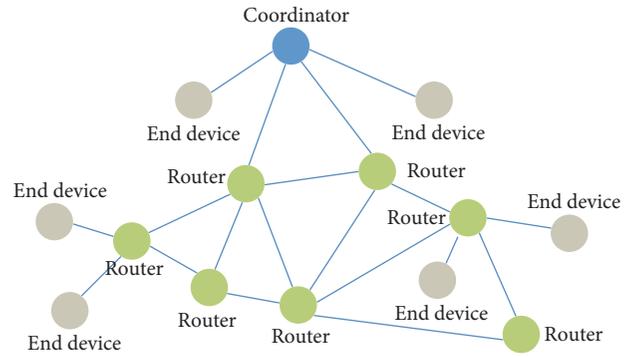


FIGURE 4: Mesh topology.

2.3.3. *Mesh Network.* The advantage of this topology is that it automatically finds the best path, reducing the message delay, and enhances the reliability of the system; the disadvantage is the need for more storage space [10]. Figure 4 shows the mesh network structure.

2.4. *Classification Algorithm Analysis.* In order to improve the alarm speed of the system, a classification algorithm is needed to ensure the accuracy of the same time with faster processing speed. According to this requirement, we analyzed the algorithm with excellent classification accuracy and processing speed in the case of small data volume, including SVM, decision tree, and random forest [11].

2.4.1. *Support Vector Machine.* Support vector machine (SVM) was first proposed by Cortes and Vapnik in 1995. Support vector machines show many unique advantages in solving small sample, nonlinear, and high-dimensional pattern recognition and can be extended to function fitting and other machine learning problems, which in many areas has been widely used. SVM is based on the Vapnik-Chervonenkis dimension theory and the structural risk minimization

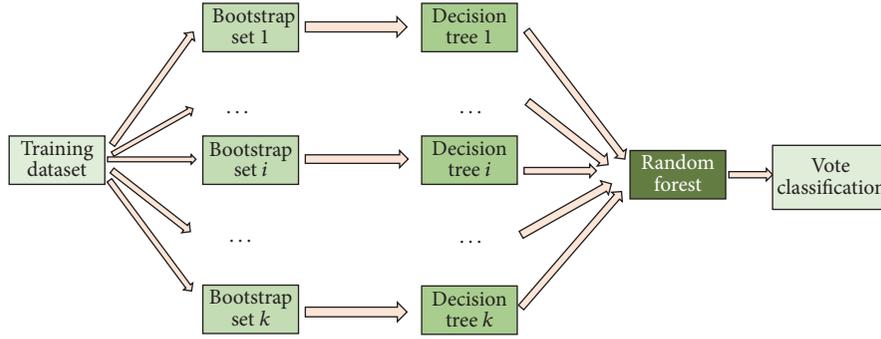


FIGURE 5: Random forest process.

TABLE 2: Classification algorithm effect comparison.

	Error rate	Error rate of train sets (5-fold CV)	Error rate of test sets (5-fold CV)
Decision tree	0.01599247	0.01599253	0.01599337
Bagging	0.01599247	0.01564398	0.01599337
Random forest	0.00047037	0.00047038	0.01270257
Support vector machine	0.01081844	0.01046568	0.01740293
$K$ -nearest neighbor	0.00423330	0.00282222	0.02916432

principle of statistical learning theory [12]. Based on the complexity of the model information in the model (i.e., the learning accuracy of the specific training samples) and the learning ability, we accurately predict the ability of any sample to seek the best compromise between them in order to obtain the best promotion ability (i.e., generalization ability). The basic principle of the support vector machine is to find a reliable kernel of high-dimensional space to a high-dimensional space so that these points are linearly separable in the new space. Support vector machine is based on the linear division and the use of linear division of the principle of the final classification of prediction [13].

**2.4.2. Decision Tree.** Decision tree is a basic classification method, based on the process of feature classification, originally originated from the ID3 algorithm proposed by Quinlan. A decision tree is a method of defining a Boolean function whose input is a set of properties describing the object, and the output is yes or no. Decision trees represent a hypothesis that can be written as a logical formula. The decision-making tree's ability to express is limited to propositional logic, and any test of any of the properties of the object is a proposition [14].

Within the context of propositional logic, the decision tree's ability to express is complete. A decision tree can represent a decision process that determines the classification of the training set, and each node of the tree corresponds to an attribute name or a specific test that divides the training set based on the possible outcome of the test. Each part of the division corresponds to a classification subproblem of the corresponding training case subspace, which can be solved by a decision tree. Thus, a decision tree can be seen as a division of the target classification and access strategy [15].

**2.4.3. Random Forest.** Random forest is a random way to use the decision tree as a base classifier into a forest algorithm, for the prediction of classes, so that each decision tree in the forest is used to make a prediction of the sample. Finally, the results of all the decision trees are combined to obtain high-precision classification results. It is a kind of statistical learning theory, randomized by bootstrap resampling method, from the original training sample set to extract multiple versions of the sample set. The decision tree for each sample set training models the final combination of all decision trees. The final forecast can be established by voting mechanism [16]. The construction of the random forest is shown in Figure 5.

**2.5. Classification Algorithm Effect Comparison.** Using cardiocography (CTG) data [17] as an example, the data has 2126 observations and 23 variables, including the fetal heart rate (FHR) and the uterine contraction (UC) characteristics classified by experts based on guardianship records. For the comparison of decision tree, bagging, random forest, SVM, and  $k$ -nearest neighbor method, bagging is a method for generating multiple versions of a predictor and using these to get an aggregated predictor [18], and the  $k$ -nearest neighbors algorithm is a nonparametric method used for classification and regression [19]. We use R language to fit each method and use 5-fold cross-validation method. To balance the three types of dependent variables, the sample size is randomly divided into 5 substitutions, 5 classification methods' error rate, and 5-fold cross-validation results.

As can be seen from Table 2, random forest performance is the best of these five classification methods.

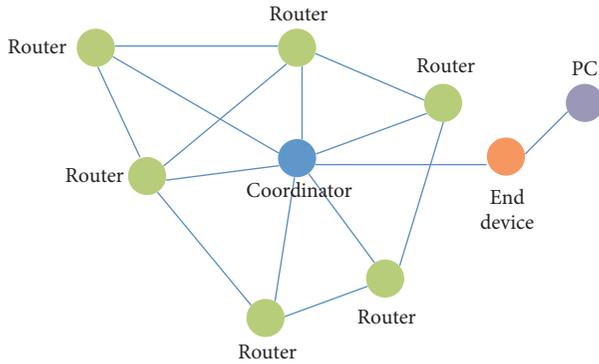


FIGURE 6: Intelligent smoke alarm system network topology.

### 3. System Design

**3.1. Network Topology and System Composition.** On the basis of the traditional smoke alarm, the ZigBee mesh network structure is introduced. It includes several router nodes (router) and a terminal device (end device) connected with the PC and a coordinator node (coordinator). Each router will automatically find the router close to it and will automatically establish connection with these routers. In order to avoid router collisions, it is required that each router can connect to up to three other routers; if the distance exceeds a certain range or the signal strength is lower than a certain range of nodes, connection will be refused [24]. Connected nodes can communicate directly between each other; if two nodes are not connected, the data needs to be forwarded to the target node via the intermediate node.

The improved network structure as shown in Figure 6 has many advantages: the communication range of each node increases; there can be more choices of path; we can choose one of the most stable node forwarding data. When a routing path is damaged, due to the dynamic nature of the entire network, we can reobtain the best routing path in the network and send messages between the various devices, using multihop transmission to increase the coverage of the network which can also transparently transmit data, and can send data to any node as needed.

As shown in Figure 7, the intelligent smoke alarm system consists of a smoke detection module, wireless communication module, and intelligent identification and data visualization module. The smoke detection module is responsible for the environmental data collection of the system. It is composed of a thermometer (temperature sensor), a humidity sensor, air pressure sensor, dust sensor, trophy, and a piece of integrated CC2630 module STM32. The thermometer, humidity sensor, air pressure sensor, and dust sensor are responsible for detecting the current environmental information. STM32 microcontroller is responsible for receiving three types of sensor signals. CC2630 module is responsible for three types of data transmission from the coordinator to end device and then transfers data through the USB serial port to the intelligent identification and data visualization module. Random forest algorithm is implemented for data classification and returns the classification result to the tape

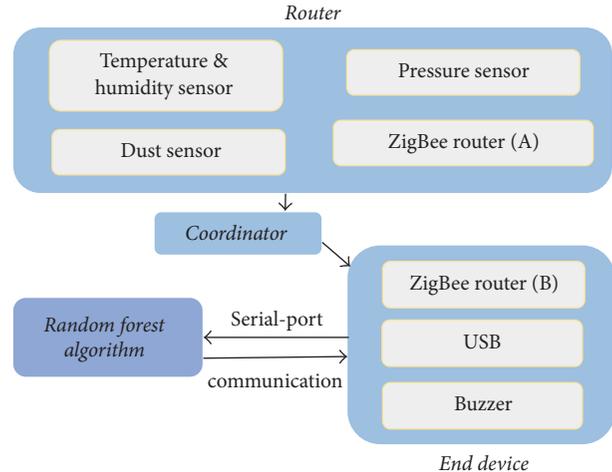


FIGURE 7: System composition diagram.

end device via USB serial port. When the output value is repeated several times for the set alarm value, that is, when the fire generates smoke, the alarm is pushed.

#### 3.2. Hardware Design

**3.2.1. Wireless Communication Module.** The wireless data transmission network is based on Texas Instruments' CC2630 programmable RF chip for development. CC2630 is a high-performance, low-cost, low-power system integration chip that supports Smart Bluetooth, ZigBee, 6LoWPAN, and ZigBeeRF4CE remote control applications development platform. CC2630 dual ARM core 32-bit CPU chip design, with Cortex-M0 core, is responsible for wireless communication processing, and Cortex-M3 core as the main CPU is responsible for ZigBee protocol processing. This architecture allows the CC2630 to build a larger network, a more stable network, and more capacity data when building a ZigBee protocol [25].

CC2630 CPU acts as the whole node subsystem. On the one hand, it is responsible for collecting the sensor in the room to monitor the real-time data and the data along the routing path to the target terminal equipment. On the other hand, it garners the data from surrounding neighbors to establish contact when forwarding the request and data is forwarded to the next node. In addition, the system is set up as a network status indicator; you can use the indicator to determine whether the current node is online or offline or send and receive data status.

The sensors used in this system include temperature and humidity sensors, pressure sensor, and dust sensor. Multiple environmental data including temperature, humidity, barometric pressure, PM2.5 concentration, and PM10 concentration can be monitored in real time. The use of multiple types of sensors avoids the errors of the single indicators in the prior part. And through wireless communication, the environment acquisition module can be placed with the user at random to carry out real-time environmental data collection, which greatly improves its practicability.

The entire system consists of four replaceable ordinary 1.5 V dry batteries, and the supply voltage is 6 V. In addition,

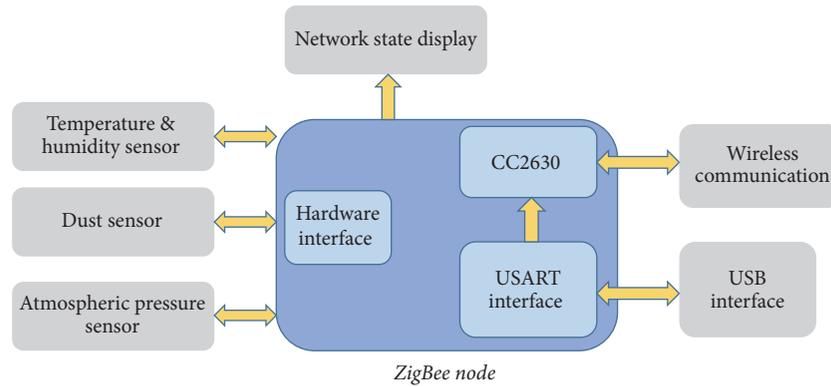


FIGURE 8: Router structure diagram.

the system also uses a low-power optimization program. Running the specified task will trigger the low-power sleep mode, until a wakeup signal is issued.

For the general design of the module, please see Figure 8.

**3.2.2. Smoke Detection Module.** At present, the smoke detectors on the market are mainly infrared detectors. The principle is in the smokeless state, the photosensitive element cannot receive the infrared light, and the smoke particles will refract and reflect to the infrared light, so that the photosensitive element receives the infrared light. When the photosensitive elements receive enough optical signals, the electrode distortion is generated to trigger the alarm signal [26]. But its shortcomings are also very obvious: water mist from kitchen, cooking fumes, and even the haze in winter are likely to trigger the smoke sensor, resulting in frequent false positives.

In order to distinguish between the above-mentioned several different situations, it is necessary to collect different types of environmental data, such as the temperature, humidity, air pressure, and smoke concentration which are the environmental variables that play a decisive role. Therefore, we need to select the sensor as accurately as possible for these four environmental parameters, and the sensor is required to have good enough sensitivity to react very quickly to the environment [27].

For the barometric pressure data, we use the barometric pressure sensor GY-68 BMP180, which can be used for GPS precise navigation (dead reckoning, upper and lower bridge detection, etc.), indoor and outdoor navigation, leisure, sports and medical health monitoring, weather forecast, vertical speed indicating (rising/sinking speed), fan power control, and other fields. The sensor is LCC8 package, whose size is  $3.6 \text{ mm} \times 3.8 \times 0.93 \text{ mm}$ . In low-power mode, the measured air pressure can be accurate up to  $0.06 \text{ hPa}$  ( $0.5 \text{ m}$ ). In high linear mode, the measured air pressure can be accurate up to  $0.03 \text{ hPa}$  ( $0.25 \text{ m}$ ), and the I2C interface is used, the control is simple, and the communication is regular. Compatibility can be used for a variety of processing equipment [28].

For environmental humidity and temperature data, we use the temperature and humidity sensor DHT11; the sensor can be used in laboratories, industrial production, warehouse

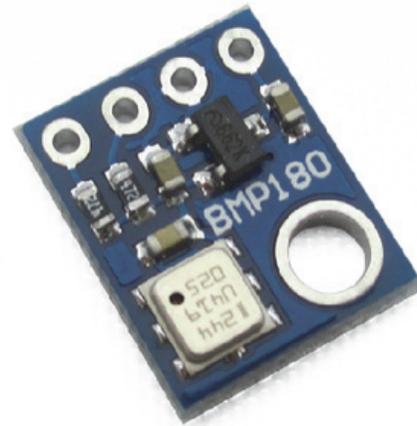


FIGURE 9: GY-68 BMP180.

storage, indoor environmental monitoring, air conditioning devices, intelligent monitoring equipment, and other fields. The sensor humidity measurement ranges from 20% to 95%, the temperature measurement ranges from 0 degrees to 50 degrees, and the use of digital output form is simple and convenient and is easy to control [29].

For the particulate matter in the ambient air, we use the GP2Y1051AU00F dust sensor, which can be used in the field of PM2.5 detector, purifying new fan, filtration system, and so on. The sensor outputs an analog voltage proportional to the measured dust concentration, with sensitivity of  $0.5 \text{ V}/0.1 \text{ mg}/\text{m}^3$ , the smallest particle detection value being 0.8 microns in the range of  $-10$  degrees to 65 degrees, which can detect small particles.

Figures 9, 10, and 11, respectively, correspond to the pressure sensor, the temperature and humidity sensor, and the dust sensor.

### 3.3. Algorithm

**3.3.1. Random Forest.** The random forest algorithm makes good use of randomness (including randomly generated sub-sample sets, random selection of subfeatures), minimizes the relevance of the trees, and improves the overall classification



FIGURE 10: DHT11.



FIGURE 11: GP2Y1051AU0F.

performance, and because the time of each tree is very short and the forest can be parallelized, the random forest classification is very fast.

Assume a random forest classifier  $\{h_i(x, \theta_i, i = 1, \dots, N)\}$ ; the class label of the classification result is obtained by each decision tree  $h_i(x, \theta_i)$  and probability averaging for the test instance  $x$ . The environmental information (barometric pressure, humidity, temperature, and particulate matter) is collected by the system; the prediction class tag  $c_p$  includes normal air, water mist, kitchen cooking, and fire smoke. The classification process is shown below:

$$c_p = \underset{c}{\operatorname{argmax}} \left( \frac{1}{N} \sum_{i=1}^N I \left( \frac{n_{h_j, c}}{n_{h_j}} \right) \right), \quad (1)$$

$$c_p = \underset{c}{\operatorname{argmax}} \left( \frac{1}{N} \sum_{i=1}^N w_i \left( \frac{n_{h_j, c}}{n_{h_j}} \right) \right),$$

where  $\operatorname{argmax}$  denotes the parameter  $c$  with the highest score,  $N$  denotes the number of decision trees in the random forest,  $I(*)$  denotes the exponential function,  $n_{h_j}$ ,  $c$  denotes the classification result of the decision tree for the class  $c$ ,  $n_{h_j}$

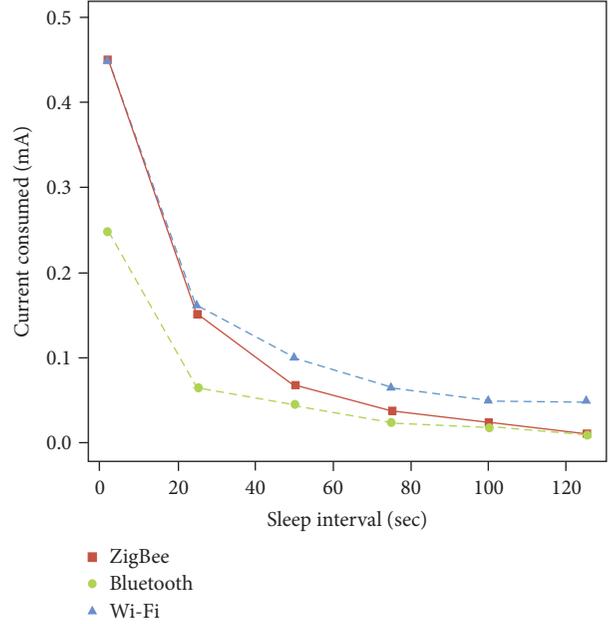


FIGURE 12: Power consumption for ZigBee, Bluetooth, and Wi-Fi.

represents the number of leaf nodes of the decision tree  $h_j$ , and  $w_i$  represents the weight of the  $i$  tree in the random forest [30].

**3.3.2. Data Visualization.** Data visualization is based on the open-source project E-charts implementation. E-charts is a pure JavaScript graphical library, relying on the lightweight Canvas Library Z-Render, providing intuitive, vivid, interactive, highly personalized data visualization chart. Innovative drag-and-drop recalculation, data view, range roaming, and other features greatly enhance the user experience, giving the user data mining and integration capabilities [31].

In this system, taking into account the diversity of data, and in order to create a more intuitive viewing experience, we choose to use the line graph which shows the various types of data over time numerical changes. First, in the front page script tags are used to introduce E-charts master files and JQuery. Then, we use PHP for data processing, including extracting the data from the database and converting it into a JSON format. Finally, in the front, the page uses the \$.get JSON method to implement asynchronous loading of data, which is initialized by the *echarts.init* method.

In order to prevent the last data volume caused by excessive display or display error, due to the data storage time interval, we set to show only nearly half of the data.

## 4. Experimental Design and Verification

**4.1. Power Consumption Comparison.** We use MATLAB to build a virtual system for testing the power consumption of the three WLAN (ZigBee, Bluetooth, and Wi-Fi) technologies during sleep interval [32]; the results are shown in Figure 12. According to the experimental results, ZigBee power consumption is lower than Wi-Fi in all sleep intervals,

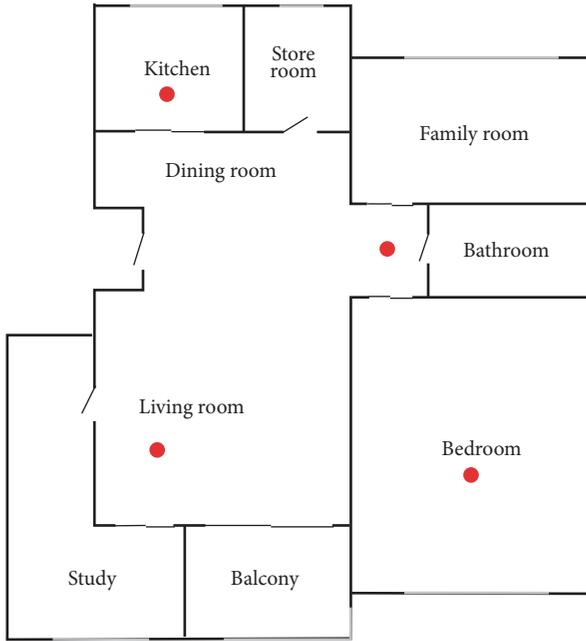


FIGURE 13: Test house plan.

and in the case of long sleep interval, ZigBee and Bluetooth are close to 0.

**4.2. System Effectiveness Testing.** In order to verify the performance of the system, this study selected a house with a living area of about 110 square meters as a test site (located in Shuangliu District, Chengdu, Sichuan Province). The house includes one bedroom, one bathroom, one family room, one kitchen, one store room, one living room, one balcony, and a study. The layout of these rooms is shown in Figure 13.

In order to better test the classification of random forest in each case, we produced four smoke detection modules, ensuring that the data transmission is stable and the packet acceptance success rate is higher than 90% of the case [33]. The four smoke detection modules are placed next to the bedroom door, bathroom ceiling, kitchen ceiling, and living room sofa. Then, we started to set the simulation. Scene 1: at the same time, in the bedroom, we lit a bunch of clothes to simulate a fire smoke and cooking in the kitchen to produce fumes, closed the bathroom door, opened hot water to produce water mist, and in the living room did nothing to ensure that the air status is normal. After a number of simulations, we started to set up the simulation. Scene 2: at the same time, in the living room, we lit a bunch of clothes to simulate a fire smoke, boil water in the kitchen to produce water mist, and in the bedroom and the bathroom we did nothing to maintain the normal air state, and we carried out the same number of simulations. These two types of simulation scenarios are the settings that we think are most close to the life scene. Table 3 is a summary of these two simulated scenarios.

A total of 2,5006 observations and 5 variables were collected from April and May 2017, including the normal

TABLE 3: Air condition in different rooms under different scenes.

	Bedroom	Kitchen	Bathroom	Living room
Scene 1	Fire	Cooking oil	Water mist	Normal
Scene 2	Normal	Water mist	Normal	Fire

environment, fire smoke, fumes, and water mist. The random forest algorithm was used to classify these data. After analyzing the classification results, the error rate was found to be 0. Figure 14 shows the results of random forest identification. It can be seen from the figure that the air conditions have obvious characteristics under these four indicators.

**4.3. Visualization of Experimental Data.** Figure 15 shows the visualization of smoke concentration (unit:  $\text{mg}/\text{m}^3$ ) in the living room at 09:54 to 10:40 on April 13. Since the collected data is updated every second, in order to prevent the data from being displayed too fast, the mechanism of automatic updating every two minutes is adopted to ensure the timeliness of the data and the normal display function.

## 5. Discussion

According to the simulation scenario testing, we can see that the system performed well in the indoor smoke classification and can meet the needs of users for remote monitoring and has certain practical significance for realizing the intelligent home environment monitoring. But the whole system can also be improved in function.

First of all, the current system can only identify four types of circumstances: the normal environment, fire smoke, fumes, and water mist environment; in fact, smoke is generated in several forms, such as secondary smoke generated from primary smoke and construction site smoke.

Second, in the data visualization, the current visualization page can only provide a simple browsing function and cannot perform complex operation such as data extraction.

Moreover, the system is currently vulnerable to network communications and external intrusion. For the whole system, we must have a noise security mechanism; this is an important future research direction.

Finally, when the number of routers in the system keeps increasing, the amount of data will multiply, making it easy for data to overflow, generating some unpredictable errors in the stored procedure. The test only used four smoke detectors and has not yet met the problem of excessive data volume. In order to prevent the possibility of simultaneous storage errors, our current solution is to set up a separate data storage cycle for each smoke detection module. But later, we need to optimize these two issues.

## 6. Conclusion

In this paper, we introduced an implementation method of intelligent smoke alarm system, including sensor network, classification algorithm, and visual interface. Then, the problem of low precision and weak timeliness of traditional smoke alarms is given. The solution of smoke information

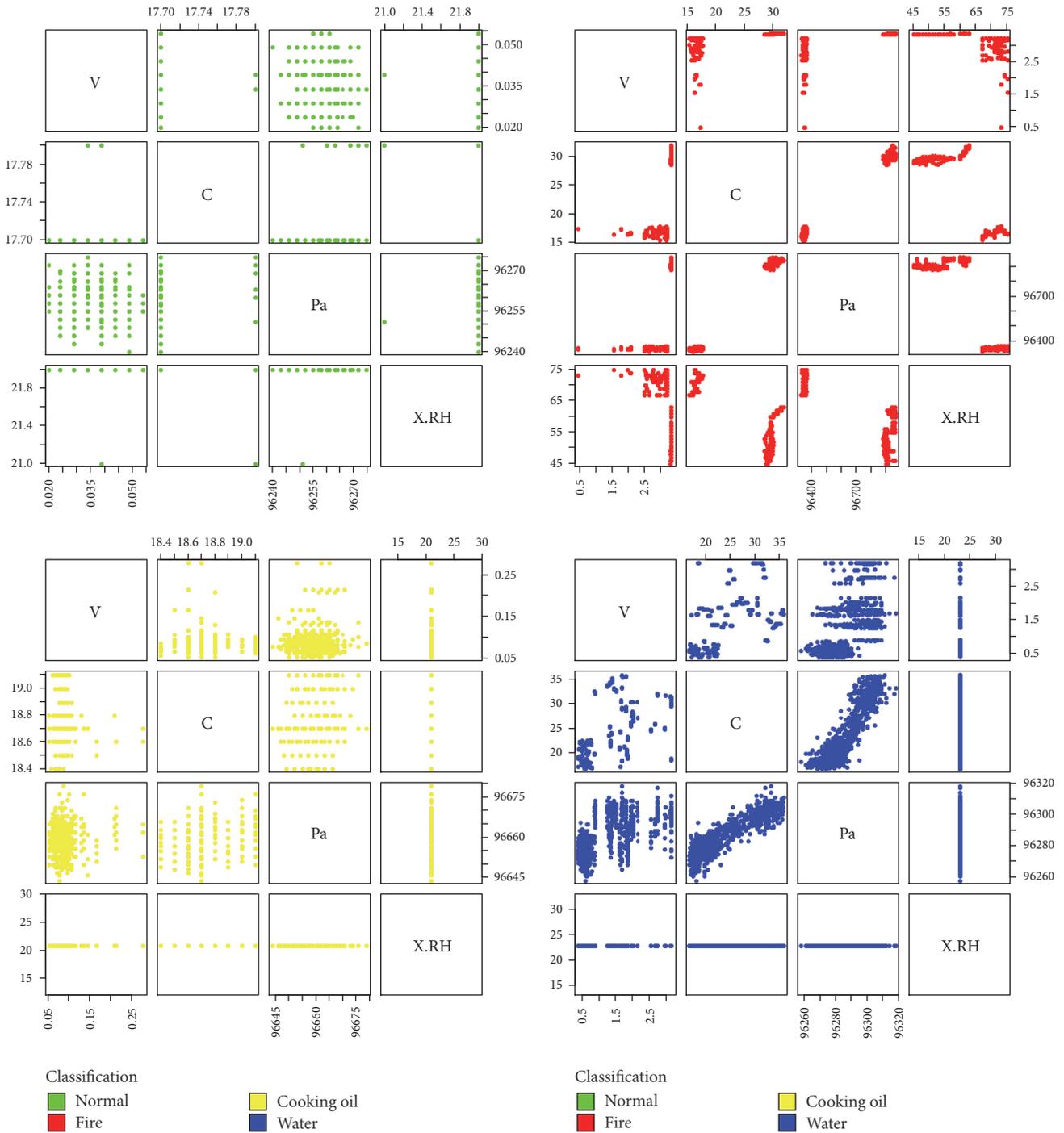


FIGURE 14: The distribution of temperature, pressure, humidity, and dust in different environments. Data sets were classified by temperature, pressure, humidity, and dust.

classification and display and component wireless sensor network is put forward. (1) Smoke information classification solves the problem of low accuracy of traditional smoke alarms. (2) Smoke information visualization can help users understand the air situation more intuitively. (3) Wireless sensor networks can improve the timeliness of the alarm. The experimental results show that, based on the proposed

intelligent smoke alarm system, the air condition can be more accurately distinguished, and the probability of false alarms is greatly reduced, so that the air information can be viewed in real time and the alarm information can be received in a timely manner. In addition, we have discussed some of the shortcomings of the current system. In the future, we will try to identify more types of smog in the same sensor system.

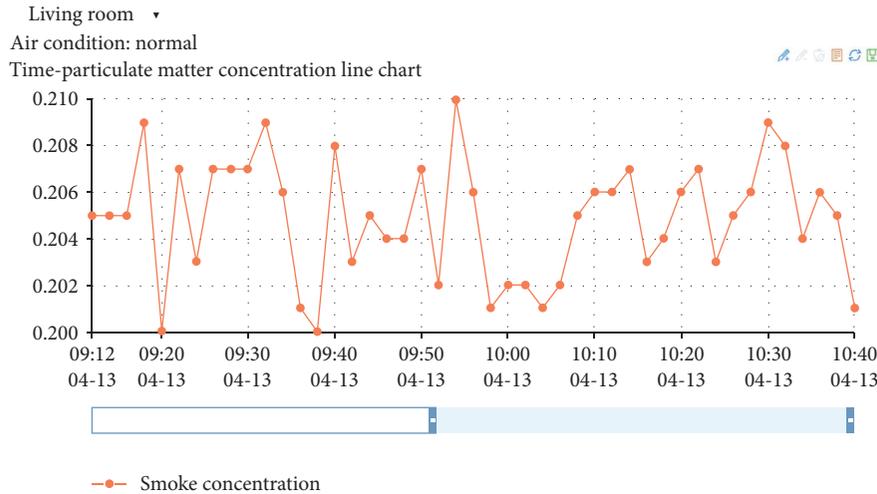


FIGURE 15: Time-smoke concentration line chart.

And the visual function is extended to add more user customization options and enhance the security of the system to ensure the stability of data transmission and storage. Finally, consider the system's requirements for the communication environment; the intelligent smoke alarm system should be used in the weak electromagnetic environment such as the family.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] X. Hu, T. H. S. Chu, H. C. B. Chan, and V. C. M. Leung, "Vita: a crowdsensing-oriented mobile cyber-physical system," *IEEE Transactions on Emerging Topics in Computing*, vol. 1, no. 1, pp. 148–165, 2013.
- [2] M. Yang and C. Zhang, "Smoke Alarm System," *Wireless*, vol. 9, pp. 50–51, 2006.
- [3] X. Hu, J. Zhao, B.-C. Seet, V. C. M. Leung, T. H. S. Chu, and H. Chan, "S-afame: agent-based multilayer framework with context-aware semantic service for vehicular social networks," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 1, pp. 44–63, 2015.
- [4] Y. Guo, X. Hu, B. Hu, J. Cheng, M. Zhou, and R. Y. Kwok, "Mobile cyber physical systems: current challenges and future networking applications," *IEEE Access*, no. 99, pp. 1–1, 2017.
- [5] M. Ahrens, *US Experience with Smoke Alarms and Other Fire Detection/Alarm Equipment*, NFPA, Quincy, MA, USA, 2004.
- [6] J. E. Parmer, P. S. Corso, and M. F. Ballesteros, "A cost analysis of a smoke alarm installation and fire safety education program," *Journal of Safety Research*, vol. 37, no. 4, pp. 367–373, 2006.
- [7] A. H. Buchanan and A. K. Abu, *Structural Design for Fire Safety*, John Wiley & Sons, 2017.
- [8] S. R. Turns, *An Introduction to Combustion*, vol. 287, McGraw-hill, New York, NY, USA, 1996.
- [9] J. Warnatz, U. Maas, and R. W. Dibble, *Combustion*, Springer, Berlin, Germany, 2012.
- [10] J. Li, X. Zhu, N. Tang, and J. Sui, "Study on ZigBee network architecture and routing algorithm," in *Proceedings of the 2010 2nd International Conference on Signal Processing Systems (ICSPS)*, vol. 2, pp. 2–389, IEEE, 2010.
- [11] P. B. Brazdil and C. Soares, "A comparison of ranking methods for classification algorithm selection," in *Proceedings of the European conference on machine learning*, pp. 63–75, Springer, Berlin, Germany, 2000.
- [12] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [13] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.
- [14] T. M. Mitchell, *Machine Learning*, vol. 45, McGraw Hill, Burr Ridge, IL, USA, 1997.
- [15] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
- [16] A. Liaw and M. Wiener, "Classification and regression by random forest," *The R Journal*, vol. 2, no. 3, pp. 18–22, 2002.
- [17] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.
- [18] Z. Ning, X. Hu, Z. Chen et al., "A cooperative quality-aware service access system for social internet of vehicles," *IEEE Internet of Things Journal*, pp. 1–1.
- [19] X. Hu, V. C. Leung, Y. Kwok et al., "SAFeDJ: a Crowd-Cloud Co-design Approach to Situation-aware Music Delivery for Drivers," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 12, no. 1s, p. 21, 2015.
- [20] W. A. Arbaugh, *Real 802.11 security: Wi-Fi protected access and 802.11 i*, Addison-Wesley Longman Publishing Co., Inc., 2003.
- [21] *How It Works — Bluetooth Technology Website*, 2017, from <https://www.bluetooth.com/what-is-bluetooth-technology/how-it-works>.
- [22] N. A. Somani and Y. Patel, "Zigbee: A low power wireless technology for industrial applications," *International Journal of Control Theory and Computer Modelling (IJCTCM)*, vol. 2, pp. 27–33.
- [23] M. Knight, "How safe is Z-Wave? [Wireless standards]," *IET Computing and Control Engineering*, vol. 17, no. 6, pp. 18–23, 2006.

- [24] S. Tree, *Wireless sensor networks. Self, 1(R2), C0.CC2630 SimpleLink™ 6LoWPAN, ZigBee® Wireless MCU*, 2014, <http://www.ti.com/lit/ds/swrs177b/swrs177b.pdf>.
- [25] Home Smoke Alarms A Technology Roadmap. (2012). *cpsc.gov*. Retrieved 4 June 2017, from <https://www.cpsc.gov/s3fs-public/homesmokealarm.pdf>.
- [26] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [27] BMP180, "Bosch-sensortec.com," 2017, [https://www.bosch-sensortec.com/bst/products/all\\_products/bmp180](https://www.bosch-sensortec.com/bst/products/all_products/bmp180).
- [28] C. Strobl, A.-L. Boulesteix, A. Zeileis, and T. Hothorn, "Bias in random forest variable importance measures: illustrations, sources and a solution," *BMC Bioinformatics*, vol. 8, article 25, 2007.
- [29] D. Robotics, "DHT11 Humidity & Temperature Sensor," ECharts · Doc, 2017, <http://echarts.baidu.com/echarts2/doc/doc-en.html>.
- [30] J. S. Lee, Y. W. Su, and C. C. Shen, "A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi," in *Proceedings of the 33rd Annual Conference of the IEEE Industrial Electronics Society, IECON 2007*, pp. 46–51, IEEE, 2007.
- [31] "UCI Machine Learning Repository: Cardiotocography Data Set," 2017, from <http://archive.ics.uci.edu/ml/datasets/Cardiotocography>.
- [32] G. Yao, B. Zhuang, D. Zhao, and X. Huo, "Design of intelligent home system based on ZigBee wireless technology," *Modern Electronics Technique*, vol. 39, no. 22, pp. 81–84, 2016.
- [33] K. Fukunaga and P. M. Narendra, "A Branch and Bound Algorithm for Computing k-Nearest Neighbors," *IEEE Transactions on Computers*, vol. C-24, no. 7, pp. 750–753, 1975.

## Research Article

# A Research on Fast Face Feature Points Detection on Smart Mobile Devices

Xiaohe Li , Xingming Zhang, and Haoxiang Wang 

*College of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China*

Correspondence should be addressed to Haoxiang Wang; [hxwang@scut.edu.cn](mailto:hxwang@scut.edu.cn)

Received 1 September 2017; Revised 12 December 2017; Accepted 28 December 2017; Published 25 February 2018

Academic Editor: Zhaolong Ning

Copyright © 2018 Xiaohe 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.

We explore how to leverage the performance of face feature points detection on mobile terminals from 3 aspects. First, we optimize the models used in SDM algorithms via PCA and Spectrum Clustering. Second, we propose an evaluation criterion using Linear Discriminative Analysis to choose the best local feature descriptions which plays a critical role in feature points detection. Third, we take advantage of multicore architecture of mobile terminal and parallelize the optimized SDM algorithm to improve the efficiency further. The experiment observations show that our final accomplished GPC-SDM (improved Supervised Descent Method using spectrum clustering, PCA, and GPU acceleration) suppresses the memory usage, which is beneficial and efficient to meet the real-time requirements.

## 1. Introduction

Face feature point detection and tracking is a hot topic in the research field of computer vision. It is the foundational technology of face pose evaluation, face expression transfer or synchronization, and so on. With the popularity of smart mobile terminals, a real-time tracking of facial feature points in mobile terminal is highly on demand. For example, the recently designed mobile [1] or vehicle [2] social networks need high quality face feature points tracking and detection for security authentication. It may also benefit the information retrieval within mobile crowd sensing architectures [3, 4]. However, due to the limited computational capacity of floating points on mobile terminal and the battery power, a series of face feature detection algorithms, which were successful on computer, cannot be realized directly on mobile terminals. Thus we come to the problem about how to realize real-time detection and tracking of face feature points on mobile terminals with limited computation power.

Nowadays, numerous research studies of facial feature point detection have been embodied in the experimental exploration on personal computer (PC) side. According to literature [5], from the model design, the method of facial feature point detection methods can be divided into local facial feature based, global facial feature based, and the

hybrid method combining local features and global features. In comparison with other methods developed in recent years, four local feature based algorithms, such as Supervised Descent Method (SDM) [6], Coarse to Fine Shape Searching (CFSS) [7], Ensemble Regressing Trees (TREES) [8], and Explicit Shape Regression (ESR) [9], have achieved outstanding performances. Surprisingly, although these methods do not take into account the geometric features of facial feature points, they still manifest a good expression of facial occlusion. In spite of the simplicity, the speed and accuracy of SDM algorithm are still very competitive so far. And the cascade regression method firstly contributed by SDM has been continuously learned by the scholar community [10–13] until now. Besides, based on a local receptive field of artificial neurons, Baltrusaitis et al. [14] in CMU proposed a method of facial feature point detection, which performed well under different illuminations. Based on both global and local feature descriptions, Hasan et al. [15] developed a shape regression method, which made a breakthrough on the W300 test set. Jaiswal et al. [16] proposed Gabor-LBP based face feature detection using model selection and support vector regression. According to the experiment observations, this method plays a superior role in capturing the face feature points with local occlusion and background clutters. Specially, Martinez et al. [17] broke the traditional routines of

sequential regression method on this problem and proposed multivisual cues aggregation regression approach. With the continuous success of deep learning in computer vision tasks, Jourabloo and Liu [13] employed CNN as both shape and coefficients of 3D face model estimators for large pose variant face alignment and it can estimate 3D face shapes as shown in their experiments.

Recently, mobile devices have attracted tremendous attentions. Choi et al. [18] developed a face feature detection system for real-time training on mobile phones. Tresadern et al. [19] introduced the fixed-point method [20] and Haar-Like feature into the Active Shape Model [21] (ASM) algorithm to perform real-time tracking of facial feature points. Based on hierarchical model, Jian-kang et al. [22] proposed a detection method, achieving better results on mobile phones. Jiang [23] developed a facial feature detection system based on Android platform. With facial point detection on Android platform, Hp [24] invented a 3D human face pose estimation system.

Despite the numerous studies on the face feature detection algorithm for mobile terminals, it is still in its infancy, and most of them are based on mobile operating system. In this article, we improved SDM algorithm [6] with two aspects: the calculation and model. First of all, we reduced the dimension of features needed for localizing the facial feature points by using spectrum clustering. Second, we decreased the number of linear models in SDM by PCA. Finally, the local feature extraction and SDM model calculation were combined and optimized on the mobile GPU to further improve the performance of SDM algorithm. Since the SDM method is widely learned by most recently released face feature point detection methods, our modification should be also meaningful to boost the performance of other approaches.

## 2. Solution Pipeline

Figure 1 presents the schematics of collecting video data with mobile phone cameras. Face feature point detection is applied to a frame extracted from the video stream. Face detection is performed using Viola's face detector. In order to overcome the illumination problems, the preprocessing steps have been employed in this work (details are available in Section 3). The local image feature extraction algorithm is performed on the initial feature points distribution, using LBP, HOG [25], SIFT [26], and so on. The feature extraction algorithm is implemented following GPU parallel design, which is copied into shared memory according to the pixel values of face images needed by each computing unit. At the same time, the SDM algorithm model is also embedded into the shared memory according to the computing unit. The calculated results of each computational unit are the regression components of human face shape.

After obtaining the regression components, we merge the regression components and update the shape model based on the integrated regression components. The process is repeated until that the shape model satisfies the error requirements in the training process.

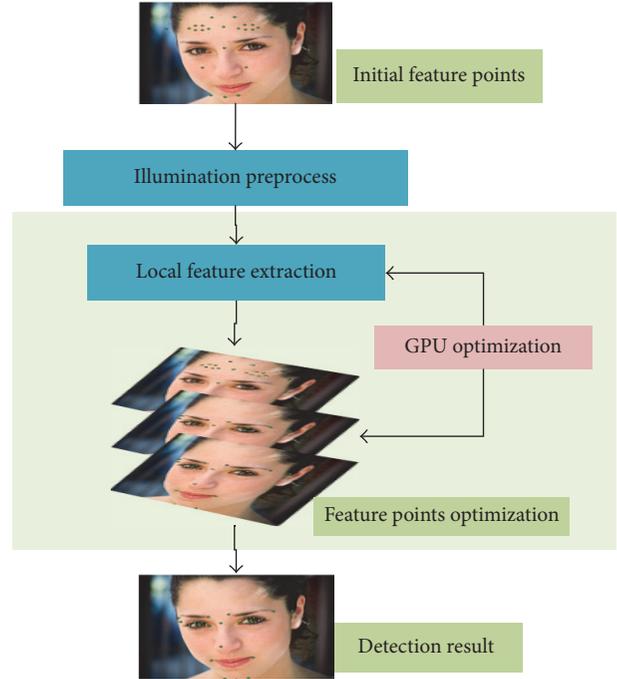


FIGURE 1: The solution pipeline.

## 3. Preprocessing

Considering the real video data acquisition, camera may have noise in dim condition, resulting in large image gradient or irregular filter responses, which severely influence the robustness of the detection algorithm. We normalized the face image from two aspects: noise reduction and brightness stretching as shown in Figure 2.

Firstly, we performed median filter to reduce the noise probably caused by the low light conditions. Then the histogram equalization is performed, in order to unify the light effect, clear the surface details of dim face image, and prepare for the subsequent steps.

## 4. Optimized SDM

*4.1. Introduction to SDM.* The core idea of SDM algorithm is to minimize the object function (1), aiming at finding the right modification of current position to the ground truth. The expression is as follows:

$$\Delta x = \arg \min_{\Delta x} \|h(d(x_0 + \Delta x) - h(d(x^*)))\|, \quad (1)$$

where  $x^*$  represents for the labeled facial feature points position;  $d$  stands for face images;  $h$  is the local description extracted at certain positions;  $x_0$  is the initial shape. According to Newton method, minimizing (1) yields

$$\Delta x = -2H_h^{-1} J_h (h(d(x^*)) - h(d(x_0))), \quad (2)$$

where  $H_h$  is the Hessian Matrix of feature extraction function denoted by  $h$ ;  $J_h$  is the Jacobian of  $h$ . Although the function is



FIGURE 2: Face image preprocessing.

ill posed since the feature extraction function  $h$  is commonly unknown in practice, it can be reformulated as

$$\Delta x = R * h(d(x_0)) + b. \quad (3)$$

Since the facial feature points are labeled at each instance,  $j = 1, 2, \dots, n$ , the difference between initial and optimal feature points  $\Delta x^j$  is also known; thus we have the following linear system:

$$\Delta x^j = R(h(d^j(x_0))) + b. \quad (4)$$

Let  $d^j$  represent the  $j$ th training image;  $\Delta x_j$  is used to solve  $R$  and  $b$  in the training phase. After the  $i-1$  regression model  $R_i$ ,  $b_i$  is obtained; we update each initial shape of each individual training samples:

$$x_i^j = x_{i-1}^j - R(R_{i-1}(h(d^j(x_{i-1}^j)))) + b_{i-1}. \quad (5)$$

Training was accomplished after obtaining the regression models:  $\{R_i, b_i\}$ ,  $i \in 1, 2, \dots, K$ .

In the research of facial feature point tracking, SDM algorithm is still applicable. The difference between the face feature points detection and the face feature points tracking is in the initial feature points distribution. In the training step of facial feature points tracking, each individual's initial facial points position  $x_0^i$  is randomly sampled once or several times from its label  $x^{*i}$  within 95% or 20 pixels rescale or translation. Then, the tracking models are obtained with the same process discussed above. It is reasonable because the face shapes in consequence frames change slowly in the video.

**4.2. Model Compression.** Through the above analysis, we can find that SDM algorithm uses multiple regression to achieve the face feature point alignment, and its computational complexity is mainly concentrated in the calculation of regression models:

$$\begin{aligned} R_t &= X(S_t)A_t + B_t, \\ S_{t+1} &= S_t + R_t. \end{aligned} \quad (6)$$

Assuming  $S_t$  is the feature points position after  $t$  times regression, considering we have 66 feature points to align, we will need  $66 * 2 = 132$  linear model at each regression model. However, with careful consideration, we can find that if two feature points  $y_i, y_j$  in shape  $S$  are not strongly correlated, the local feature extracted at  $y_i$  does not help much to locate  $y_j$ . Thus, we separate  $S$  to  $S_1, S_2, \dots, S_k$ , making feature points in the same subshape mostly correlated, and feature points in different subshapes not that relevantly. In order to accurately estimate the number of categories, we introduce spectral clustering for analysis. First, the correlation coefficients between different feature points are defined as

$$\rho_{k,j} = \frac{1}{D} \sum_d \left| \frac{E(p_{k,d} - E(p_{k,d}))E(p_{j,d} - E(p_{j,d}))}{\sigma_{k,d}\sigma_{j,d}} \right|. \quad (7)$$

Let  $n$  be the number of training samples,  $D$  be the dimension of feature points, and  $\sigma_{k,d}$  be the covariance of the  $k$ th feature point in dimension  $d$ .

We use the correlation coefficient matrix directly to define the Laplacian Matrix:

$$L = D - W. \quad (8)$$

The weight matrix  $W_{i,j} = e^{-\rho_{i,j}}$ , and  $D_{ii} = \sum_j W_{i,j}$ . According the spectrum clustering, the number of smallest Eigen values of  $L$  minus 1 is the number of catalogs denoted as  $K$ . Thus we can get  $K$  subshapes by performing  $k$ -means on the last second Eigen vector to the  $K + 1$  Eigen vectors. Assuming that face feature points can be divided into  $K$  classes, the facial feature point location of SDM algorithm can be rewritten as

$$\begin{aligned} S_k^t &= X(S_k^{t-1})R_k^T + b_k^T + S_k^{t-1}, \\ S^t &= \bigcup_k S_k^t. \end{aligned} \quad (9)$$

Equation (9) uses union of subshapes' regression model to determine the whole shape of face. The method reduces

the length of each linear model  $1/K$  times to the original and relaxes the computational complexity of SDM. However, the number of linear models in the framework of the SDM algorithm has not been reduced. Therefore, we use PCA to further simplify the SDM cascade linear model. The process is as follows:

Assuming we have the following linear model:

$$XA + B = Y, \quad Y = (y_1, y_2, \dots, y_m) \quad (10)$$

we perform dimension reduction using PCA on  $Y$  so that  $YW = \tilde{Y}$ .  $W \in R^{m \times n}$  is the principals of  $Y$  and  $n \ll m$ . We have

$$\tilde{Y}W^T = Y(WW^T). \quad (11)$$

If we denote  $W^{\text{full}} = [W, W^r]$ , this yields

$$\begin{aligned} Y - YWW^T &= YW^{\text{full}}W^{\text{full}T} - YWW^T \\ &= Y(WW^T + W^rW^{rT}) - YWW^T \\ &= YW^rW^{rT}. \end{aligned} \quad (12)$$

From matrix decomposition of  $Y^TY$ , we have

$$W^rW^{rT}Y^TYW^rW^{rT} = W^r\Lambda^RW^{rT}. \quad (13)$$

$\Lambda^R$  is the diagonal matrix with Eigen values corresponding to  $W^r$ . Because  $\Lambda^R \approx \mathbf{0}$ , we can know  $Y - YWW^T \approx \mathbf{0}$ . So (11) can be rewritten as

$$XA + B = \tilde{Y}W. \quad (14)$$

Since  $W$  is orthogonal, multiplying  $W^T$  on both sides of (14), we get

$$XAW^T + BW^T = \tilde{Y}. \quad (15)$$

It can be rewritten as

$$X\tilde{A} + \tilde{B} = \tilde{Y}. \quad (16)$$

From the analysis above, we can know that the width of each linear model can be reduced to number of principal of  $Y$ .

**4.3. GPU Acceleration.** In the design principle of SDM algorithm, we know that the robustness of local feature extraction from facial feature points has a very important impact on the effect of face alignment. In practice, we find that the feature extraction and the model computation consume half of the time cost of SDM algorithm, respectively. With the development of computer vision, scholars have proposed an endless stream of local feature descriptors. Among them, the most widely used features are HOG (Histogram of Gradients) [25], SIFT (Scale Invariant Feature Transformation) [26], and LBP (Local Binary Patterns) [27]. However, these features are either too time consuming or difficult to be parallelized. For example, HOG feature is hard to be paralleled for lack

of feature points. The reason may lie in overlapping cell. Statistical histograms of gradient directions are very easy to generate conflicts in memory accesses of computing units. While using parallel SIFT features, because of the need to obtain images pyramid, it will not save computational overhead but will lower the performance of the algorithm when the feature points' size is small. Therefore, the main work of GPU acceleration is summarized as follows: (1) picking up the suitable image feature extraction algorithm which is easy to be parallelized and robust under small scale data. (2) By taking benefits of mobile phone GPU, parallelize the feature extraction in face alignment problem and the calculation of SDM algorithm

For the judgment of local description features, we derive heuristics from the linear discrimination analysis (LDA). Given an algorithm, denoted as  $A_i$ , we expected local features extracted using  $A_i$  should have smaller covariance at the same feature point in training set  $D$ . The features extracted at different feature points should have enough difference. Apply the algorithmic idea of LDA; that is, in the same class, variance of features should be as small as possible. On the contrary, the variance between classes should be as large as possible. We denote  $L_{i,j}^k$  as the local feature extracted in the  $k_{\text{th}}$  sample's  $j_{\text{th}}$  feature points  $P_j$  using algorithm  $A_i$ . The covariance of  $P_j$ 's local feature can be expressed as

$$\text{Cov}(i, j) = \frac{\sum_k (L_{i,j}^k - m_{i,j})(L_{i,j}^k - m_{i,j})^T}{|D|}, \quad (17)$$

where  $m_{i,j}$  is the mean of all samples feature extracted at  $P_j$  using algorithm  $A_i$ ,  $m_{i,j} = \sum_k L_{i,j}^k/n$ .

Thus, the covariance of different feature points using  $A_i$  can be calculated with

$$\text{Cov}(i) = \frac{\sum_j (m_{i,j} - \bar{m}_i)(m_{i,j} - \bar{m}_i)^T}{P}, \quad (18)$$

where  $\bar{m}_i = \sum_j m_{i,j}/P$ . We believe that if the feature extraction algorithm  $A_i$  is good enough, the covariance should be small with the same feature points but distinguishable at different feature points. Thus, we want to evaluate  $A_i$  using the similar method as used in LDA. Despite the Eigen vectors, we are only concerned about the portion of the max  $k$  Eigen values:

$$\text{Score}(A_i) = \frac{\max_k \text{Eigen}(\sum \text{Cov}(i, j))^{-1} \text{Cov}(i)}{\sum \text{Eigen}(\sum \text{Cov}(i, j))^{-1} \text{Cov}(i)}. \quad (19)$$

Finally, we choose the algorithm with largest portion of largest  $k$  Eigen values:

$$A^* = \arg \max_i \text{Score}_i(A_i). \quad (20)$$

We have analyzed recent popular local feature extraction algorithms for computer vision, including SIFT, SURF [28], HOG, BRIEF [29], ORB [30], BRISK [31], Freak [32], and MRLBP [27]. And we discovered that Freak has better performance in the problem of face alignment. The experiment details will be discussed in Section 5.

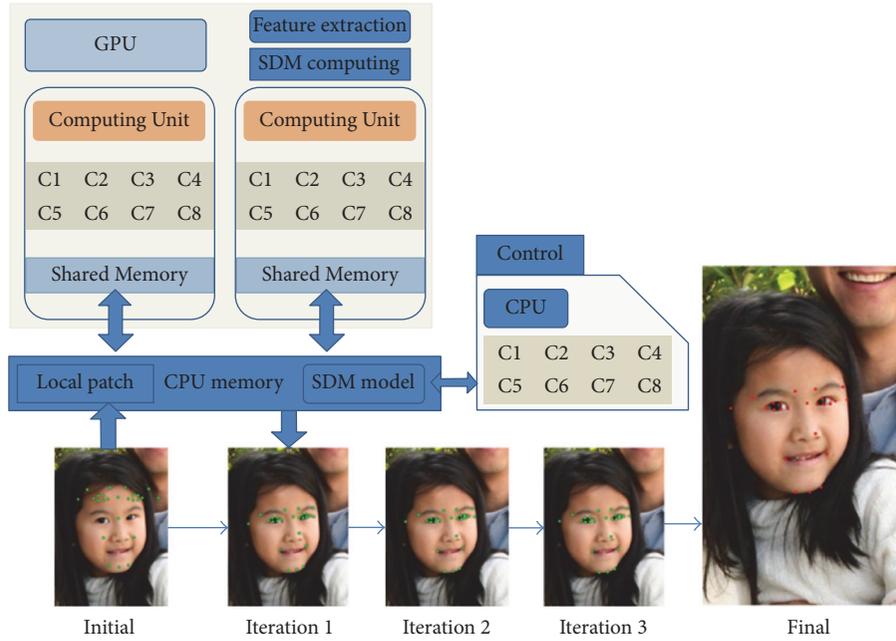


FIGURE 3: GPU parallel model for fusion feature extraction and SDM.

SDM face alignment algorithm does not have sufficient data to apply GPU acceleration. And if we do so, we will make both GPU-shared memory and CPU memory frequently visited. This will not only affect the performance of the algorithm, but also reduce memory life. Therefore, facilitating GPU architecture of mobile phone, we design a parallel SDM and feature extraction mode. As shown in Figure 3, our method parallelized features extraction and model solution of SDM. First, we load the pixel values of the subblocks at each feature point in memory according to the location of each feature point in the initial shape. Please note that if the pixel values in the subgraph block need to be accessed  $k$  times, we will store  $k$  copies of the pixel values in the shared memory. At the same time, the SDM algorithm is applied to the model fractions and image features at each patch, which is stored in the shared memory. Once the feature extraction calculation of a feature point is completed, the component of the image feature at the feature point is directly taken as the contribution to the overall shape regression.

## 5. Experiment Results & Discussion

In our experiment setup, we firstly evaluate the performance of local features as described in Section 4 with  $k = 30$ . In Table 1, we denote (Sep.) as the separable score according to (20), (p/s) as the numbers of feature points can be processed by these algorithms. To evaluate performance in mobile terminal, we use Qualcomm snapdragon 800 series platform. The dataset we used is a combination of LFW66 [33] and Helen [34].

As we can see in Table 1, the traditional local feature algorithm still has certain advantage regarding robustness or running speed. The most stable feature is still the HOG

TABLE 1: Performance comparison of each feature.

P.	A.							
	Sift	Hog	LBP	BRIEF	BRISK	ORB	FREAK	MRLBP
Sep.	0.6	0.67	0.4	0.47	0.44	0.45	0.65	0.62
p/s	47	66	200	131	147	112	150	132

feature, but the fastest algorithm is the LBP feature. Considering the stability and running speed of the feature, the fast retinal feature points (Freak) proposed in recent years has more practical value.

Consequently, we compare the performance of the original SDM algorithm with our optimization on the aspects of efficiency and robustness. In order to prove the effectiveness of each optimization step, the whole optimization procedure is divided into C-SDM: the clustering optimization based on the feature point correlation; PC-SDM: the dimensional reduction optimization based on principal component analysis and feature point clustering; GPC-SDM: the face alignment algorithm which is finally proposed in this paper with adding dynamic GPU parallel optimization. First, we compare the optimization algorithms of each step with the original SDM. In order to quantify the problem, we first define the accuracy of the face regression algorithm with

$$p(I_k) = \frac{\sum_p \|\hat{y}_k^p - y_k^p\|}{P \|\gamma_k(l) - \gamma_k(r)\|}, \quad (21)$$

where  $I_k$  is the  $k_{th}$  image in testing set and  $\hat{y}$  represents the estimated shape. The denominator is the distance between left

TABLE 2: Performance comparison with state-of-the-art algorithms.

	SDM-SVR	SDM(L1)	PC-SDM	GPC-SDM	EST	F-LBF	F-cGPRT
Fps.	4.5	47.5	27.5	48.6	100	150	220
Success.	0.92	0.85	0.915	0.906	0.72	0.75	0.78

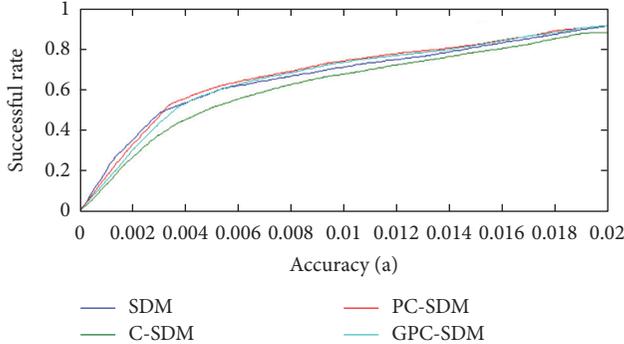


FIGURE 4: The plot of success rates.

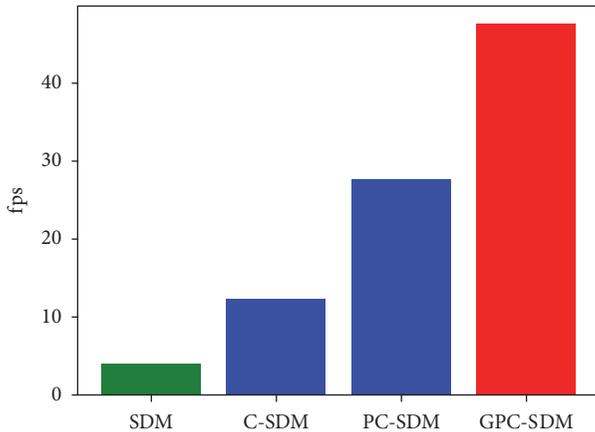


FIGURE 5: The tested fps of each optimization step.

and right eye, which is used to normalize the estimation error. We define the successful score of certain precision as follows:

$$\text{success}(\alpha) = \frac{\sum_k \delta(p(I_k) \leq \alpha)}{C}, \quad (22)$$

where  $\delta$  is the indicator pointing out whether the testing sample's estimation error is lower than  $\alpha$  and  $C$  is the number of testing samples. The successful rate of different optimization with growing  $\alpha$  is shown as Figure 4.

Combined with Figures 4 and 5, we can see that each phase of the optimization has significantly improved against the previous phase. Compared with the original SDM algorithm, C-SDM algorithm has a slight decline in the success rate. PC-SDM algorithm, integrating PCA and C-SDM, can enhance the performance further without affecting the success rate. Though the GPU-based GPC-SDM optimization should have the same success rate curve with the PC-SDM algorithm, due to the excessive GPU floating-point truncation error, the success rate decreased slightly in high

precision conditions. For all the algorithms, a relaxation of the accuracy condition can lead to the same success rate as the SDM algorithm. In a word, through comparison experiment, we can see that the optimization scheme proposed in this paper can achieve significant improvement in performance.

Finally, we compare our last two modifications with the state-of-the-art face feature points detection methods with speed advantage. The experiments were cross datasets designed to prevent overfitting phenomenon; that is, if the model was trained on one dataset, saying the Helen dataset, it must be evaluated with other datasets, say the LFW. Thus, the precisions may slightly different from the publications of the original works. Also, the fps decreases dramatically since the experiment environment moves to mobile terminal. To simplify the experiments, we choose  $\alpha = 0.02$  and compare both fps and success rate with state-of-the-art algorithms: the ensemble regression trees (EST) [9], the Gaussian process regression trees (F-cGPRT) [10], the local binary feature regression (F-LBF) [11], and the L1 norm penalty SDM (SDM(L1)) [12].

As shown in Table 2, the performance of PC-SDM and GPC-SDM boost the speed of face feature points detection and preserve the successful rate of the most robust SDM algorithm. Other state-of-the-art algorithms used regression trees or forest and simple image features to improve the speed. However, these approaches do not break through the computation framework proposed by SDM. And nowadays regression trees algorithms are easy to overfit the training set. This can be easily observed in Table 2 where although their approaches were announced to be precise, the successful rate downgraded with cross datasets evaluation. One successful modification of SDM is SDM(L1) [12], which employs sparsity with the SDM models and can speed up the algorithm easily and significantly. However, it also sacrifices the success rate compare to original SDM algorithm with support vector regression. Thus, although our proposal did not achieve the fastest performance even with GPU acceleration, it has more practical value considering both robustness and real-time requirements.

## 6. Conclusion

In this paper, we transplant the SDM algorithm on mobile devices with the following work: first, based on the correlation analysis between the feature points on the statistics of face shapes, we introduce spectral cluster and PCA to reduce the model size of SDM. Second, a local feature evaluation method based on linear discrimination classification is proposed, which decouples feature evaluation and face alignment algorithm to facilitate the experimental work. Third, a GPU-based SDM model is proposed, which combines feature extraction and model solving. Finally, the

potential of SDM algorithm is fully exploited, and the SDM algorithm is optimized and transplanted to mobile terminal.

Face alignment algorithm has broad application prospects, but the research of this problem is still in its initial stage. We still need to improve the success rate of the face alignment algorithm, especially in the presence of occlusion, light changes, night light, and face 3D pose changes. The algorithm of face alignment and facial feature point tracking success rate is still insufficient. Studying how to overcome these problems is important in our next research work.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] Z. Ning, X. Hu, Z. Chen et al., "A cooperative quality-aware service access system for social internet of vehicles," *IEEE Internet of Things Journal*, pp. 1-1, 2017.
- [2] X. Hu, T. H. S. Chu, V. C. M. Leung, E. C.-H. Ngai, P. Kruchten, and H. C. B. Chan, *A Survey on Mobile Social Networks: Applications, Platforms, System Architectures, and Future Research Directions*, *IEEE Communications Surveys Tutorials*, vol. 17, A Survey on Mobile Social Networks, Applications, 2015.
- [3] X. Hu, T. H. S. Chu, H. C. B. Chan, and V. C. M. Leung, "Vita: a crowdsensing-oriented mobile cyber-physical system," *IEEE Transactions on Emerging Topics in Computing*, vol. 1, no. 1, pp. 148-165, 2013.
- [4] X. Hu, X. Li, E. C.-H. Ngai, V. C. M. Leung, and P. Kruchten, "Multidimensional context-aware social network architecture for mobile crowdsensing," *IEEE Communications Magazine*, vol. 52, no. 6, pp. 78-87, 2014.
- [5] Y. Heng et al., *An Empirical Study of Recent Face Alignment Methods*, Computer Science, 2015.
- [6] X. Xiong and F. De la Torre, "Global supervised descent method," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '15)*, pp. 2664-2673, IEEE, Boston, Mass, USA, June 2015.
- [7] S. Zhu, C. Li, C. C. Loy, and X. Tang, "Face alignment by coarse-to-fine shape searching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 4998-5006, USA, June 2015.
- [8] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 1867-1874, Columbus, Ohio, USA, June 2014.
- [9] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by Explicit Shape Regression," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2012*, pp. 2887-2894, USA, June 2012.
- [10] D. Lee, H. Park, and C. D. Yoo, "Face alignment using cascade Gaussian process regression trees," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 4204-4212, USA, June 2015.
- [11] S. Ren, X. Cao, Y. Wei, and J. Sun, "Face alignment at 3000 FPS via regressing local binary features," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014*, pp. 1685-1692, usa, June 2014.
- [12] Q. Liu, J. Deng, and D. Tao, "Dual sparse constrained cascade regression for robust face alignment," *IEEE Transactions on Image Processing*, vol. 25, no. 2, pp. 700-712, 2016.
- [13] A. Jourabloo and X. Liu, "Large-pose face alignment via CNN-based dense 3D model fitting," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 4188-4196, usa, July 2016.
- [14] T. Baltrusaitis, P. Robinson, and L.-P. Morency, "Constrained local neural fields for robust facial landmark detection in the wild," in *Proceedings of the 14th IEEE International Conference on Computer Vision Workshops (ICCVW '13)*, pp. 354-361, Sydney, Australia, December 2013.
- [15] M. K. Hasan, C. Pal, and S. Moalem, "Localizing facial keypoints with global descriptor search, neighbour alignment and locally linear models," in *Proceedings of the 2013 14th IEEE International Conference on Computer Vision Workshops, ICCVW 2013*, pp. 362-369, aus, December 2013.
- [16] S. Jaiswal, T. R. Almaev, and M. F. Valstar, "Guided unsupervised learning of mode specific models for facial point detection in the wild," in *Proceedings of the 2013 14th IEEE International Conference on Computer Vision Workshops, ICCVW 2013*, pp. 370-377, aus, December 2013.
- [17] B. Martinez, M. F. Valstar, X. Binefa, and M. Pantic, "Local evidence aggregation for regression-based facial point detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 5, pp. 1149-1163, 2013.
- [18] K. Choi, K.-A. Toh, and H. Byun, "Realtime training on mobile devices for face recognition applications," *Pattern Recognition*, vol. 44, no. 2, pp. 386-400, 2011.
- [19] P. A. Tresadern, M. C. Ionita, and T. F. Cootes, "Real-time facial feature tracking on a mobile device," *International Journal of Computer Vision*, vol. 96, no. 3, pp. 280-289, 2012.
- [20] W. Herman, *Fix-Point Method*, Encyclopedia of statistical sciences, 1983.
- [21] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models—their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38-59, 1995.
- [22] D. Jian-kang, W. Can-tian, and L. Qing-shan, "Fast Face Alignment Method Based on Hierarchical Model and it's Application on Mobile Device," *Computer Science*, vol. 12, p. 65, 2014.
- [23] H. jiang, *The research and implement of face feature points localization based on android platform*, Beijing Jiaotong University, 2013.
- [24] W. L. W. Hp, "Real time face pose estimation based on 3D model implemented in android," *International Journal of Computer Applications*, 2015.
- [25] T. B. N. Dalal, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 1, pp. 886-893, June 2005.
- [26] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.
- [27] L. Liu, S. Lao, P. W. Fieguth, Y. Guo, X. Wang et al., "Median robust extended local binary pattern for texture classification," *IEEE Transactions on Image Processing*, vol. 25, no. 3, pp. 1368-1381, 2016.
- [28] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: speeded up robust features," in *Proceedings of the in European Conference on Computer Vision*, 2006.

- [29] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: binary robust independent elementary features," in *Computer Vision—ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, Part IV*, vol. 6314 of *Lecture Notes in Computer Science*, pp. 778–792, Springer, Berlin, Germany, 2010.
- [30] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 2564–2571, Barcelona, Spain, November 2011.
- [31] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: binary robust invariant scalable keypoints," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 2548–2555, Barcelona, Spain, November 2011.
- [32] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: fast retina keypoint," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 510–517, June 2012.
- [33] B. Huang Gary et al., *Labeled Faces in the Wild: A Database for Studying face recognition in unconstrained environment*, University of Massachusetts, Amherst, MA, USA, 2007.
- [34] Le. Vuong et al., "Interactive Facial Feature Localization," in *Proceedings of the European Conference on Computer Vision*, 2012.

## Research Article

# Congestion Control and Traffic Scheduling for Collaborative Crowdsourcing in SDN Enabled Mobile Wireless Networks

Dawei Shen,<sup>1</sup> Wei Yan,<sup>1</sup> Yuhuai Peng ,<sup>1,2</sup> Yanhua Fu,<sup>3</sup> and Qingxu Deng <sup>1,2,4</sup>

<sup>1</sup>College of Computer Science and Engineering, Northeastern University, Shenyang 110819, China

<sup>2</sup>Key Laboratory of Vibration and Control of Aero-Propulsion System of Ministry of Education, Northeastern University, Shenyang 110819, China

<sup>3</sup>College of Jangho Architecture, Northeastern University, Shenyang 110819, China

<sup>4</sup>State Key Laboratory of Rolling and Automation, Northeastern University, Shenyang 110819, China

Correspondence should be addressed to Yuhuai Peng; pengyuhuai@mail.neu.edu.cn and Qingxu Deng; dengqx@mail.neu.edu.cn

Received 8 September 2017; Accepted 3 January 2018; Published 21 February 2018

Academic Editor: Kuan Zhang

Copyright © 2018 Dawei Shen 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.

Currently, a number of crowdsourcing-based mobile applications have been implemented in mobile networks and Internet of Things (IoT), targeted at real-time services and recommendation. The frequent information exchanges and data transmissions in collaborative crowdsourcing are heavily injected into the current communication networks, which poses great challenges for Mobile Wireless Networks (MWN). This paper focuses on the traffic scheduling and load balancing problem in software-defined MWN and designs a hybrid routing forwarding scheme as well as a congestion control algorithm to achieve the feasible solution. The traffic scheduling algorithm first sorts the tasks in an ascending order depending on the amount of tasks and then solves it using a greedy scheme. In the proposed congestion control scheme, the traffic assignment is first transformed into a multiknapsack problem, and then the Artificial Fish Swarm Algorithm (AFSA) is utilized to solve this problem. Numerical results on practical network topology reveal that, compared with the traditional schemes, the proposed congestion control and traffic scheduling schemes can achieve load balancing, reduce the probability of network congestion, and improve the network throughput.

## 1. Introduction

In recent years, crowdsourcing have received extensive attention from industry and academia, which was originally proposed by American journalist Jeff Howe in 2006. Crowdsourcing means that tasks performed by employees in a company or institution before will be outsourced to the unspecific public networks in a free and voluntary form. Tasks of crowdsourcing are usually undertaken by the individual. But if the people involved need to collaborate to complete the task, there may appear in form of individual production dependent on open source. Many tasks cannot be achieved through a simple algorithm, such as image labeling, commodity evaluation, and entity recognition. These kinds of problems are difficult for machines to handle but can be done with crowdsourcing. In crowdsourcing it publishes tasks directly to the Internet and gathers unknown people on the Internet to solve

problems that are difficult to deal with by traditional computers alone, such as Wikipedia, reCAPTCHA [1], tagged images, and language translations. According to the different forms of public participation in crowdsourcing, it can be divided into collaborative crowdsourcing and crowdsourcing contest. In collaborative crowdsourcing, tasks require collaboration between the masses, but people who perform tasks usually do not have rewards. Crowdsourcing can effectively solve machine-hard tasks by leveraging machine and a large group of people on the web.

Software-Defined Networking (SDN) refers to a new network architecture developed from OpenFlow technology [2]. SDN technology can be programmed by software to control the data forwarding and ultimately achieve the purpose of free data transfer control. SDN technology has outstanding advantages in flow control; therefore, we hope to use SDN technology to solve the problem of congestion control and

traffic scheduling [3] in crowdsourcing-based Mobile Wireless Networks (MWN).

Currently, a number of crowdsourcing-based mobile applications have been applied in mobile networks and Internet of Things (IoT), targeted at real-time services and recommendation, for example, Uber, Elance, Amazon, and Airbnb. These frequent information exchanges and data transmissions are heavily injected into the current communication networks [4], which poses great challenges for congestion control and traffic scheduling problem [5] in Mobile Wireless Networks. To solve the emerging challenges, this paper focuses on the traffic scheduling and load balancing problem in software-defined Mobile Wireless Networks for collaborative crowdsourcing. This paper first presents a network model towards traffic engineering problem and then designs a hybrid routing forwarding scheme as well as a congestion control algorithm to achieve the feasible solution. To validate the performance of the proposals, a lot of simulation experiments are carried out.

The rest of this paper is organized as follows. Related work in recent years is reviewed in Section 2. The network model is then formulated in Section 3. In Section 4, design of congestion control and traffic scheduling scheme are presented in detail. Simulation results and analysis are discussed in Section 5. Finally, conclusions are given in Section 6.

## 2. Related Work

At present, some researchers have summarized the research work of crowdsourcing from different perspectives.

Yuen et al. in [6] summarized the progress of crowdsourcing from applications, algorithms, performance, and data sets. Kittur et al. in [7] explained the challenges of crowdsourcing in 12 aspects such as synchronous collaboration, real-time response, and dynamic machines. Doan et al. in [8] reviewed the crowdsourcing system applied on the world wide web and summarized the crowdsourcing system according to the problem type and the way of collaboration. Zhao and Zhu in [9] reviewed crowdsourcing research from four perspectives: information, technology, the public, and organization. Kittur et al. in [10] have studied how to decompose complex tasks and how to integrate workers' answers to perform initial tasks and proposed a MapReduce framework to achieve the decomposition of tasks. However, their method is only suitable for specific types of tasks, and the general effect is unsatisfying. Scalability still needs to be solved. References [11–13] focus on the technology of combining machine and human with the join operation in crowdsourcing environment, which first filters the problem through the machine algorithm and then assigns the remaining problems to the workers. The authors of [12] used the transitive relationships of entities to further reduce the number of tasks, thereby saving the cost. Lofi et al. in [14] reduced the cost of the task by preprocessing data sets containing missing data through the "error model" and getting the answers from workers. Sakamoto et al. in [15] studied the ways in which crowdsourcing participants often interact in different task types. Heer et al. in [16] studied how to carry out a survey and found

that the design interface was more suitable for crowdsourcing workers through questionnaires. The authors in [17] proposed a method based on random map generation and messaging task allocation. The limitation of this method is that it can only be used for a specific type of task to the difficulty of the task. However, there are various types of task crowdsourcing platform, and some tasks need special professional knowledge, such as language translation task. Liu et al. in [18] implemented a data analysis system to ensure the quality of the results as the main goal, first through forecasting model number assigned tasks and, then, in the process of task execution, through online quality assessment results to determine whether to terminate the task ahead of time, thus saving cost and time. The authors in [19] proposed a new workers' model in crowdsourcing. Through this model, the workers' quality can be computed accurately and timely. For big data tasks, the number of tasks affects the overall cost of the tasks. The number of tasks can be reduced by effectively designing the task, thus saving the task cost. Marcus et al. in [20] proposed the strategy to transform the problem of each task into multiple subproblems. But when a task contains a large number of subproblems, the price of task needs to improve. Otherwise, it will be easy to cause only a small number of workers selected task. That is to say, even though such an approach reduces the number of tasks, the overall cost of the task is not guaranteed to be reduced. The authors of [21] presented a comprehensive system model of Crowdlet that defines the task, worker arrival, and worker ability models. In [22], the authors designed an approximate task allocation algorithm that is near optimal with polynomial-time complexity and used it as a building block to construct the whole randomized auction mechanism. Compared with deterministic auction mechanisms, the proposed randomized auction mechanism increases the diversity in contributing users for a given sensing job. The authors of [23] presented a new participant recruitment strategy for vehicle-based crowdsourcing. This strategy guarantees that the system can perform well using the currently recruited participants for a period of time in the future. The authors in [24] focused on a more realistic scenario where users arrive one by one online in a random order. The authors in [25] focused on the problem of how to efficiently distribute a crowdsourcing task and recruit participants based on D2D communications. In [26], existing definitions of crowdsourcing were analyzed to extract common elements and to establish the basic characteristics of any crowdsourcing initiative. Based on these existing definitions, an exhaustive and consistent definition for crowdsourcing is presented and contrasted in eleven cases. In [27], the authors defined traffic engineering as a large-scale network project to solve the performance evaluation and network optimization in the network. In [28], traffic engineering has been further explained, and the traffic engineering is a route optimization method to improve the quality of network service by avoiding the link congestion in the network.

## 3. Network Model

There are a number of possible next hops that may occur after the crowdsourcing task has selected the assignment object in

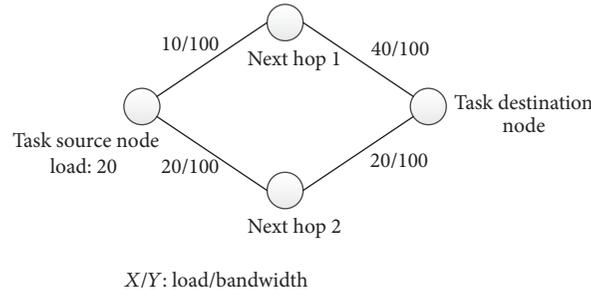


FIGURE 1: The illustration of the next hop selection in our network model.

the mobile wireless network, and different next hop options affect the load balancing in the network. As shown in Figure 1, if the crowdsourcing task source node in the figure forwards the mission to the destination node through the next hop 1, the maximum link utilization rate in the network is 0.6. If the next hop 2 is chosen to forward the task assignment, then the maximum link utilization in the network is 0.4. Therefore, the SDN controller in the network needs to calculate the next hop periodically to achieve the load balancing in the network.

Taking into account the fact that the routing network already exists in the current mobile wireless network, it requires a lot of manpower and resources to replace all the wireless nodes for the SDN node [29]. Therefore, we consider the SDN node in the mobile wireless network part of the configuration of the scenario. We assume the nodes in the mobile wireless network run the OSPF protocol, so the SDN controller can collect the load information of the links in the network. And the SDN nodes can obtain the link utilization rate of all the links in the network. When the crowdsourcing task leaves the SDN node, it may pass through other SDN nodes on its forwarding path. These nodes can also have multiple next hops, as shown in Figure 2, where the yellow nodes represent SDN nodes, while the white nodes represent non-SDN nodes. In addition, the solid line represents the forwarding path, and the dotted line represents the possible forwarding paths. It is assumed that the current task node 1 is an SDN node, and it selects node 2 as its next hop. There is also another SDN node 4 on the forwarding path, and the next hop of node 4 may be node 5 or node 6. Through the coordination of multiple SDN nodes distributed in the Mobile Wireless Networks, we can have multiple possible forwarding paths to carry the crowdsourcing tasks to achieve load balancing for global networks.

Therefore, we first need to find out all the possible paths that the package task forwards. We use the tree structure to build all possible forwarding paths [30]. First, we construct the source node of the task as the root of the tree. Each node in the tree can be divided into SDN nodes and non-SDN nodes. If it is an SDN node, then it can have multiple child nodes; otherwise, it only has a child node. We assume that when the package task is forwarded, each node in the network will inject an identity packet of the current node. When passing through the SDN node, we check this identity packet and remove the branch path containing the nodes that already exist in the current identity packet, ensuring that the

loopback is not generated when the packet task is forwarded. In Figure 2, for example, the tree structure of all possible forwarding paths can be constructed as shown in Figure 3.

In what we described above, there is only one crowdsourcing task in the wireless sensor network, but, in reality, there can be multiple crowdsourcing task in the network [31].

Formally, the wireless sensor network can be modeled as  $G(V, E)$ , with the node set  $V(1, 2, \dots, v)$  and link set  $E(e_{12}, e_{13}, \dots, e_{ij})$ .

Assume that there is no interference between nodes and links. Suppose  $T$  is the crowdsourcing tasks matrix, and the task set is  $N(1_2^1, 2_2^1, \dots, n_d^s)$  ( $s$  is the crowdsourcing task source node and  $d$  is the crowdsourcing task destination node). And the amount of task is  $L(n_d^s)$ . Define  $C(e)$  as the link capacity. Define link utilization as  $U(e)$ , which can be formulated as in

$$U(e) = \frac{\sum_e L(n_d^s)}{C(e)}. \quad (1)$$

Define that when a crowdsourcing task passes through a node, all possible forwarding paths are added to set  $P_v^{n_d^s}$ . There are two scenarios in the mobile wireless network: when a number of crowdsourcing tasks pass through non-SDN nodes, they can only be forwarded in accordance with the OSPF protocol to its next hop. And when multiple crowdsourcing tasks pass through SDN nodes, we have multiple possible forwarding paths. In the mobile wireless network, we can only control SDN nodes. Therefore, when the crowdsourcing task traffic passes through the SDN node, the problem we need to solve is as follows: given  $G, N, C(e)$  and  $P$ , how we schedule the task  $N$  over the path  $P$  with the path capacity  $C(e)$  to minimize the maximum link utilization  $U$ , then achieving load balancing. We describe it as problem  $S(N, L(n_d^s), C(e), P, U)$ .

Given the definitions above, the problem can be formalized as follows:

$$\text{Minimize } U \quad (2)$$

$$\text{Subject to: } \sum_e L(n_d^s) \leq U \cdot C(e) \quad \forall e \in E \quad (3)$$

$$P^{L(n_d^s)} \geq 0 \quad \forall P \quad (4)$$

$$L(n_d^s) \geq 0 \quad \forall n_d^s \in N. \quad (5)$$

Formula (3) indicates that the size of the task on any link is less than or equal to the maximum link utilization in the

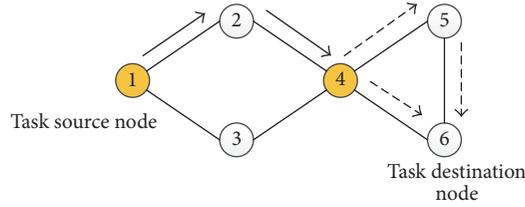


FIGURE 2: Multipath selection illustration.

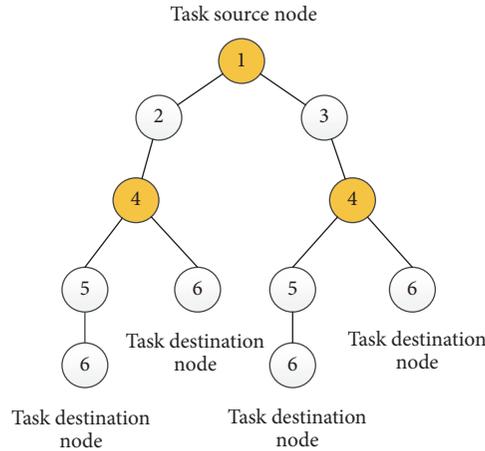


FIGURE 3: All possible path tree diagram.

network multiplied by the link capacity. Formula (4) indicates that the amount of task on any forwarding path should be nonnegative. Formula (5) indicates that task should be non-negative.

#### 4. Congestion Control and Traffic Scheduling Schemes

*4.1. Design of Hybrid Routing and Forwarding Algorithm.* In our model, we divided the nodes in the mobile wireless network into two categories: SDN nodes and non-SDN nodes. When the crowdsourcing task traffic passes through the non-SDN node, we use OSPF protocol to perform the next hop routing. When the crowdsourcing task traffic passes through the SDN node, we describe this as problem  $S(N, L(n_d^s), C(e), P, U)$ . There is a special case of problem  $S$ , where  $P = 1$  and  $U = 1$ . The problem  $S(N, L(n_d^s), C(e), 1, 1)$  is NP, and we can reduce the well-known 0-1 knapsack problem [32] to this problem. Therefore,  $S(N, L(n_d^s), C(e), 1, 1)$  is NP-hard. Thus, the more general problem  $S(N, L(n_d^s), C(e), P, U)$  is also NP-hard. This means the computation cannot be completed in a reasonable time for large networks. Therefore, we develop a heuristic algorithm for this problem with polynomial-time complexity.

On the algorithm we make the following assumptions:

- (1) SDN control center can be aware of the relevant information in the network correctly and timely.
- (2) Network topology is stable in a short time, and we do not consider the interference of wireless networks.

- (3) All the nodes are running standard OSPF protocol nodes in the mobile wireless network in addition to SDN nodes.
- (4) Mobile wireless network has only one SDN controller.
- (5) In the process of routing, SDN nodes select only one path to forward when processing a crowdsourcing task flow.
- (6) The task flow is forwarded hop by hop.

In this case, we assume that none of the links in the network will be congested, and there will not be a number of crowdsourcing task traffic on a link exceeding the capacity of the link. Therefore, when the SDN node forwards the crowdsourcing task, we can sort the crowdsourcing tasks according to the task load. Then, according to the greedy algorithm, the crowdsourcing task is distributed to the corresponding link, which makes the value of maximum link utilization in the network minimum.

The hybrid routing and forwarding algorithm is given in Algorithm 1.

Since we define the utilization of the link as the ratio of the link capacity of the data flow on the current link, if the data flow is far greater than our link capacity, our link utilization will be greater than 1. So the network's maximum link utilization is greater than 1, which is contrary to the idea of load balancing in traffic engineering. Therefore, our crowdsourcing task traffic matrix cannot be generated arbitrarily; as

Algorithm for hybrid routing and forwarding

- (1) **Begin**
- (2) **Input:** mobile wireless network topology graph  $G(V, E)$ , crowdsourcing task flow matrix  $T$ ;
- (3) **for** each row in  $T$  **do**
- (4)   **If**  $v$  is non-SDN node **then**
- (5)     Assign the task flow to its next hop forwarding link;
- (6)   **repeat**
- (7)      $\forall v \in V, v++$ ;
- (8)   **until** all non-SDN nodes are traversed;
- (9)   **If**  $v$  is SDN node **then**
- (10)     Sort the task in ascending order according to the load of the task flow;
- (11)     Compute all possible forwarding path  $P$ ;
- (12)     Use the greedy algorithm to assign task to its next hop forwarding link;
- (13)   **repeat**
- (14)      $\forall v \in V, v++$ ;
- (15)   **until** all SDN nodes are traversed;
- (16) Compute link utilization on all links in the network. Get the maximum link utilization  $U$ ;
- (17) Update the crowdsourcing task traffic matrix  $T$  to  $T'$ ;
- (18) **If**  $U' \geq U$  **then**
- (19)    $U \leftarrow U'$ ;
- (20) Return to the third step;
- (21) **Output:** the maximum link utilization  $U$ ;
- (22) **End.**

ALGORITHM 1: Hybrid routing and forwarding algorithm.

for the task flow size, according to the method described in literature [33], we generate the formula as follows:

$$d_{ij} = \sigma_i \sum_{\{t|(i,t) \in E\}} c_{(i,t)} \frac{\sum_{\{t|(t,j) \in E\}} c_{(t,j)}}{\sum_{\{(m,n)|(m,n) \in E\}} c_{(m,n)} - \sum_{\{t|(i,t) \in E\}} c_{(i,t)}} \quad (6)$$

$i, j \in V.$

In formula (6),  $d_{ij}$  represents the size of the traffic flow from the source node  $i$  to the destination node  $j$ ,  $\sigma_i$  represents a random number in an interval  $[0, 1]$ ,  $c(i, t)$  represents the link capacity between the source node  $i$  and its neighboring node  $t$ ,  $c(t, j)$  is the link capacity between destination node  $j$  and its neighboring node  $t$ , and  $c(m, n)$  represents the capacity on the link  $(m, n)$ . We generate 40 sets of crowdsourcing task flow matrices as simulation data according to formula (6). According to the above conditions, we have simulated the proposed algorithm.

**4.2. Design of Congestion Control Algorithm.** As mentioned above, we assume that there will be no congestion in the mobile wireless network, but in fact congestion is inevitable in the process of mass crowdsourcing. Therefore the problem  $S(N, L(n_d^s), C(e), P, U)$  should be  $S(N, L(n_d^s), C(e), P_e, 1)$ , because the maximum utilization of the link is 1 and  $P_e$  is the first link of the possible path  $P$ . In this case, when an SDN node is forwarding the crowdsourcing task, it needs to

select a subset of its task set  $N \{1, 2, \dots, n\}$  first. Then, these subtasks will be assigned to the possible forwarding link  $P_e$ , with the maximum value of assigned tasks under the limitation of each link. It is a multiknapsack problem. Multiple Knapsack Problem (MKP) refers to the selection of a subset of items in an item collection  $N \{1, 2, \dots, n\}$  to be loaded into  $M \{1, 2, \dots, m\}$  backpack. The purpose is to maximize the total value of selected items, with the total capacity not exceeding the volume of each backpack. Here we use the AFSA algorithm in [34] to solve this problem. Artificial Fish Swarm Algorithm (AFSA) is a new intelligent optimization algorithm for biomimetic group. Artificial fish can make AFSA better intelligent and suitable for solving large-scale complex optimization problems. We assign the crowdsourcing tasks as many as possible to the link without exceeding the link capacity. According to this heuristic rule, if we want to assign the task  $i$  to the link  $j$ , there are two possibilities. One is the link capacity  $C(j) < L(i)$ , and we cannot assign the task to the link. The other one is the link capacity  $C(j) \geq L(i)$ . Let  $C_r(e)$  represent the remaining capacity of the link  $e$ . There are two conditions: (1)  $C_r(j) \geq L(i)$ , if task  $i$  is never assigned to any link, then task  $i$  is assigned to the link  $j$ , and  $C_r(j) = C_r(j) - L(i)$ ; if task  $i$  was assigned to link  $k$  ( $k \neq j$ ), we firstly execute TakeOut( $i, k$ ) (TakeOut( $i, k$ ) which means taking the task  $i$  out of link  $k$ , and then  $C_r(k) = C_r(k) + L(i)$ ). Then we assign the task  $i$  to the link  $j$ , and the remaining capacity of the link  $j$  decreases  $L(i)$ . (2)  $C_r(j) < L(i)$ , we

```

Algorithm for Congestion control
(1) Begin
(2) Input: mobile wireless network topology graph  $G(V, E)$ , crowdsourcing task flow matrix  $T$ .
(3) for each row in  $T$  do
(4)   If  $v$  is non-SDN node then
(5)     Assign the task flow to its next hop forwarding link;
(6)   repeat
(7)      $\forall v \in V, v++$ ;
(8)   until all non-SDN nodes are traversed;
(9)   If  $v$  is SDN node then
(10)    Compute all possible forwarding path  $P_e$ ;
(11)    Compute the link capacity  $C_r(P_e)$ ;
(12)    Use the AFSA algorithm to assign task flow to its next hop forwarding link;
(13)  repeat
(14)     $\forall v \in V, v++$ ;
(15)  until all SDN nodes are traversed;
(16) Update the crowdsourcing task traffic matrix  $T$  to  $T'$ ,  $T'_m = T_m + 1$ . Return to the third step.
(17) Output: Number of times:  $T_m$ .
(18) End.

```

ALGORITHM 2: Congestion control algorithm.

execute  $\text{TakeOut}(p, j)$  ( $p$  is any task that is assigned to the link  $j$ ) until  $C_r(j) \geq L(i)$ , and then we execute (1). The artificial fish is always kept in a feasible solution and close to the bound boundary. The effective optimization of artificial fish under the guidance of behavior strategy was carried out by artificial fish feeding, rear-ending, and clustering.

Since we can only control SDN nodes in the network, we will take the traffic of non-SDN nodes in the forwarding link first. The remaining capacity of the link is the backpacking capacity of our multibackpack problem. We also need to assume that the crowdsourcing task flow cannot be split. Assume that when the number of tasks on a link exceeds the link capacity of the link, it causes the task to be discarded and needs to be reposted. Define  $T_m$  as the number of times that the crowdsourcing task has been forwarded. Finally, we evaluate our congestion control algorithm by calculating the link throughput. We use formula (7) to compute the throughput of the network:

$$\text{Throughput} = \frac{\sum L(n)}{T_m}. \quad (7)$$

The congestion control algorithm is given in Algorithm 2.

## 5. Number Results and Analysis

We mainly use VS2010 to complete the simulation, which is coded in C/C++. We use the wireless network standard based on IEEE 802.11b [35] to build our mobile wireless network with a maximum bandwidth of 11 Mbps, which means the maximum link capacity can be set to 11 M. Here we use

the method described in [36] to set the link capacity in mobile wireless network. First, divide all the nodes into two categories according to the degree of each node, A class node represents those nodes whose degree is less than 3, and B class represents the set of other degrees of nodes. If a link has two nodes in the B class node set, then the link capacity is 11 M; if there is a node in the link in the A class node, set the link capacity of 6 M.

Simulation of the mobile wireless network topologies are shown in Figures 4 and 5, where yellow nodes represent SDN nodes and white nodes represent non-SDN nodes, and we simulate the experiment by increasing the number of SDN nodes gradually.

For the hybrid routing and forwarding algorithm, we compare the network with no SDN nodes by increasing the amount of SDN nodes in the network, which is the network that we assume all nodes are forwarded according to the OSPF protocol. From (a) to (d) compare the maximum link utilization between our proposed hybrid routing forwarding scheme and the OSPF protocol by increasing the number of SDN nodes. The simulation results are shown in Figures 6 and 7.

Figures 6 and 7 present analyses of the maximum utilization with different SDN nodes deployment in Topology 1 and Topology 2. The simulation results are shown in Figures 6 and 7, the  $y$ -axis represents the maximum link utilization, and the  $x$ -axis represents the number of crowdsourcing task flow matrixes. We can see intuitively that with the increase in SDN nodes the overall trend of maximum link utilization is decreasing in the mobile wireless network from the simulation results in Figures 6 and 7. However, it can be seen in the

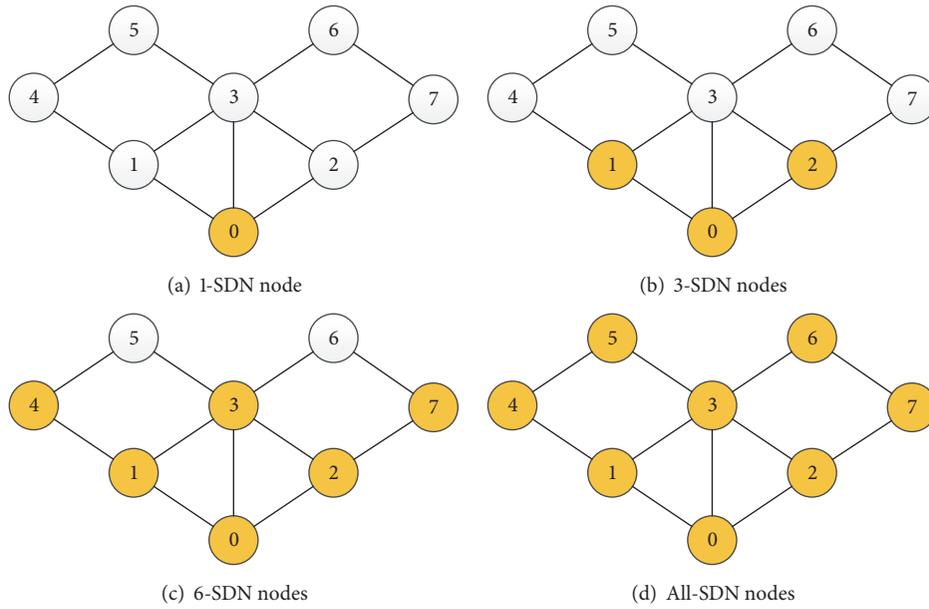


FIGURE 4: Mobile wireless network topology 1.

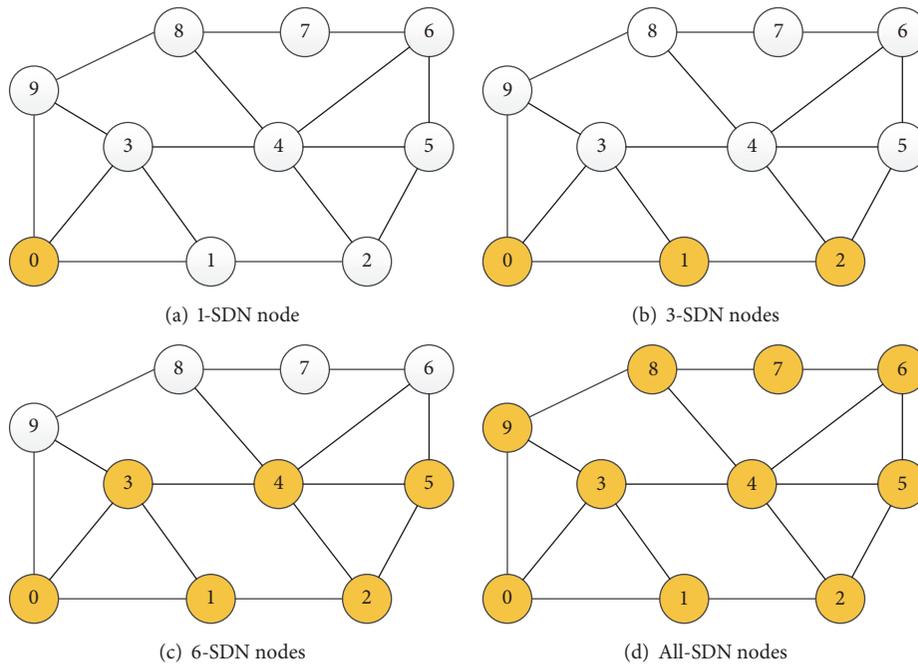


FIGURE 5: Mobile wireless network topology 2.

figure that when the SDN nodes in the network are relatively few, the maximum link utilization in the network obtained from the hybrid routing and forwarding algorithm is almost the same compared with the OSPF routing algorithm. This is because the SDN controller can only control the SDN nodes to manipulate the traffic in the network. When the SDN nodes in the network are few, the traffic in the whole network becomes uncontrollable. Although traffic through the SDN

nodes can be controlled, the maximum utilization rate of the local link in the network is reduced, and the local network can achieve load balancing, and it is difficult to achieve load balancing for the whole network. In addition, by comparing Topology 1 and Topology 2, the benefits of deploying SDN nodes will become more apparent as the number of nodes in the network increases and the network topology becomes more complex.

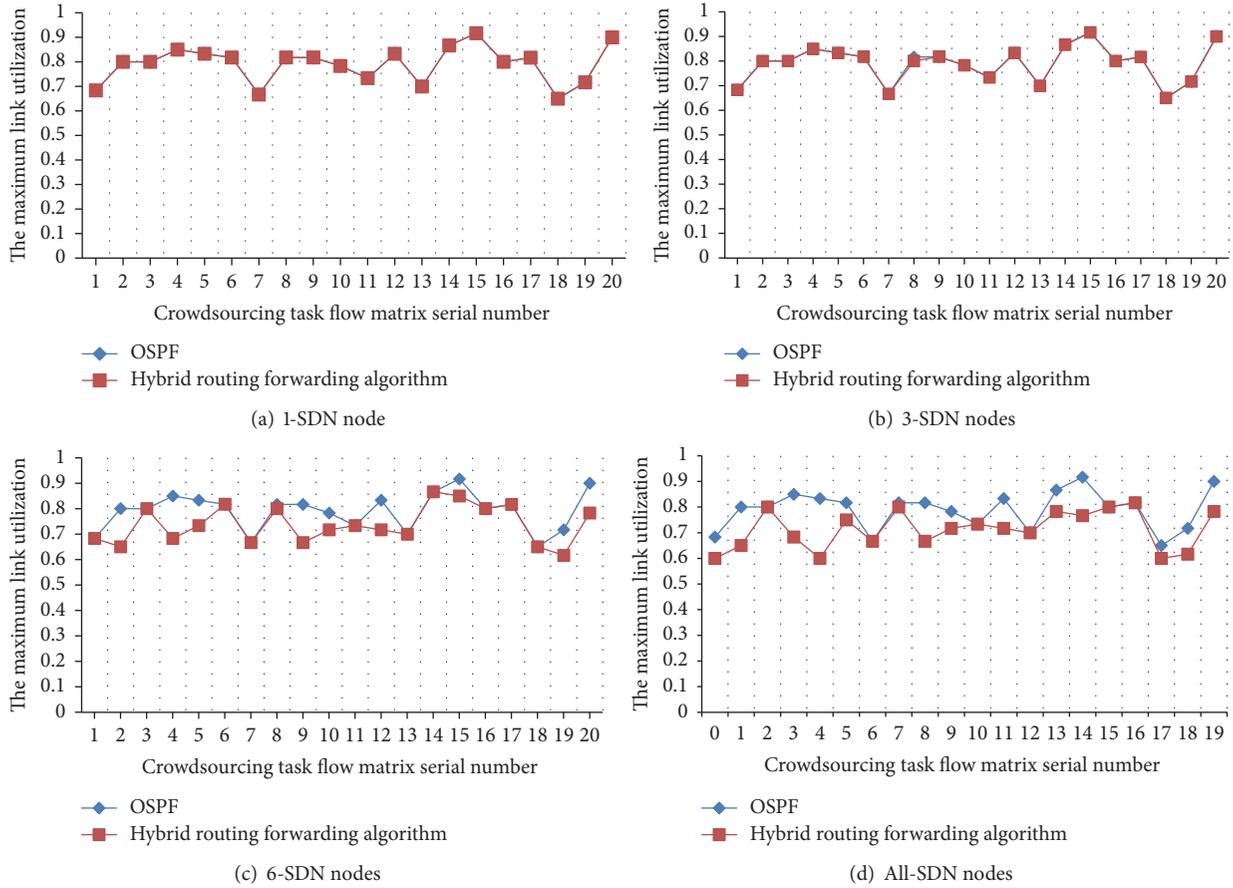


FIGURE 6: Comparison of the maximum utilization with different SDN nodes deployment in Topology 1.

For the congestion control algorithm, we gradually increase the number of SDN nodes in the mobile wireless network. We calculate the throughput of the network through formula (7) and then compare it. The simulation results are shown in Figure 8.

Figure 8 shows the analysis of throughput for different SDN nodes deployment in Topology 1 and Topology 2. It can be observed that the throughput performance of Topology 1 and Topology 2 are both better with SDN nodes increasing. From the comparison results in Figure 8, it can be concluded that our congestion control algorithm can effectively improve the network throughput.

## 6. Conclusion

At present, massive crowdsourcing-based mobile applications have been applied in mobile networks and IoT, targeted at real-time services and recommendation. The frequent information exchanges and data transmissions in collaborative crowdsourcing are continually injected into the current communication networks, which poses great challenges in Mobile Wireless Networks (MWN). This paper focuses on the traffic scheduling and load balancing problem in software-defined MWN and designs a greedy heuristic algorithm as well as a congestion control algorithm to achieve feasible solutions. The proposed traffic scheduling algorithm sorts the

tasks in ascending order according to the amount of tasks and then solves them using the greedy scheme. The packet task is assigned to the corresponding link for forwarding, so that the maximum link utilization in MWN is the least. In the proposed congestion control scheme, the traffic assignment is transformed into a multiknapsack problem, and then the AFSA algorithm is employed to solve the problem. The node selects a subset in its feasible task set and assigns it to the  $p$  links, which makes the maximum amount of tasks allocated without exceeding the limited capacity of each link. The simulation results demonstrate that compared with the traditional schemes the proposed congestion control and traffic scheduling methods can achieve load balancing, reduce the probability of network congestion, and improve the network throughput.

## Conflicts of Interest

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

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (61501105), the National Key Research and Development Program of China (2016YFC0801607),

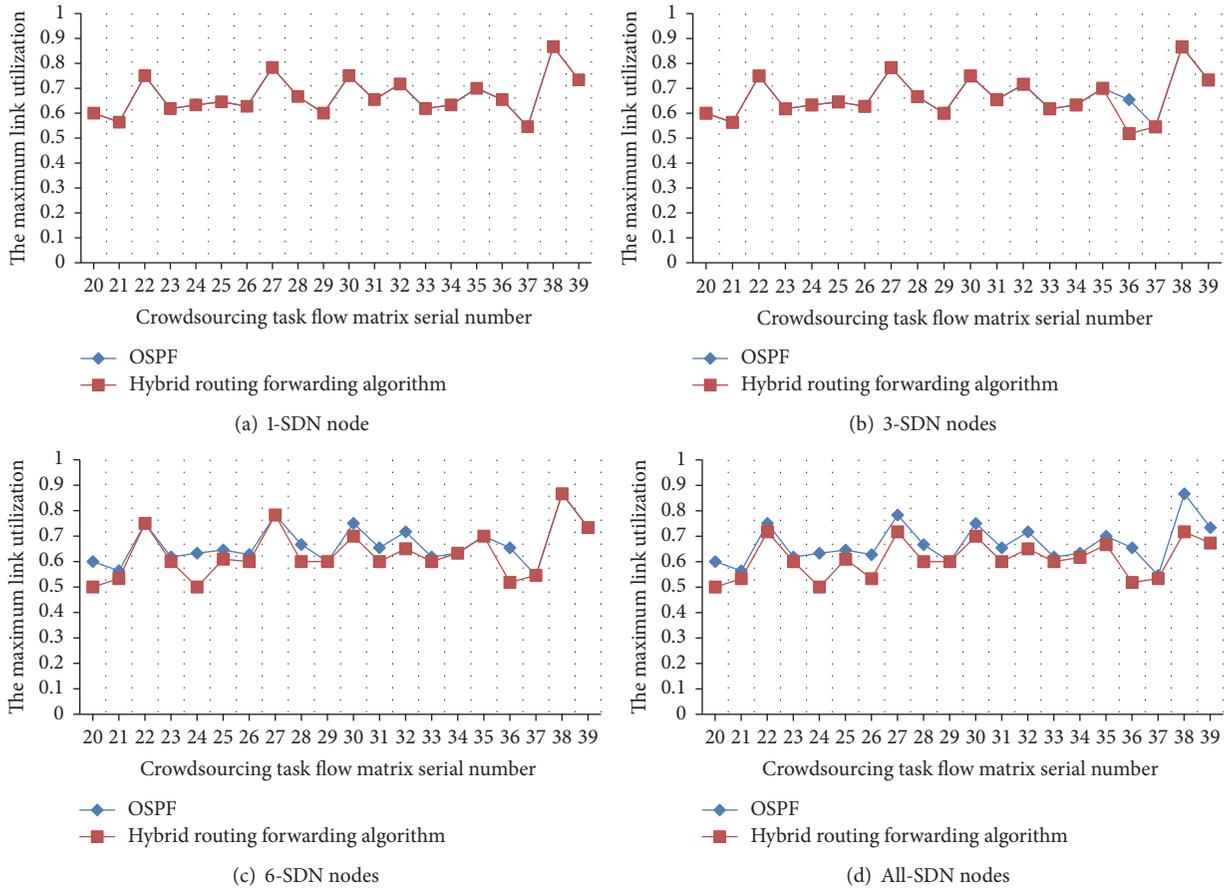


FIGURE 7: Comparison of the maximum utilization with different SDN nodes deployment in Topology 2.

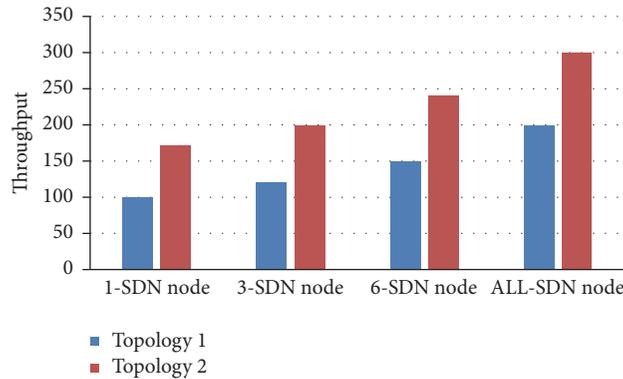


FIGURE 8: Comparison of throughput with different SDN nodes deployment.

the Fundamental Research Funds for the Central Universities (N150404018, N130304001, N150401002, and N161608001), and the Open Research Fund from the State Key Laboratory of Rolling and Automation, Northeastern University, Grant no. 2017RALKFKT002.

**References**

[1] L. von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum, “reCAPTCHA: human-based character recognition via web

security measures,” *American Association for the Advancement of Science: Science*, vol. 321, no. 5895, pp. 1465–1468, 2008.

[2] N. McKeown, T. Anderson, H. Balakrishnan et al., “OpenFlow: enabling innovation in campus networks,” *ACM Sigcomm Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008.

[3] R. Wang, D. Butnariu, and J. Rexford, “Open Flow-based server load balancing gone wild,” in *Proceedings of The 11th USENIX Conference on Hot Topics in Management of Internet, Cloud, And Enterprise Networks And Services*, pp. 12-12, USENIX Association, 2011.

- [4] X. Kong, X. Song, F. Xia, H. Guo, J. Wang, and A. Tolba, "LoTAD: long-term traffic anomaly detection based on crowdsourced bus trajectory data," *World Wide Web Information Systems*, pp. 1–23, 2017.
- [5] Z. Ning, F. Xia, N. Ullah et al., "Vehicular social networks: enabling smart mobility," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 16–55, 2017.
- [6] M. C. Yuen, I. King, and K. S. Leung, "A Survey of crowdsourcing systems," in *Proceedings of the IEEE Third International Conference on Social Computing (SocialCom)*, pp. 766–773, IEEE, Boston, MA, USA, October 2012.
- [7] A. Kittur, J. V. Nickerson, M. Bernstein et al., "The future of crowd work," in *Proceedings of the 2013 Conference on Computer supported cooperative work*, vol. 3-4, pp. 1301–1318, Social Science Electronic Publishing, San Antonio, Texas, USA, February 2013.
- [8] A. Doan, R. Ramakrishnan, and A. Y. Halevy, "Crowdsourcing systems on the world-wide web," *Communications of the ACM*, vol. 54, no. 4, pp. 86–96, 2011.
- [9] Y. Zhao and Q. Zhu, "Evaluation on crowdsourcing research: Current status and future direction," *Information Systems Frontiers*, vol. 16, no. 3, pp. 417–434, 2014.
- [10] A. Kittur, B. Smus, S. Khamkar, and R. E. Kraut, "CrowdForge: Crowdsourcing complex work," in *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, UIST'11*, pp. 43–52, USA, October 2011.
- [11] J. Wang, T. Kraska, M. J. Franklin et al., "CrowdER: crowdsourcing entity resolution," *Proceedings of the VLDB Endowment*, vol. 5, no. 11, pp. 1483–1494.
- [12] J. Wang, G. Li, T. Kraska, M. J. Franklin, and J. Feng, "Leveraging transitive relations for crowdsourced joins," in *Proceedings of the 2013 ACM SIGMOD Conference on Management of Data, SIGMOD 2013*, pp. 229–240, USA, June 2013.
- [13] G. Demartini, D. E. Difallah, and P. Cudré-Mauroux, "Zen-Crowd: Leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking," in *Proceedings of the 21st Annual Conference on World Wide Web, WWW'12*, pp. 469–478, France, April 2012.
- [14] C. Lofi, K. El Maarry, and W.-T. Balke, "Skyline queries over incomplete data - Error models for focused crowd-sourcing," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 8217, pp. 298–312, 2013.
- [15] Y. Sakamoto, Y. Tanaka, L. Yu, and J. V. Nickerson, "The Crowdsourcing design space," in *Proceedings of the International Conference on Foundations of Augmented Cognition: Directing the Future of Adaptive Systems*, vol. 6780 of *Lecture Notes in Computer Science*, pp. 346–355, Springer-Verlag, 2011.
- [16] J. Heer and M. Bostock, "Crowdsourcing graphical perception: Using mechanical Turk to assess visualization design," in *Proceedings of the 28th Annual CHI Conference on Human Factors in Computing Systems, CHI 2010*, pp. 203–212, USA, April 2010.
- [17] D. Karger R, S. Oh, and D. Shah, "Iterative learning for reliable crowdsourcing systems," in *Proceedings of the International Conference on Neural Information Processing Systems*, pp. 1953–1961, Curran Associates Inc., 2011.
- [18] X. Liu, M. Lu, B. C. Ooi et al., "CDAS: A crowdsourcing data analytics system," *Proceedings of the VLDB Endowment*, vol. 5, no. 10, pp. 1040–1051, 2012.
- [19] J. Feng, G. Li, H. Wang et al., "Incremental quality inference in crowdsourcing," in *Proceedings of the International Conference on Database Systems for Advanced Applications, Lecture Notes in Computer Science*, pp. 453–467, Springer International Publishing, 2014.
- [20] A. Marcus, E. Wu, D. Karger, S. Madden, and R. Miller, "Human-powered sorts and joins," *Proceedings of the VLDB Endowment*, vol. 5, no. 1, pp. 13–24, 2011.
- [21] L. Pu, X. Chen, J. Xu, and X. Fu, "Crowdlet: Optimal worker recruitment for self-organized mobile crowdsourcing," in *Proceedings of the 35th Annual IEEE International Conference on Computer Communications, IEEE INFOCOM 2016, USA, April 2016*.
- [22] J. Li, Y. Zhu, Y. Hua, and J. Yu, "Crowdsourcing sensing to smartphones: A randomized auction approach," in *Proceedings of the 23rd IEEE International Symposium on Quality of Service, IWQoS 2015*, pp. 219–224, USA, June 2015.
- [23] Z. He, J. Cao, and X. Liu, "High quality participant recruitment in vehicle-based crowdsourcing using predictable mobility," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications and Networks, IEEE INFOCOM 2015*, pp. 2542–2550, May 2015.
- [24] D. Zhao, X.-Y. Li, and H. Ma, "How to crowdsource tasks truthfully without sacrificing utility: Online incentive mechanisms with budget constraint," in *Proceedings of the 33rd IEEE Conference on Computer Communications, IEEE INFOCOM 2014*, pp. 1213–1221, May 2014.
- [25] Y. Han and H. Wu, "Minimum-Cost Crowdsourcing with Coverage Guarantee in Mobile Opportunistic D2D Networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 10, pp. 2806–2818, 2017.
- [26] E. Estellés-Arolas and F. González-Ladrón-De-Guevara, "Towards an integrated crowdsourcing definition," *Journal of Information Science*, vol. 38, no. 2, pp. 189–200, 2012.
- [27] Y. Lee and B. Mukherjee, "Traffic engineering in next-generation optical networks," *Communications Surveys & Tutorials IEEE*, vol. 6, no. 3, pp. 16–33.
- [28] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, "Overview and principles of internet traffic engineering," IETF RFC 3272, Academy of Science, Engineering and Technology, 2002.
- [29] Z. Ning, Q. Song, Y. Yu, Y. Lv, X. Wang, and X. Kong, "Energy-aware cooperative and distributed channel estimation schemes for wireless sensor networks," *International Journal of Communication Systems*, vol. 30, no. 5, Article ID e3074, 2017.
- [30] C.-Y. Chu, K. Xi, M. Luo, and H. J. Chao, "Congestion-aware single link failure recovery in hybrid SDN networks," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications and Networks, IEEE INFOCOM 2015*, pp. 1086–1094, Hong Kong, May 2015.
- [31] Z. Ning, X. Hu, Z. Chen et al., "A cooperative quality-aware service access system for social internet of vehicles," *IEEE Internet of Things Journal*, pp. 1-1.
- [32] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, Calif, USA, 1979.
- [33] K. Li, S. Wang, S. Xu, and X. Wang, "ERMAO: An enhanced intradomain traffic engineering approach in LISP-capable networks," in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference: "Energizing Global Communications", GLOBECOM 2011, USA, December 2011*.
- [34] X. MA and Q. LIU, "Artificial fish swarm algorithm for multiple knapsack problem," *Journal of Computer Applications*, vol. 30, no. 2, pp. 469–471, 2010.

- [35] Z. Ning, F. Xia, N. Ullah, X. Kong, and X. Hu, "Vehicular Social Networks: Enabling Smart Mobility," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 16–55, 2017.
- [36] H. Yang, *IP routing based on topology of network traffic engineering research*, University of electronic science and technology, 2013.

## Research Article

# Channel Assignment Mechanism for Multiple APs Cochannel Deployment in High Density WLANs

Jianjun Lei , Jianhua Jiang, and Fengjun Shang 

*School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*

Correspondence should be addressed to Jianjun Lei; [leijj@cqupt.edu.cn](mailto:leijj@cqupt.edu.cn)

Received 5 September 2017; Revised 29 December 2017; Accepted 4 January 2018; Published 7 February 2018

Academic Editor: Zhaolong Ning

Copyright © 2018 Jianjun Lei 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 Wireless Local Area Networks (WLANs), cochannel deployment can bound channel access delay and improve network capacity due to mitigating the collision and interference among different Access Points (APs). In this paper, we present a network model and an interference model for multiple APs cochannel deployment and propose a channel assignment mechanism which formulates the channel assignment problem into a time slot allocation problem. Meanwhile, we assign the channel based on the vertex coloring algorithm and make extra polls by utilizing the time slot reservation strategy to improve the channel assignment. Furthermore, we optimize the polling list of APs through classifying the clients to improve the channel utilization. The simulation results show that our proposed algorithm can improve the performance in terms of network throughput, transmission delay, and packet loss rate compared with the DCF (Distributed Coordination Function) and TMCA algorithms.

## 1. Introduction

In recent years, with the development of wireless technologies, the demand of WLANs has risen and its deployment has become denser and denser. The wireless APs are placed everywhere in our daily life environment [1, 2]. A plenty of APs will cause serious collision, influencing users' experience. Hence, eliminating interference among APs under the limited number of channels has become an important issue.

Typically, the clients for IEEE 802.11 operate on the DCF mode, in which clients use the carrier sense medium access with collision avoidance (CSMA/CA) mechanism [3] to compete for the wireless medium to transmit data. Due to the contention nature and throughput fair characteristic of DCF, it is difficult to provide guaranteed bandwidth and bounded access time for each client. Especially, in high density and multi-AP WLANs, high collision probability leads to low efficiency of channel access, and thus too much time is spent in dealing with collision for retransmission [4]. This significantly affects networks performance and user's experience. Although many optimal extensions of DCF are proposed, the competition among nodes cannot be eased and the transmission failure and retransmission increase sharply

with the increase of the deployment density. Due to the limitation of DCF, some literatures suggest adopting centralized channel access mechanism such as the Point Coordination Function (PCF). Although PCF provides contention-free channel access among clients, it is mainly designed to be used in single-AP network and cannot be directly used in multi-AP WLANs.

In this paper, we propose a multi-AP deployment framework, which lets all APs operate in the same channel and coordinates them to obtain the appropriate channel, reducing the system interference. The core idea of the algorithm is to maintain moderate cooperation among APs by the central Access Controller (AC) to achieve high channel utilization. As a result, transmission collision is reduced and wasted time is minimized. We first formulate the channel assignment problem into vertex coloring problem and then present an improved channel assignment strategy which makes extra poll for APs without collision. It is referred to as CCA (cochannel assignment) mechanism and it has been demonstrated that this strategy can improve the channel utilization and guarantee the bandwidth demand in our previous work [5]. Meanwhile, aiming at the possible empty polling problem, we also propose a priority channel access strategy

to improve the channel utilization, which is referred to as optimized CCA mechanism. We preserve as much features of IEEE802.11 standard as possible. Thus, little hardware modification for APs is needed.

The contributions of this paper are listed as follows.

(1) We present the system framework of cochannel deployment and workflow of coordination data traffic among APs, which formulates the channel assignment problem into vertex coloring problem and gives the constraint conditions for channel allocation.

(2) Benefiting from the cochannel deployment framework, we propose a basic CCA algorithm and an optimized CCA algorithm, which not only provides the high concurrent data transmission, but also reduces some empty poll to further enhance the channel utilization.

(3) Compared to our previous work, we conduct some more simulations under the different network densities and scales to demonstrate the feasibility and improvement of algorithm.

The rest of this paper is organized as follows: we survey some related works in Section 2; in Section 3, the network interference model and conflict constraint are introduced; the cochannel assignment strategy and the priority channel access strategy are presented in Section 4; we evaluate the performance of the proposed algorithm in Section 5 and make conclusion in Section 6.

## 2. Related Work

In WLANs, to degrade the interference, the researchers usually assign nonoverlapping channels to neighboring APs. Achanta proposes the classic Least Congested Channel Search (LCCS) [6] algorithm. In the algorithm, APs scan each channel and detect the condition of data transmission in all channels. APs can obtain the load level in each channel and choose the channel in which the load is lighter and the interference is smaller. Since this method only assigns the channel statically on the AP side, the performance may drop sharply under the high density environment. In [7], the authors propose a kind of global channel allocation method based on graph theory. The main idea is to select the vertex according to the saturation and adjacent degree of vertices to color from the available color sets. A similar channel assignment scheme is proposed in [8], where the expected transmission delay due to the interference from a neighboring BSS is regarded as the weight of the edge. In [9], a kind of distributed flexible channel assignment, called DFCA, is designed to assign the channel width and center frequency to each link based on the interfering relation among clients in the network. Similarly papers [10, 11] maximize throughput by the way of allocating nonoverlapping channel. The authors of paper [12, 13] try to eliminate the interference by researching channel allocation and power control jointly. However, both of the methods do not consider the competition between different APs and load distributions. Thus, due to a limited number of channels available, they cannot eliminate interference completely.

Other literatures focus on eliminating the interference by utilizing the PCF (Point Coordination Function). In [14], the authors propose a kind of collision-free client polling

method to mitigate the conflict, which polls the clients in a time slot manner. Though the interference is eliminated effectively, this method needs too much time slots to poll all clients so that the utilization of channel is degraded. In [15], a client polling framework, named MiFi, is proposed for multi-AP deployment, in which CFPs (contention-free periods) at all APs are synchronized and divided into time slots. Neighboring APs are assigned with different time slots, so as to poll all clients without collisions. Nevertheless, in the AP interference model, there are too many constraints to suppress the concurrent polls, thus resulting in the decline of aggregated throughput. In [16], the authors describe an adaptive AP cooperation scheme where each AP autonomously controls its polling period according to channel conditions. Yet, these algorithms mainly concentrate on allocating nonoverlapping channels to APs while the coordination mechanism between APs is ignored.

In this paper, we propose a collision-free channel assignment algorithm to improve the aggregated throughput by utilizing cochannel deployment, which consists of three procedures: (1) conducting a basic channel assignment by vertex coloring algorithm that determines the minimum number of time slots; (2) constructing an improved channel assignment by making extra polls for APs to improve the channel utilization; (3) classifying the clients to optimize the polling list of APs and reduce empty polls. Additionally, since all APs only operate in one channel in our algorithm, multiple vertically nonoverlapping channels also can be used to improve the network capacity in high density WLANs.

## 3. Network Interference Model and Conflict Constraint

*3.1. Network Model.* Figure 1 shows a framework of network deployment. All APs operate in the PCF mode and utilize the same channel. Each client is associated with the AP which has the largest received signal strength indication (RSSI) and can obtain an optimal physical data rate according to the channel condition. The central AC gathers the information of clients by all APs. And with the global information, the AC makes the client data transmission decision in a centralized manner. The workflow is shown in Figure 2.

Due to the global network state awareness of centralized data transmission, the AC gathers all necessary information from other components in this system. In our scheme, the information refers to the associating relation between APs and clients and the neighboring relation among all APs and clients. Then, with the global information, the AC can make an interference table. According to this table, the AC can make an appropriate transmission decision when the AP wants to make polling with a client.

Furthermore, the contention-free periods (CFPs) of APs are divided into several time slots, and within each time slot the AP polls one client or transmits a frame to one of its clients. Different client may have different physical data rates; thus the referred length of time slot  $\Delta$  can be calculated roughly by

$$\Delta = \frac{L_{\text{MaxMPDU}}}{\sum r(C_r/C)}, \quad (1)$$

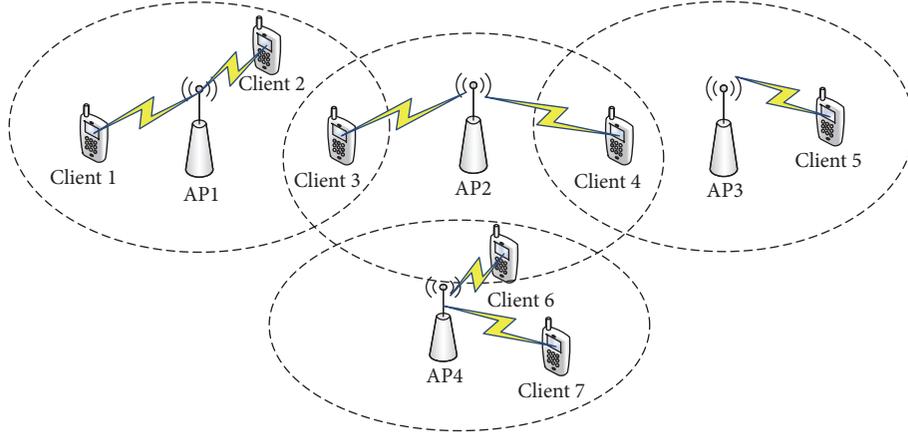


FIGURE 1: An example of WLAN deployment.

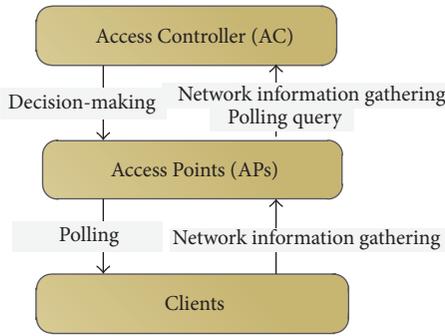


FIGURE 2: Workflow of clients and APs by centralized AC.

where  $L_{\text{MaxMPDU}}$  is the length of the maximum Message Protocol Data Unit (MPDU) and  $C$  and  $C_r$  are the total number of clients and the number of clients whose data rate is  $r$ , respectively. In addition, the value of  $\Delta$  can be updated periodically to adapt to the dynamic of network.

**3.2. Interference Model.** In our proposed channel assignment mechanism, since all APs are operating in the PCF mode, there would not be any interference between two clients in the same AP. However, two clients from neighboring APs may interfere with each other. Assume that two APs of  $AP_m$  and  $AP_n$  have  $S$  clients totally. The interference probability  $I_e$  between  $AP_m$  and  $AP_n$  can be estimated by

$$I_e = \frac{\rho_1 + \rho_2 + \dots + \rho_s}{S}. \quad (2)$$

Equation (2) is subject to

$$\rho_s = \begin{cases} 1, & \text{if } \exists (i, j) \in \Phi_m \times \Phi_n \text{ s.t. } i, j \text{ interfere} \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

where  $\rho_s$  describes the interference relationship and  $\Phi_m$  and  $\Phi_n$  are the sets of clients associated with  $AP_m$  and  $AP_n$ , respectively. Note that the threshold  $I_{\text{th}}$  can change according to the network scenario.

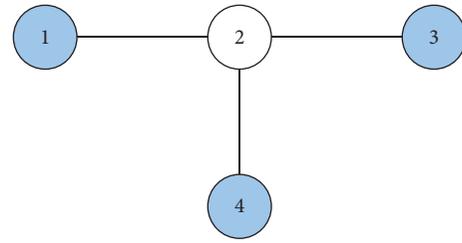


FIGURE 3: Polling conflict graph of the example of WLAN.

**3.3. Conflict Constraint.** Naturally, only one client in the same AP is allowed to transmit or receive data during a time slot. Simultaneously, to avoid transmission collisions, interfering clients from neighboring APs should be polled in different time slots. Hence, a variable  $T_i$  is defined to denote the time slot assigned to client  $i$  and then such conflict constraint can be formally expressed as follows:

$$\begin{aligned} 1 &\leq T_i \leq R, \quad \forall i \in C, \\ T_i &\neq T_j, \quad \text{if } I_e = 1, \quad \forall i, j \in C, \\ T_i &\neq T_j, \quad \text{if } A_m = A_n, \quad \forall i, j \in C, \quad \forall m, n \in A, \end{aligned} \quad (4)$$

where  $R$  is the number of time slots,  $C$  is the set of clients, and  $A$  is the set of APs. Furthermore, the conflict constraint of concurrent polling can also be described by a polling conflict graph  $G = (V, E)$ , where each vertex represents an AP, and there is an edge between two vertices if the interference probability  $I_e$  exceeds the threshold  $I_{\text{th}}$ . Figure 3 shows the polling conflict graph for WLANs deployment in Figure 1.

Note that the problem of polling conflict graph can be solved by the vertex coloring algorithm. In other words, allocating time slot to the corresponding AP in the network is equivalent to coloring a vertex in the graph. However, it is a  $K$ -colorable problem in graph theory, which is a well-known NP-hard problem.

```

Input:
Polling Conflict Graph  $G = (V, E)$ ;
Time Slots Assignment Matrix  $P$ ;
Output:
Time Slots Matrix  $P$ ;
Procedure:
Initiate the element of matrix  $P$  to zero and time slot  $T_r = 0$ 
while  $(|V| > q)$ 
   $I_m = \text{MIS}()$ {Use MIS algorithm to find a maximal independent set in  $G$ }
end while
  Find the maximal independent sets as  $\{I_1, \dots, I_r\}$ .
for  $i = r$  down to 1
  Assign  $r$  minimum available colors to  $\{I_r\}$  and set  $T_r = 1$  of  $\{I_r\}$ .
end for

```

ALGORITHM 1: The basic channel assignment procedure.

## 4. Channel Assignment Algorithm

In this section, we propose the CCA mechanism based on vertex coloring for cochannel channel assignment, which includes a basic channel assignment and an improved channel assignment strategy. In addition, to minimize polling overhead, we also propose an optimized CCA, which arranges the different channel access priority based on the clients' load level.

*4.1. Basic Channel Assignment.* The basic channel assignment for a cochannel deployment is aiming to avoid the interference among APs by allocating time slots to all APs in a collision-free manner based on PCF. It includes two components. The first one is the coloring algorithm given a graph  $G = (V, E)$  and the number of colors  $R$ , which seeks out a feasible color scheme with minimal number of colors. The second component works out an efficient slot assignment by using the coloring scheme, which also can be described by time slots assignment matrix.

To solve the vertex coloring problem, a mass of heuristic algorithms has been proposed in graph theory and many of them have been applied into practice, such as the maximal independent set (MIS) [17] and the EXTRACOL [18] which both can provide the optimal results. Thus, we use the MIS algorithm to solve the problem of the vertex coloring graph and realize the basic channel assignment strategy. The procedure is described by Algorithm 1.

We construct the interference collision graph by using the proposed interference model. And then, the MIS algorithm is applied to solve the coloring problem repeatedly and the coloring scheme is recorded. After that, the time slots assignment matrix is established according to the coloring scheme and all APs can poll their associating clients in the assigned time slots. By this procedure, an AP cannot poll any of its associating clients when one of its neighboring APs is receiving or transmitting data from or to its clients.

*4.2. Time Slots Reservation Scheme.* In the basic channel assignment, part of APs may be idle in some time slots. To utilize such idle time slots adequately and improve channel

utilization, we propose an improved channel assignment mechanism that is called time slots reservation (TSR) scheme. Crucially, the AP can poll its associating clients in idle time slots by querying the polling list of all the neighboring APs. As mentioned in previous section, such queries can be supported by the AC with the global network neighboring table and data transmission dynamic information. Therefore, we define the variable  $C_{\text{sen}}$  to denote the set of clients which are in the carrier sensing range of APs. The procedure is described by Algorithm 2.

The idle APs in the next time slot query the set  $N$  consisting of all adjacent APs and also query the polling list of all neighboring APs in the set  $N$ . By that, APs can judge whether the next polling client satisfies the conflict constraint, which can avoid the interference among the neighboring APs in the next polling. By this procedure, the concurrent poll is permitted among the neighboring APs, thus resulting in improving the channel utilization. A concurrent data transmitting example shown in Figure 1 is that client 3 can be required to transmit data by the polling from AP2 when the AP1 is transmitting the data to client 1. However, in legacy IEEE802.11 DCF or PCF standard, this concurrency cannot occur due to their CSMA/CA mechanism.

*4.3. Priority Channel Access Strategy.* Some undesirable poll occurring will result in the wasting of bandwidth if the AP polls some clients without data to send or receive. Therefore, two-class priority client polling mechanism based on queuing theory is proposed in some literatures [19–21]. However, due to the side-effects of PCF, they work inefficiently in multiple APs cochannel deployment and high density WLANs. Thus, to minimize the waste of bandwidth and polling overhead, we present an optimized CCA strategy, which classifies the clients into three categories and polls them in different priority. The procedure is described by Algorithm 3.

In this strategy, an extra field is used to describe whether the clients need to transmit data. The clients in the polling list of AP are classified into three categories: (1) the clients with both uplink and downlink data, (2) the clients with uplink or downlink data, and (3) the clients without data needed to

**Input:**  
The requirement of AP  $k$  to reserve the time slot.

**Output:**  
A Boolean value;

**Procedure:**  
Query the set  $N$  consisting of the neighboring APs of AP  $k$ ;  
*while* ( $m = 1 : |N|$ )  
  Query the next polling client  $j$  of  $N_m$  ( $N_m \in N$ );  
  *if* (Client  $j \notin C_{\text{sen}}$ ) *then*  
    Reservation succeeds.  
  *else*  
    Reservation fails.  
  *end if*  
*end while*

ALGORITHM 2: The improved channel assignment procedure.

**Input:**  
The original polling list  $N$ .

**Output:**  
The new polling list  $N'$ ;

**Procedure:**  
*while* ( $i = 1 : |N|$ )  
  *if* ( $\text{down}(\text{client}[i]) == 1 \ \&\& \ \text{up}(\text{client}[i]) == 1$ ) *then*  
     $\text{queue1}[r] += \text{client}[i]$ ; //  $r$  is the list of clients with both the uplink and  
    downlink data.  
  *else if* ( $\text{down}(\text{client}[i]) == 1 \ \parallel \ \text{up}(\text{client}[i]) == 1$ ) *then*  
     $\text{queue2}[s] += \text{client}[i]$ ; //  $s$  is the list of clients with the uplink of downlink  
    data.  
  *else* ( $\text{down}(\text{client}[i]) == 0 \ \&\& \ \text{up}(\text{client}[i]) == 0$ )  
     $\text{queue3}[t] += \text{client}[i]$ ; //  $t$  is the list of clients without data needed to  
    transmit.  
  *end if*  
*end while*

ALGORITHM 3: Priority channel access strategy.

transmit. The AP polls the clients in category 1 with highest priority, assuming that the probability is  $1 - P$ . And then, the clients in categories 2 and 3 are polled with probability  $P$ . Note that the value of probability  $P$  should be adjusted and the polling list should be updated according to the real network scenario. It is noted that only a trivial modification for the data frame of client is needed to adapt this mechanism. In legacy IEEE802.11 standard, one bit field “more data” in “frame control” field is used to denote that more data will be transmitted from AP to some sleeping clients. In our algorithm, this flag bit also is used by the clients; thus the extra modification hardly occurs for clients. In addition, the time slot allocations for both basic channel assignment and improved channel assignment are updated only when interference or network topology changes. Therefore, the overhead of both mechanisms is negligible compared with data traffic in the network.

## 5. Performance Evaluations

**5.1. Simulation Methodology.** In this section, we conduct simulations by OPNET14.5, evaluate the proposed CCA and optimized CCA mechanisms, and compare them with the basic DCF method and the TMCA algorithm proposed in [22]. In the simulation, we set up the FTP server to simulate the real network traffic and set the uplink and downlink traffic in half, respectively. The experimental parameters are shown in Table 1. The clients are randomly distributed in a  $1000 \times 800 \text{ m}^2$  field and each client is associated with an AP according to the largest RSSI mechanism mentioned in previous section. And we acquire the transmission rate of each client by utilizing the relationship between the transmission rate  $r_{ij}$  and the Signal to Interference plus Noise Ratio (SINR)  $\gamma_{ij}$ , which is shown in Table 2 [23].

We evaluate the performance in terms of transmission delay, packet loss rate, and network throughput under the

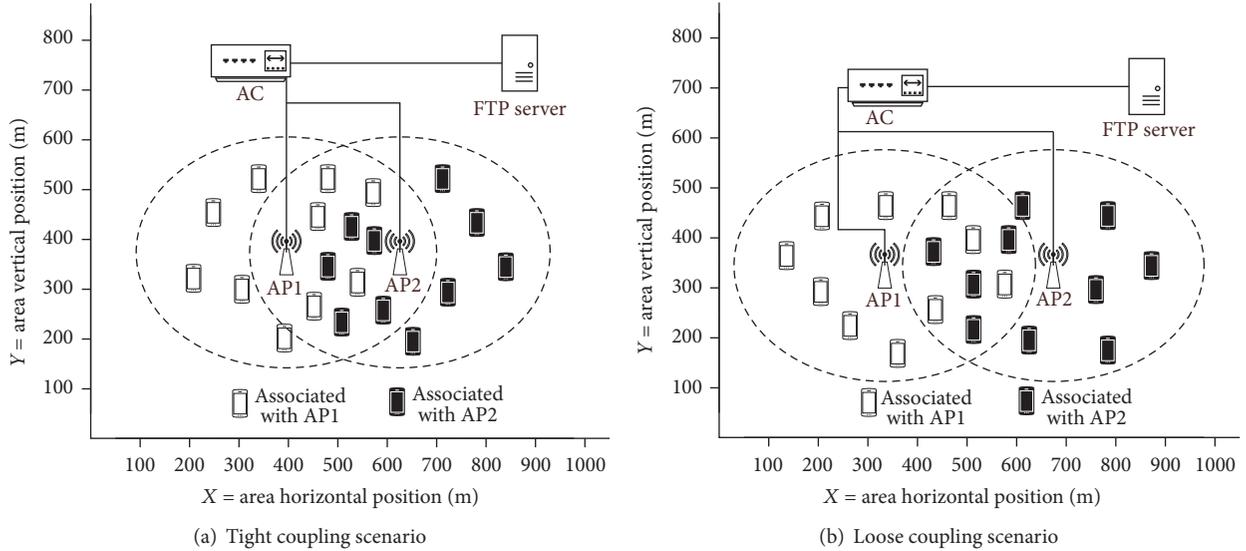


FIGURE 4: Basic network deployment with two APs.

TABLE 1: Experimental parameters.

Parameters	Value
ACK	14 bytes
RTS	20 bytes
DIFS	50 us
CTS	14 bytes
SIFS	10 us
EIFS	300 us

scenario of basic and high density network deployment. The transmission delay refers to the average time from the sending time in source client to the reception time in target client, which is computed by

$$T = \frac{1}{N} \sum_{i=1}^N (RT(i) - ST(i)), \quad (5)$$

where  $N$  is the number of successfully transmitted data frames and  $RT(i)$  and  $ST(i)$  are the reception and transmission times of the  $i$ th data frame, respectively.

**5.2. Simulation in Basic Network Deployment.** The basic network deployment is shown in Figure 4. Two APs are deployed in the scenario of tight coupling and loose coupling, which denotes that the APs have the same coverage overlap area but they can interfere with each other directly or indirectly, respectively.

**5.2.1. Transmission Delay.** We first examine the transmission delay of the DCF, CCA, and optimized CCA algorithms under various client densities, where the number of clients increases from 5 to 30 for each AP. As shown in Figure 5, the transmission delay of three algorithms increases with the increasing of the number of clients. Reasonably, a larger

number of clients result in higher collision probability. Nevertheless, our proposed CCA and optimized CCA algorithms perform almost similarly and better compared with the DCF scheme in both tight and loose coupling deployment. The delay of our algorithms is reduced by 45% in tight coupling and 32% in loose coupling, respectively.

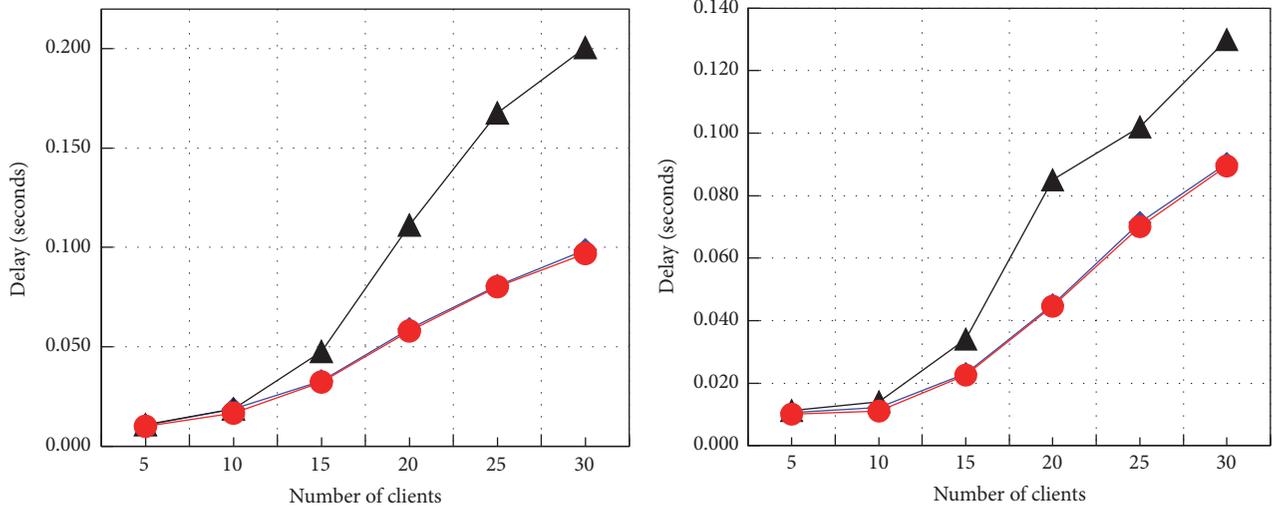
**5.2.2. Packet Loss Rate.** We also evaluate the packet loss rate under various client densities. Figure 6 shows the simulation results of three algorithms. Obviously, our proposed algorithms can achieve slightly lower packet loss rate in both tight and loose coupling deployment. The reason is that our algorithms utilize the time-sharing scheduling strategy, which can reduce the occurrence conflict and thus result in the lower packet loss rate.

**5.2.3. Network Throughput.** Figure 7 shows the average network throughput tendency as the network density changes. When the number of clients for each AP is not beyond 20, the DCF algorithm obtains better throughput compared with the CCA and optimized CCA algorithms, which may be due to little channel collision and low network overhead for DCF mechanism when the network density is low. However, the network throughput of DCF method begins to decrease as more clients compete for the channel. Our proposed algorithms still keep high throughput in high density deployment. Obviously, the optimized CCA algorithm can achieve higher throughput compared with the other algorithms in high density network, which is due to the priority scheduling policy performing effectively.

**5.3. Simulation in High Density Network Deployment.** We also evaluate our proposed algorithms and TMCA algorithm in more complex and higher density networks; the scenarios are shown in Figure 8.

TABLE 2: The relationship between effective bit rates and SINRs in IEEE 802.11 Standard.

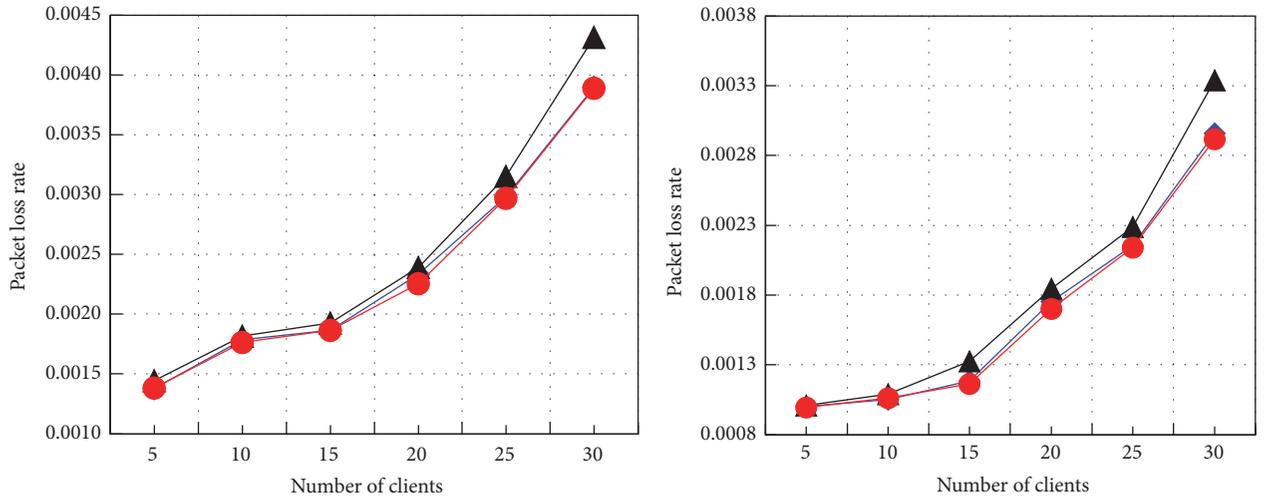
$\gamma$ (dB)	6-7.8	7.8-9	9-10.8	10.8-17	17-18.8	18.8-24	24-24.6	>24.6
$C_r$ (Mbps)	6	9	12	18	24	36	48	54



(a) The transmission delay in tight coupling deployment

(b) The transmission delay in loose coupling deployment

FIGURE 5: Transmission delay under various network densities.



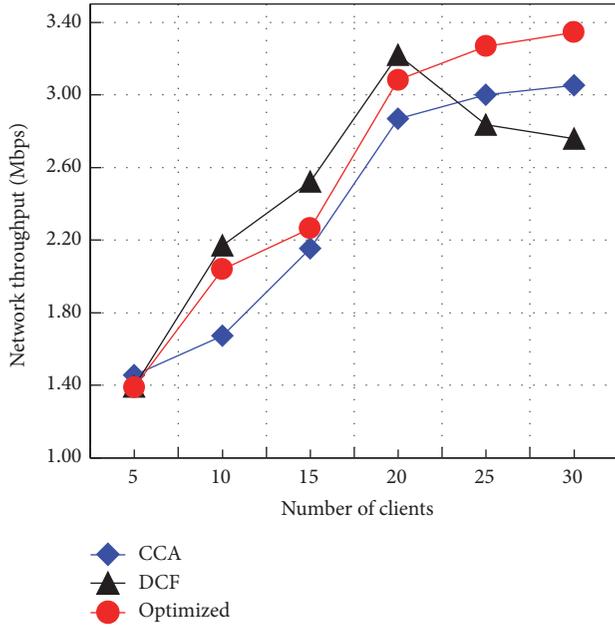
(a) The packet loss rate in tight coupling deployment

(b) The packet loss rate in loose coupling deployment

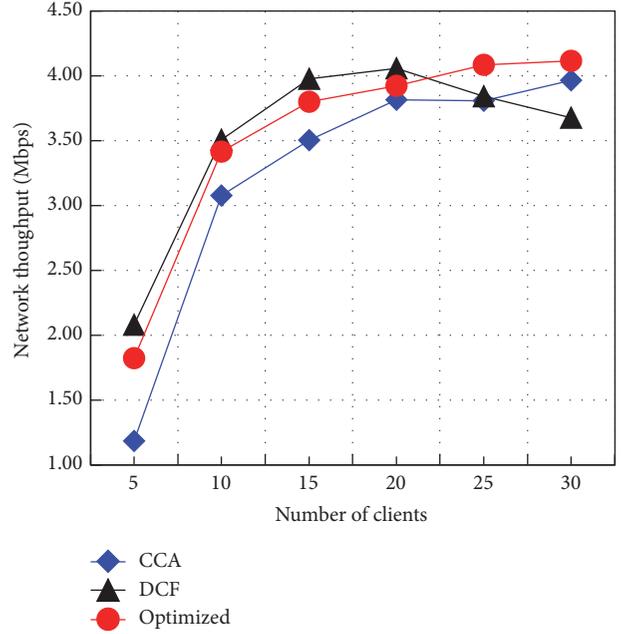
FIGURE 6: Packet loss rate under various network densities.

5.3.1. *Transmission Delay.* Figure 9 shows the transmission delay of our proposed algorithms and TMCA algorithm under the topology depicted in Figures 6(a) and 6(b). The CCA and optimized CCA algorithms perform better and they can reduce transmission delay by 15% compared with

the TMCA algorithm. The TMCA algorithm performs badly in high density deployment due to its binary back-off algorithm. Nevertheless, our proposed algorithms can operate the channel access coordinately among APs and thus result in bounded access delay.

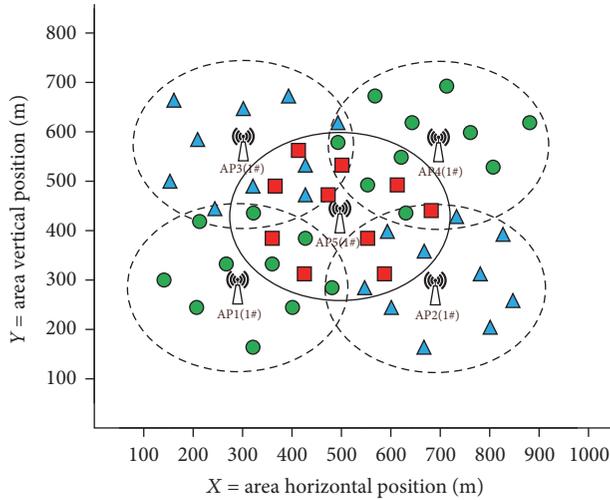


(a) The average network throughput in tight coupling deployment

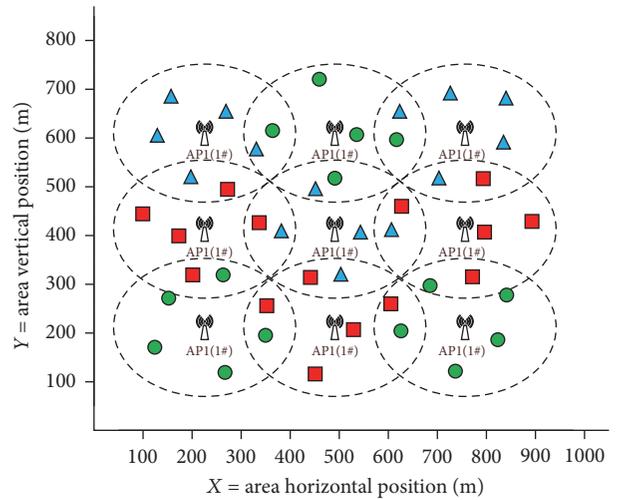


(b) The average network throughput in loose coupling deployment

FIGURE 7: Average network throughput under various network densities.



(a) 5 APs' scenario



(b) 9 APs' scenario

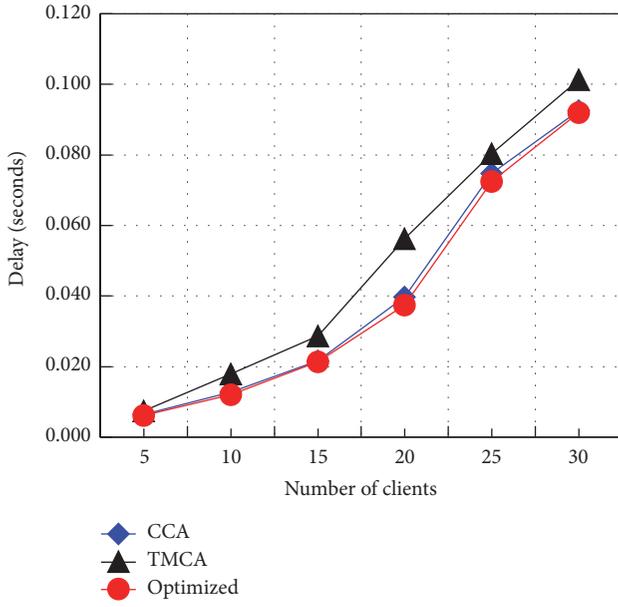
FIGURE 8: High density network deployment with 5 and 9 APs.

5.3.2. *Packet Loss Rate.* Figure 10 shows the packet loss rate for 5 and 9 APs' deployment under various client densities. Obviously, our proposed CCA and optimized CCA algorithms can achieve lower packet loss rate compared with TMCA algorithm. The TMCA algorithm only has three nonoverlapped channels available, which can cause severe cochannel interference when more than three APs are deployed in the same overlap area. However, our proposed algorithms can coordinate AP operating to reduce collisions and packet loss rate.

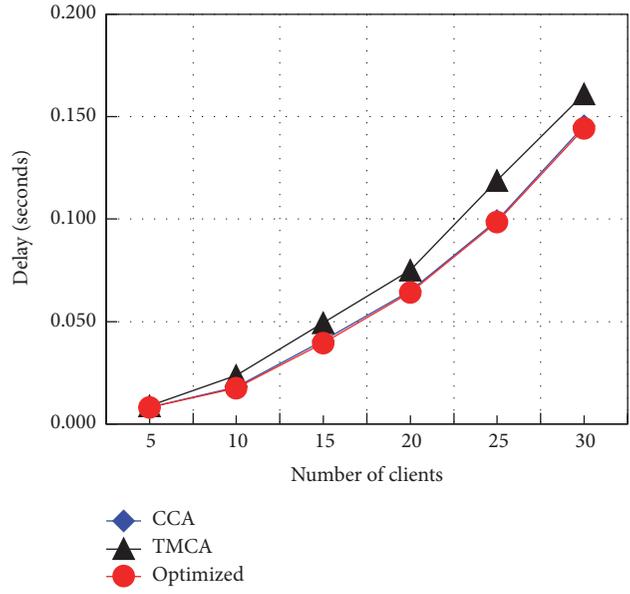
5.3.3. *Network Throughput.* The TMCA, CCA, and optimized CCA algorithms are also evaluated in terms of the normalized throughput, which is computed by

$$X_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}, \quad (6)$$

where  $X$  is the measurement data and  $X_{\max}$  and  $X_{\min}$  are the maximum and minimum of the measurement data, respectively. Figure 11 shows the simulation results for the number of clients for each AP increasing from 5 to 30. The

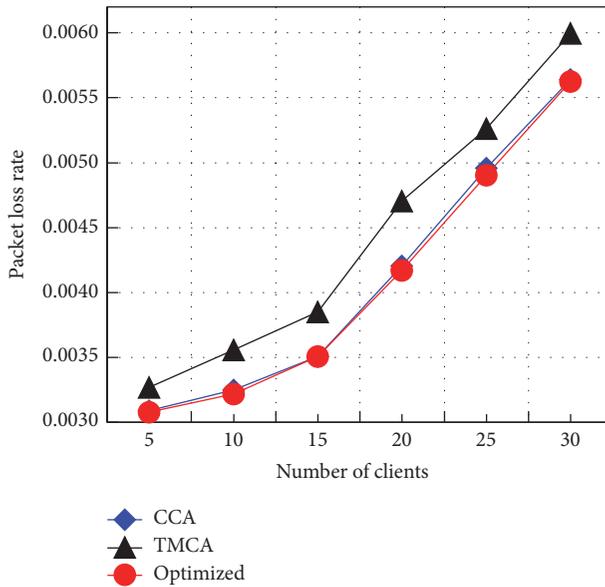


(a) The delay in the topology with 5 APs

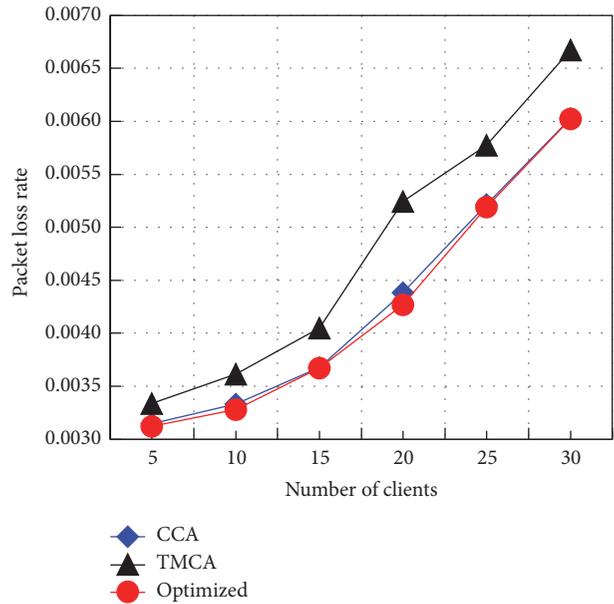


(b) The delay in the topology with 9 APs

FIGURE 9: Transmission delay in 5 and 9 APs' deployment under various network densities.



(a) The packet loss rate in the topology with 5 APs



(b) The packet loss rate in the topology with 9 APs

FIGURE 10: Packet loss rate in 5 and 9 APs' deployment under various network densities.

throughput of TMCA algorithm is slightly higher than that of our proposed algorithms when the number of clients is less than 20. But as the number of clients increases, the throughput of TMCA algorithm decreases while those of the CCA and optimized CCA algorithms always stay at a high level and they can achieve 10% improvement compared with TMCA, which is due to more collision for TMCA algorithm only using three nonoverlapped channels. Meanwhile, the

optimized CCA algorithm performs more impressively since it optimizes the scheduling strategy and reduces the polling number of those clients without data transmission, thus improving the throughput.

## 6. Conclusions

In this paper, we propose a channel assignment algorithm based on PCF mechanism for multiple APs cochannel

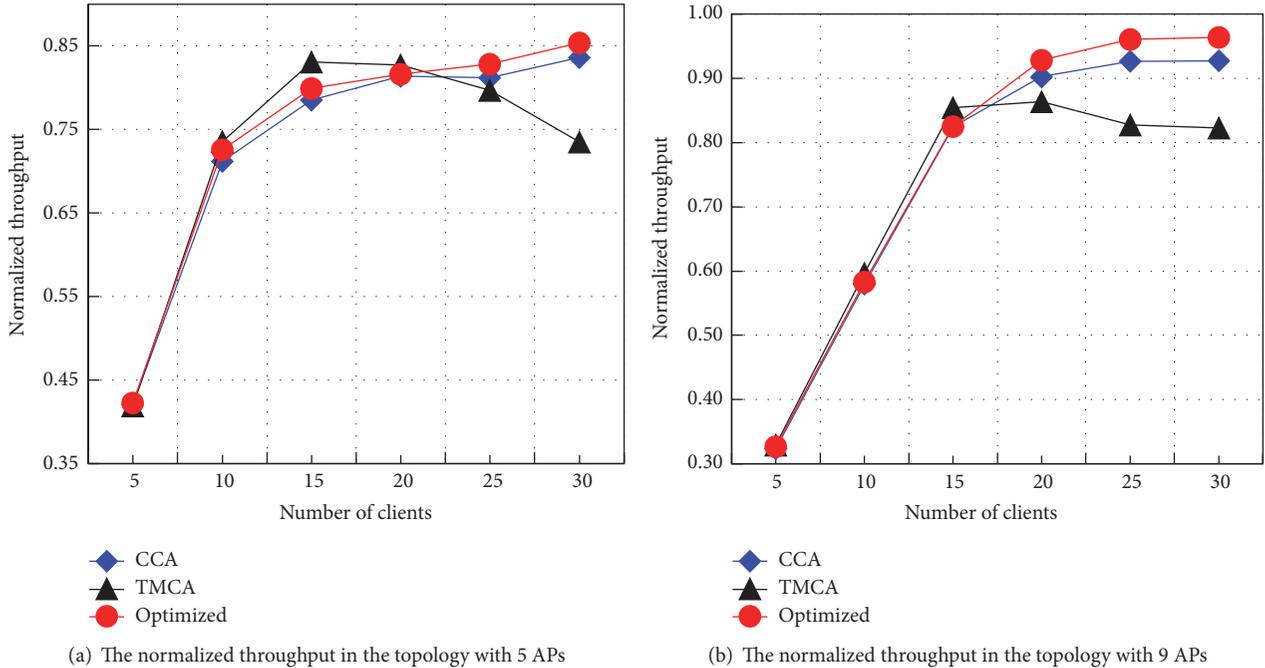


FIGURE 11: Normalized throughput under various network densities in 5 and 9 APs' deployment.

deployment scenario. We present the network and interference model and formulate the problem of channel assignment into a vertex coloring problem. We propose the CCA algorithm which assigns the channel based on the vertex coloring approach, and then, by applying time slot reservation, make extra polls to improve the efficiency of channel assignment by applying time slot reservation. In addition, an optimized CCA algorithm also is proposed by classifying the clients and optimizing the polling list to reduce empty polls. Finally, we evaluate the proposed algorithms under different network densities, which shows that our proposed algorithm can obtain high throughput, low packet loss rate, and bounded access delay.

## Disclosure

Figures 3, 4, and 8 and Table 1 are reprinted by permission from Springer Nature: Proceeding of the 6th Computer Science On-line Conference (CSOC2017) (A Novel Co-Channel Deployment Algorithm Based on PCF in Multiple APs and High Density WLANs, Jianjun Lei and Jianhua Jiang) copyright 2017.

## Conflicts of Interest

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

## Authors' Contributions

Fengjun Shang contributed to presenting the priority channel access strategy and guiding the evaluating experiments for the optimized CCA mechanism.

## Acknowledgments

This work was supported by the National Science Foundation of China (61672004), the Program for Innovation Team Building at Institutions of Higher Education in Chongqing (CXTDX201601021), and Scientific and Technological Research Program of Chongqing Municipal Education Commission (KJ1500439). This paper was presented in part at Springer CSOC2017. The authors would like to thank the anonymous CSOC2017 reviewers for their helpful comments on an earlier version of this work.

## References

- [1] S. Fiehe, J. Riihijvi, and P. Mahonen, "Experimental Study on Performance of IEEE 802.11n and Impact of Interferers on the 2.4 GHz ISM Band," in *ACM IWCMC*, pp. 47–51, 2010.
- [2] D. Gong and Y. Yang, "AP Association in 802.11n WLANs with Heterogeneous Clients," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '12)*, pp. 1440–1448, March 2012.
- [3] T. Cuzanuskas and A. Anskaitis, "Multi-polling game for IEEE 802.11 networks," in *Proceedings of the 3rd IEEE Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE '15)*, November 2015.
- [4] D. Zhao, M. Zhu, and M. Xu, "Leveraging SDN and OpenFlow to mitigate interference in enterprise WLAN," *Journal of Networks*, vol. 9, no. 6, pp. 1526–1533, 2014.
- [5] J. Lei and J. Jiang, "A Novel Co-channel Deployment Algorithm Based on PCF in Multiple APs and High Density WLANs," in *Proceedings of the the 6th Computer Science On-line Conference (CSOC '17)*, vol. 575, pp. 126–135, 2017.
- [6] M. Achanta, Method and apparatus for least congested channel scan for wireless access points. USA, 20060072602, 2006.

- [7] H. Zhang, H. Ji, and W. Ge, "Channel assignment with fairness for multi-AP WLAN based on distributed coordination function," in *Proceedings of the 2011 IEEE Wireless Communications and Networking Conference (WCNC '11)*, pp. 392–397, March 2011.
- [8] Y. Liu, W. Wu, B. Wang, T. He, S. Yi, and Y. Xia, "Measurement-based channel management in WLANs," in *Proceedings of the IEEE Wireless Communications and Networking Conference 2010 (WCNC '10)*, pp. 1–6, April 2010.
- [9] C.-C. Hsu, Y.-A. Liang, J. L. G. Gomez, C.-F. Chou, and C.-J. Lin, "Distributed flexible channel assignment in WLANs," in *Proceedings of the 2013 IEEE Wireless Communications and Networking Conference (WCNC '13)*, pp. 493–498, April 2013.
- [10] B. Kauffmann, F. Baccelli, A. Chaintreau, V. Mhatre, K. Papagiannaki, and C. Diot, "Measurement-based self organization of interfering 802.11 wireless access networks," in *Proceedings of the IEEE INFOCOM 2007: 26th IEEE International Conference on Computer Communications*, pp. 1451–1459, May 2007.
- [11] H. Zhang, H. Ji, and W. Ge, "Channel Assignment with Fairness for multi-AP WLAN based on Distributed Coordination," in *Proceedings of the 2011 IEEE Wireless Communications and Networking Conference (WCNC '11)*, pp. 392–397, March 2011.
- [12] D. Xu, Z. Feng, Y. Li, and P. Zhang, "Fair channel allocation and power control for uplink and downlink cognitive radio networks," in *Proceedings of the 2011 IEEE GLOBECOM Workshops (GC Wkshps '11)*, pp. 591–596, December 2011.
- [13] Y. Wu, Y. Sun, Y. Ji, J. Mao, and Y. Liu, "A joint channel allocation and power control scheme for interference mitigation in high-density WLANs," in *Proceedings of the 15th IEEE International Conference on Communication Technology (ICCT '13)*, pp. 98–103, November 2013.
- [14] D. Gong, Y. Yang, and H. Li, "High-throughput collision-free client polling in multi-AP WLANs," in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference: "Energizing Global Communications" (GLOBECOM '11)*, December 2011.
- [15] Y. Bejerano and R. S. Bhatia, "MiFi: A framework for fairness and QoS assurance for current IEEE 802.11 networks with multiple access points," *IEEE/ACM Transactions on Networking*, vol. 14, no. 4, pp. 849–862, 2006.
- [16] H. Nishimura, R. Fujiwara, M. Miyazaki, and M. Katagishi, "Access point cooperation for collision-free polling between distributed real-time networks," in *Proceedings of the 26th IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communications, PIMRC 2015*, pp. 1057–1061, China, September 2015.
- [17] S. Gualandi and F. Malucelli, "Exact solution of graph coloring problems via constraint programming and column generation," *INFORMS Journal on Computing*, vol. 24, no. 1, pp. 81–100, 2012.
- [18] Q. Wu and J. K. Hao, "Coloring large graphs based on independent set extraction," *Computers and Operations Research*, vol. 39, pp. 283–290, 2012.
- [19] L.-Y. Bao, D.-F. Zhao, and Y.-F. Zhao, "An efficient polling scheme for wireless LANs," in *Proceedings of the 2010 International Conference on Computer Design and Applications, ICCDA 2010*, pp. V4540–V4544, China, June 2010.
- [20] Q. Liu, D. Zhao, and H. Ding, "An improved polling scheme for PCF MAC protocol," in *Proceedings of the 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '11)*, pp. 407–411, September 2011.
- [21] J.-Y. Yeh and C. Chen, "Support of multimedia services with the IEEE 802.11 MAC protocol," *IEEE International Conference on Communications*, vol. 1, pp. 600–604, 2002.
- [22] D. Gong, M. Zhao, and Y. Yang, "Channel assignment in multi-rate 802.11n WLANs," in *Proceedings of the 2013 IEEE Wireless Communications and Networking Conference (WCNC '13)*, pp. 392–397, April 2013.
- [23] Y. Cui, W. Li, and X. Cheng, "Partially overlapping channel assignment based on "node orthogonality" for 802.11 wireless networks," in *Proceedings of the IEEE INFOCOM 2011*, pp. 361–365, April 2011.

## Research Article

# IoT-B&B: Edge-Based NFV for IoT Devices with CPE Crowdsourcing

He Zhu  and Changcheng Huang

*Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada*

Correspondence should be addressed to He Zhu; [hzhu@sce.carleton.ca](mailto:hzhu@sce.carleton.ca)

Received 23 August 2017; Revised 6 December 2017; Accepted 13 December 2017; Published 4 February 2018

Academic Editor: Kuan Zhang

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

For embracing the ubiquitous Internet-of-Things (IoT) devices, edge computing and Network Function Virtualization (NFV) have been enabled in branch offices and homes in the form of virtual Customer-Premises Equipment (vCPE). A Service Provider (SP) deploys vCPE instances as Virtual Network Functions (VNFs) on top of generic physical Customer-Premises Equipment (pCPE) to ease administration. Upon a usage surge of IoT devices at a certain part of the network, vCPU, memory, and other resource limitations of a single pCPE node make it difficult to add new services handling the high demand. In this paper, we present IoT-B&B, a novel architecture featuring resource sharing of pCPE nodes. When a pCPE node has sharable resources available, the SP will utilize its free resources as a “bed-and-breakfast” place to deploy vCPE instances in need. A placement algorithm is also presented to assign vCPE instances to a cost-efficient pCPE node. By keeping vCPE instances at the network edge, their costs of hosting are reduced. Meanwhile, the transmission latencies are maintained at acceptable levels for processing real-time data burst from IoT devices. The traffic load to the remote, centralized cloud can be substantially reduced.

## 1. Introduction

Customer-premises equipment (CPE) devices, such as routers, switches, residential gateways, and set-top boxes, have been deployed at the subscriber’s premises to originate, route, and terminate communications between the customer premises and the central office [1]. In the wake of cloud computing and Network Function Virtualization (NFV) [2, 3], Service Providers (SPs) leverage virtual Customer-Premises Equipment (vCPE) as Virtual Network Function (VNF) instances on top of generic physical Customer-Premises Equipment (pCPE), in search of rebuilding a dynamic revenue stream [4]. There can be enough resources for pCPE to deploy VNFs locally [5], while pCPE can also coordinate with the cloud if VNF scale-out is needed to accommodate heavier usage.

Taking advantage of centralized cloud services in the core networks has benefits [6] because of scalable and flexible computing capabilities. However, large-number deployments of Internet-of-Things (IoT) devices bring challenges to VNFs

running in a centralized cloud, as the network traffic load would be drastically increased by transmitting data between the core and the edge of the network. Such traffic overhead can become unacceptable with excessive data transmission, causing high processing delay or even service outage due to the congestion of the network. Meanwhile, high usage of the cloud networks would jack up the price per usage, resulting in a higher-than-expected operating expense (OPEX).

Recent research has been aware of the explosive growth of devices in the edge of the networks. The concepts of fog computing [7] and edge computing [8] were proposed to move the initial handling of raw data to the edge for IoT devices. Although the fog can mitigate the load of the core network, the power of the Customer Edge (CE), namely, the computational capabilities of CPE, is buried. While a single pCPE node has limited resources and typically serves a designated location, the aggregated computing capabilities of pCPE nodes across the edge of a network can be powerful: pCPE nodes have time-varying resource usage that does not always reach full workloads. For instance, the home gateways

typically have significantly lower usage in business hours as their users leave for work, while office gateways become idle after hours. If the spare resources of pCPE can be shared within the network edge, VNFs will be able to roam around the edge. Both SP and users will benefit from the considerable capabilities of the sharable resources. Meanwhile, the VNFs deployed on the pCPE nodes keep most traffic within the edge and reduce the traffic to the core network.

It certainly sounds interesting to utilize the time-varying computational resources. However, CPE resource sharing faces challenges:

- (i) It is unclear how much SP can benefit from spare resources of pCPE nodes compared to traditional virtualization without resource sharing.
- (ii) Service availability becomes a concern. A pCPE node's availability can be jeopardized if it no longer has enough resources to host VNFs. It can also be down due to power outages or user pulling the plugs. The availability of offloaded VNFs must be ensured by enforcing proper redundancy.
- (iii) The users need to be motivated to consent to contributing their pCPE nodes for resource sharing. They would not do so without incentives or discounts. Incentives returned to users are required in order to benefit both the SP and its end users.

In this paper, we propose an architecture to allow sharing resources of pCPE within the network edge, namely, IoT-B&B. We discuss the scenario that SP deploys VNFs, which are vCPE instances, to both the cloud and the available pCPE nodes participating in the resource sharing program. As Figure 1 shows, when a sharable pCPE node is not actively used by its owner, it will be treated as a "bed-and-breakfast" place for vCPE instances to "stay." SP will have the permission to utilize free resources through the resource manager to deploy VNFs of other users from the same edge network. The following contributions are made for enabling crowdsourcing at the network edge by utilizing resources of pCPE nodes:

- (i) We propose an architecture extended from ETSI NFV architecture and interfaces [9] to support resource sharing of pCPE nodes. The pCPE nodes at the network edge are treated as compute hosts which can have VNF instances deployed. They are grouped together and abstracted into the NFV Infrastructure (NFVI) layer. A resource manager is embedded in the NFVI and can leverage placement algorithms to make placement decisions.
- (ii) A model is presented to evaluate the cost of assigning a VNF instance to a pCPE node and to the remote cloud. Multiple factors are considered to determine the cost, including remaining resources, network transmission delay, and availability requirements.
- (iii) A placement algorithm called "IoT-B&B Algorithm" is also presented to assign vCPE instances to pCPE nodes with the goal of finding a cost-efficient pCPE node for each VNF.

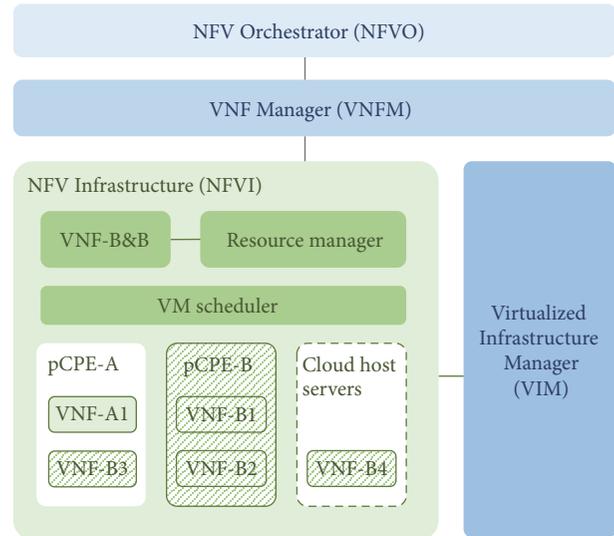


FIGURE 1: The system architecture is extended from ETSI NFV architecture and interfaces [9]. It has a resource manager integrated with NFVI to schedule VNF placement, leveraging the VNF-B&B algorithm. The architecture enables pCPE resources to be part of the NFVI along with cloud host servers, where each pCPE node can have VNFs deployed for multiple users in the edge network. For instance, when pCPE-B needs more VNFs to handle its overloaded use, they can be deployed on both pCPE-A (VNF-B3) and the cloud (VNF-B4).

- (iv) We implement a system with the IoT-B&B architecture with steps to set up the NFVI and the system's life cycle. Numerical results are shown to demonstrate the placement algorithm's effectiveness.

We divide the contents into the following sections. Section 2 formulates the problem. Section 3 proposes the IoT-B&B Algorithm based on the problem formulation. Then, Section 4 is presented, covering the actual implementation of the system, followed by the numerical results in Section 5. The related work is illustrated in Section 6. Section 7 concludes the paper and lists future work items.

## 2. Problem Formulation

In this section, we formulate the problem by modeling the resource properties and constraints of the network edge. The resource types we discuss are limited to CPU, memory, and network bandwidth. We believe these three types of the resources are most representative for cost estimation and optimization. Adding consideration of more resource types will not necessarily change the optimization model. Then, the properties of the VNF instances are defined and annotated. All notations defined and used are also summarized in the Notations Used in Problem Formulation. Note that the terms "VNF"; "VNF instance"; and "vCPE instance" in this paper are interchangeable.

*2.1. Connected pCPE at Network Edge.* We discuss one particular network edge, which includes all pCPE nodes under

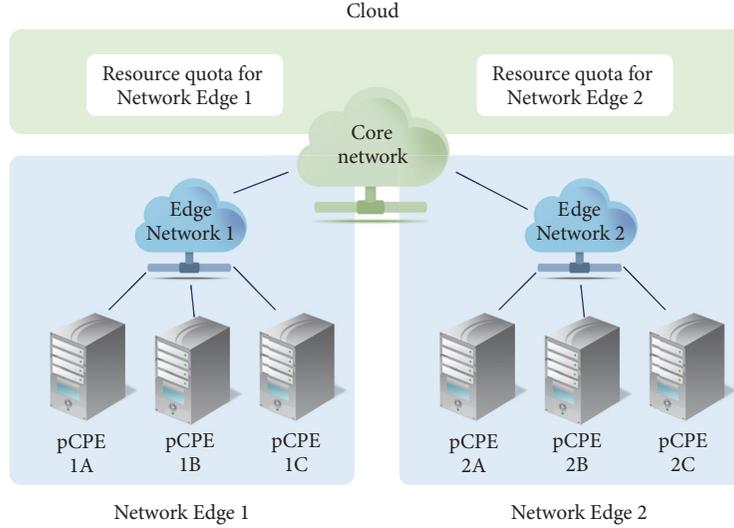


FIGURE 2: Topologies at network edges. All pCPE nodes can communicate with each other if they are within the same network edge.

it. A network edge is defined as the networks connecting all pCPE nodes under it. We model a network edge to have a topology such that each pCPE node within it can communicate with another. An example with two network edges can be found in Figure 2. The nodes of pCPE 1A, 1B, and 1C group as one network edge connected to each other via Edge Network 1, while the other one consists of the nodes 2A, 2B, and 2C, connected by Edge Network 2.

Based on the definition above, the pCPE nodes and their links at the network edge can be modeled as a directed graph  $G = (V, L)$ . Set  $V$  represents all pCPE nodes at the network edge, while  $L$  is the set of all connected links from one vCPE node to another. A pCPE node in  $V$  is denoted by  $v$ , and there is  $v \in V$ . Define the total number of pCPE nodes to be  $n_V$ . Let  $v_i$  be a specific pCPE node in  $V$ , such that

$$v_i \in V, \quad \forall i \in [1 \cdots n_V]. \quad (1)$$

Since all pCPE nodes are connected to each other,  $G$  is strongly connected. For any data transmitted from one node  $v_i$  to another node  $v_j$  in  $V$ , there exists a link  $l_{ij}$ , such that

$$l_{ij} \in L, \quad \forall i, j \in [1 \cdots n_V], \quad i \neq j. \quad (2)$$

The network edge is connected to the core network via a logical link, denoted by  $l_c$ . The capacity of  $l_c$  is limited due to budget reasons: the network edge has a certain bandwidth quota. The total bandwidth to the cloud must be kept within the quota. In reality,  $l_c$  can be a group of links connecting the remote cloud.

## 2.2. VNF Types and Resource Requirement Profiles (Flavors).

The network functions serving the IoT networks have been encapsulated into various types of VNFs. Each type provides a user with a specific service. When the demand increases for a certain type of VNF to a certain level that it exceeds the maximum capacity of current VNF instances, the VNF scales out by increasing the number of Virtual Machine (VM)

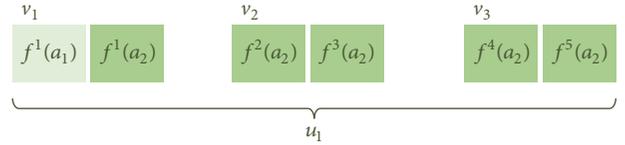


FIGURE 3: An example of VNF instances grouped by  $u_1$ .

instances, so that more requests can be processed at the same time. Depending on the purpose, different types of VNF have different resource requirements. The term “flavor” is used to describe the resource requirements profile of an instance of the VNF, including the number of vCPUs, the amount of memory, the size of the disk. We use  $f$  to define a VNF instance.

Assign  $a$  to identify a specific type of network function and  $n_a$  to be the number of network function types. A VNF instance with type “ $a$ ” can then be represented by  $f(a)$ . The CPU, memory, and bandwidth requirements of  $f(a)$  are then denoted by  $U(f(a))$ ,  $M(f(a))$ , and  $B(f(a))$ .

**2.3. The User of a VNF Instance.** Another property of a VNF instance is the user who owns and uses it. Suppose each pCPE node has one unique owner. This is a valid assumption as the actual device on customer premise is typically not shared and belongs to the entity who pays for the service. For a pCPE node  $v$ , its owner  $u$  uses a set of VNF instances to satisfy its needs, regardless of where the instances are deployed. When referring to a specific node  $v_i$ ,  $i \in [1 \cdots n_V]$ , we represent the node’s owner by  $u_i$ , where  $i \in [1 \cdots n_V]$ .

Figure 3 illustrates an example of VNF instances grouped by the user  $u_1$ . There is 1 instance of  $f(a_1)$  and 5 instances of  $f(a_2)$ . The only instance of  $f(a_1)$ , denoted by  $f^1(a_1)$ , is on  $v_1$ , along with  $f^1(a_2)$ . Instances  $f^2(a_2)$  and  $f^3(a_2)$  are on  $v_2$ . Instances  $f^4(a_2)$  and  $f^5(a_2)$  are on  $v_3$ . The numbers and types of VNF instances grouped by their users are determined by

user activities and can change dynamically according to user demands.

Based on the user of VNF instances, the annotation of  $f$  can be extended from  $f(a)$  to  $f(a, u)$ , where  $u$  is the user who owns and uses  $f$ .

**2.4. Places to Deploy VNF Instances.** Placement decisions are made based on the flavors, which are the resource requirement profiles, of VNF instances and the resource capacities of pCPE nodes. A VNF instance of a certain user  $u_i$ , denoted by  $f(a, u_i)$ , can be deployed at either of the locations below.

**2.4.1. B&B Deployment.** For a VNF instance  $f(a, u_i)$ , any pCPE node within the same network edge can be considered as a candidate place to deploy, also known as B&B deployments. A B&B deployment is performed when  $f(a, u_i)$  is deployed on a pCPE node  $v_j$ . For a B&B deployment of  $f(a, u_i)$ , its notation can be extended to  $f(a, u_i, v_j)$ , where  $v_j$  is the place to deploy  $f$ . We use the set  $F_v$  to define all VNF instances deployed on the pCPE node  $v$ .

Particularly, when there are enough resources on the pCPE node  $v_i$  of  $u_i$ ,  $f(a, u_i)$  can be deployed on  $v_i$  locally. For a local deployment of  $f(a, u_i)$ , its notation becomes  $f(a, u_i, v_i)$ , where  $v_i$  is the place to deploy  $f$ .

**2.4.2. Cloud Deployment.** Besides B&B deployments, the remote cloud can be chosen as an alternative place to deploy VNF instances. For a cloud deployment of  $f(a, u_i)$ , its notation can be extended to  $f(a, u_i, c)$ , where  $c$  is the place to deploy  $f$ , and  $c$  stands for the remote cloud location. We use the set  $F_c$  to define all VNF instances deployed on the cloud.

**2.5. Factors to Impact Placement Decisions.** The following factors will impact the placement decisions.

**2.5.1. pCPE Resource Capacity.** Every pCPE node  $v$  has its own resource capacity to host a limited number of VNF instances. As a compute node, the resources for VNF instances are as follows: virtual CPUs (vCPUs) and memory. We assume that there is a plenty of disk space on each pCPE node for any virtual instances deployed. Therefore, the disk space of the pCPE nodes is not in the scope of discussion. Let  $U(v)$  denote the number of vCPUs  $v$  can provide. Let  $M(v)$  be the total amount of memory for VNF instances from  $v$ . Figure 4 provides an example of the resource capacity for a pCPE node with 20 vCPUs and 10 GB memory in total. There are currently 2 instances of VNF  $f(a_1)$  and 3 instances of  $f(a_2)$  deployed on it.

**2.5.2. Edge Network Transmission Delay.** Comparing with the core network transmission delay to be discussed in the next section, the transmission delay between pCPE nodes at the network edge can be ignored as the bandwidth between edge nodes is considered plenty and the transmission delay is small enough to be ignored in the discussion.

**2.5.3. Core Network Transmission Delay.** The core network transmission delay of a VNF instance  $f$  offloaded to the cloud

vCPU capacity = 20					
$f^1(a_1)$	$f^2(a_1)$	$f^1(a_2)$	$f^2(a_2)$	$f^3(a_2)$	Free vCPU
2	2	4	4	4	
Memory capacity = 10 GB					
$f^1(a_1)$	$f^2(a_1)$	$f^1(a_2)$	$f^2(a_2)$	$f^3(a_2)$	Free memory
2 GB	2 GB	1 GB	1 GB	1 GB	

FIGURE 4: Capacities of the number of vCPUs and the amount of memory of a pCPE node. At the moment, there are 2 instances of VNF  $f(a_1)$  and 3 instances of VNF  $f(a_2)$ .

is defined as the time consumed by offloading the VNF instance to the cloud and is denoted by  $t(f)$ , while  $T_{\max}(f)$  is the maximum allowed network delay for a specific network function instance. The actual delay must not exceed this limit, or the requests would eventually overflow the buffer and cause malfunction of the VNF.

There are many factors that may affect the core network transmission delay. As stated in [10], the transmission delay to the cloud can be calculated by

$$\text{Transmission delay} = \frac{\text{Message size}}{\text{Network bandwidth}}. \quad (3)$$

For the same message, the less available bandwidth left from the network edge to the cloud, the longer transmission delay will be. During the peak hours, more VNF instances are requested by users concurrently across the network and would congest  $l_c$ . The severity of direct oversubscription is reflected by the residue bandwidth of the link from the edge switch to the cloud, denoted by  $R(l_c)$ , which is the link's total bandwidth  $B(l_c)$  less the mean bandwidth usage for all remote VNF instances. The smaller residue bandwidth  $R(l_c)$  there is, the bigger core network transmission delay  $t(l_c)$  we should expect.

Because all network function instances offloaded to a remote cloud share the same link  $l_c$  that connects the edge network to the remote cloud and the same cloud environment, we assume the core network transmission delay is the same for all offloaded network function instances to simplify the modeling process. Let  $B(l_c)$  be the total bandwidth of  $l_c$  and  $t(l_c)$  be the transmission delay of  $l_c$ . VNF instances take up the bandwidth of  $l_c$  to communicate with the pCPE nodes. The latency of the core network is therefore highly correlated to the bandwidth consumption of  $l_c$ .

Let  $F_c$  denote the set of VNF instances deployed in the remote cloud.  $R(l_c)$  can be calculated by

$$R(l_c) = B(l_c) - \sum_{f \in F_c} B(f), \quad \forall f \in F_c. \quad (4)$$

Even if  $l_c$  is not congested, the remote cloud environment may be degraded by other sources. For example, overloaded VNFs from other users or applications of a shared cloud could affect other VNFs on the same host because of overcommitting. To better utilize the resources on a host, overcommitting is enabled by default [11]. However, the performance could be jeopardized if some VNFs are taking up most of the resource

[12]. We define the delay not directly caused by the network edge as a  $T_d$ , where the value of  $T_d$  changes according to the load of the cloud environment.

Oversubscription will result in a higher core network delay  $t(l_c)$ . For all VNF instances offloaded to the cloud, there must be

$$t(l_c) \leq T_{\max}(f), \quad \forall f \in F_c. \quad (5)$$

The equation above ensures the functionality of all VNFs offloaded to the cloud with the existence of  $t(l_c)$ . It draws a limit of how much VNF offloading can be done, since an oversubscribed cloud environment would increase  $t(l_c)$ . Based on the calculation of transmission delay, we further model  $t(l_c)$  to be inversely proportional to  $R(l_c) + b$  and proportional to  $T_d$  with  $b$  as a constant scoping the maximum core network delay when the bandwidth of  $l_c$  is depleted:

$$t(l_c) = \frac{T_d}{R(l_c) + b}, \quad 0 \leq R(l_c) \leq B(l_c). \quad (6)$$

Combining (5) with (6), we have

$$\frac{T_d}{R(l_c) + b} \leq T_{\max}(f), \quad \forall f \in F_c. \quad (7)$$

When  $T_d$  gets higher or  $R(l_c)$  becomes lower, the value of  $t(l_c)$  would exceed  $T_{\max}(f)$  of one or more offloaded VNF instances.

**2.6. Cost of Offloading to Edge Network.** By enabling the resource sharing of pCPE nodes across the network edge, SP benefits from extended containers hosting the VNFs. The costs of leveraging these resources include the following.

**2.6.1. Incentives Returned to End Users.** By encouraging the users to participate in the resource sharing program and to consent to share, it is necessary to give incentives to the users based on the amount of resource shared. We denote the unit incentive of CPU, memory, and bandwidth usage for pCPE node  $v$  as  $w_U(v)$ ,  $w_M(v)$ , and  $w_B(v)$ , respectively. One exception is that when the pCPE node is hosting VNF instances used by its own user, the incentives do not apply.

**2.6.2. Extra Redundancy.** Since the availability of the pCPE nodes is lower than the cloud, more standby VNF instances are needed. We define the redundancy factor  $\gamma$  to be the mean number of standby VNF instances needed for one VNF instance on B&B nodes.

Based on the two factors above, the cost of offloading a VNF instance  $f$  to any of the pCPE nodes, denoted by  $S(f, v)$ , is calculated as below:

$$S(f, v) = (1 + \gamma) \cdot [w_U(v)U(f) + w_M(v)M(f) + w_B(v)B(f)]. \quad (8)$$

**2.7. Cost of Offloading to Cloud.** As defined earlier, we use  $c$  to represent the remote cloud in general to deploy VNF instances, to make it distinct from B&B deployments. Although the resources in the cloud, especially in the public cloud, can be considered infinite [13] due to the elasticity of the amount of resources available, for a specific edge network, there are budgets for resources assigned to it. Therefore, resources in the cloud to an edge network are limited when we model them.

The total amounts of vCPUs, memory, and network bandwidth assigned to the edge network we discuss in the cloud are denoted by  $U(c)$ ,  $M(c)$ , and  $B(c)$ , respectively. The cost of offloading to the cloud depends on its usage. In general, the less resources left in the cloud for the edge network, the higher unit price our model gives, because the cloud needs to be available as an alternative place to host vCPE instances. Allowing the cloud resources to be drained too early will jeopardize the flexibility of placement and do harm to the service availability.

Let  $w_U(c)$  stand for the unit cost for consuming the cloud's CPU resource. We model  $w_U(c)$  to be inversely proportional to the cloud's remaining vCPUs with the constant of proportionality  $W_U$ . The remaining number of vCPUs is denoted by  $R_U(c)$ . The total cost of vCPUs for a VNF instance  $f$  to be offloaded to the cloud, denoted by  $S_U(f, c)$ , is then

$$\begin{aligned} S_U(f, c) &= w_U(c)U(f) = \frac{W_U}{R_U(c) + \delta}U(f) \\ &= \frac{W_U U(f)}{U(c) - \sum_{f' \in F_c} U(f') + \delta}. \end{aligned} \quad (9)$$

In (9),  $\delta$  is a small positive number to avoid dividing by zero.

Also, let  $w_M(c)$  represent the unit cost of the memory resource in the cloud. Like vCPUs,  $w_M(c)$  is modeled to be inversely proportional to the cloud's residue memory with the constant of proportionality  $W_M$ . The residue memory resource of the cloud is denoted by  $R_M(c)$ . Similar to the induction of (9), we have the total cost of cloud memory for a VNF instance  $f$  denoted by  $S_M(f, c)$ , where  $\delta$  is a small positive number to avoid dividing by zero:

$$\begin{aligned} S_M(f, c) &= w_M(c)M(f) = \frac{W_M}{R_M(c) + \delta}M(f) \\ &= \frac{W_M M(f)}{M(c) - \sum_{f' \in F_c} M(f') + \delta}. \end{aligned} \quad (10)$$

Let  $w_B(c)$  denote the unit cost of the remote cloud's network bandwidth. The variable  $w_B(c)$  is defined to be proportional to the core network delay  $t(l_c)$  with the constant of proportionality  $W_B$ . We define the total cost of bandwidth used between the VNF instance  $f$  and the cloud as  $S_B(f, c)$ . As defined previously,  $b$  is a constant representing the maximum

core network delay when the bandwidth of  $l_c$  is depleted. Then we have

$$\begin{aligned} S_B(f, c) &= w_B(c) B(f) = W_B t(l_c) B(f) \\ &= W_B \frac{T_d}{R(l_c) + b} B(f) \\ &= \frac{W_B T_d B(f)}{B(l_c) - \sum_{f' \in F_c} B(f') + b}. \end{aligned} \quad (11)$$

From (9), (10), and (11), the cost of offloading a VNF instance  $f$  to the cloud, denoted by  $S(f, c)$ , is then calculated as

$$\begin{aligned} S(f, c) &= S_U(f, c) + S_M(f, c) + S_B(f, c) \\ &= \frac{W_U U(f)}{U(c) - \sum_{f' \in F_c} U(f') + \delta} \\ &\quad + \frac{W_M M(f)}{M(c) - \sum_{f' \in F_c} M(f') + \delta} \\ &\quad + \frac{W_B T_d B(f)}{B(l_c) - \sum_{f' \in F_c} B(f') + b}. \end{aligned} \quad (12)$$

**2.8. Objective and 0-1 Integer Programming Formulation.** SP would like to reduce the total cost of deploying and running VNF instances for all users across the network edge. The VNF instances can be deployed either to the remote cloud location  $c$  or to the pCPE location  $v$ . Based on where the VNF instances are offloaded, we identify two portions of costs offloading the VNF instances: (1) to the cloud and (2) to B&B nodes. The objective of the optimization is to minimize the total offloading cost of the SP.

#### Variables

- (i)  $X(f, v)$ : a group of Boolean variables representing if each VNF instance  $f$  is deployed on the B&B node  $v$ .
- (ii)  $X(f, c)$ : a group of Boolean variables representing if each VNF instance  $f$  is deployed on the remote cloud  $c$ .

$$X(f, v) = \begin{cases} 0, & f \text{ not deployed on } v; \\ 1, & f \text{ deployed on } v. \end{cases} \quad (13)$$

$$X(f, c) = \begin{cases} 0, & f \text{ not deployed on cloud;} \\ 1, & f \text{ deployed on cloud.} \end{cases}$$

#### Objective

$$\text{Minimize } \sum_{f \in F} \sum_{v \in V} S(f, c) X(f, c) + S(f, v) X(f, v) \quad (14)$$

#### Constraints

$$X(f, c) + \sum_{v \in V} X(f, v) = 1, \quad \forall f \in F, \quad (15)$$

$$U(c) - \sum_{f \in F_c} U(f) \geq 0, \quad (16)$$

$$M(c) - \sum_{f \in F_c} M(f) \geq 0, \quad (17)$$

$$\frac{T_d}{R(l_c) + b} \leq T_{\max}(f), \quad \forall f \in F_c, \quad (18)$$

$$U(v) - \sum_{f \in F_v} U(f) \geq 0, \quad \forall v \in V, \quad (19)$$

$$M(v) - \sum_{f \in F_v} M(f) \geq 0, \quad \forall v \in V. \quad (20)$$

#### Remarks

- (i) Function (14) is the objective function. It minimizes the total cost of offloading VNFs instances to the cloud and to B&B nodes.
- (ii) Constraint (15) ensures that every VNF instance  $f \in F$  is only deployed at one place.
- (iii) Constraints (16) and (17) are the capacity bounds of the CPU and memory of the cloud. Each type of the three resources leveraged by all VNF instances offloaded to the cloud must not exceed the cloud's allocated resource capacity for the network edge.
- (iv) Constraint (18) sets the bottom line of the residue bandwidth for  $l_c$  between the network edge and the cloud, which is essentially setting a limit for the number of VNFs offloaded to the cloud.
- (v) Constraints (19) and (20) are the capacity bounds for CPU and memory of every pCPE node. Each type of the resources used by all VNFs offloaded to the vCPEs must not exceed these bounds.

### 3. IoT-B&B Heuristic Placement Algorithm

From the 0-1 integer programming in the previous section, we design a heuristic algorithm to achieve a lower cost by choosing the first valid candidate place to deploy new VNF instances, after the candidate places are sorted by the remaining resources.

**3.1. Preliminary Resource Check.** For every request of deploying a new VNF instance, we first use Algorithm 1 to check the placement eligibility of every pCPE node, as well as the remote cloud. If the place does not meet the resource constraints of the instance, it will be excluded from the list of candidate places. By calling the function `GETCANDIDATES( $f$ )`, a list of candidate places will be returned from the input of a specific VNF instance  $f$  and the current resource level. The list will be sorted by considering the lowest percentage of

```

(1) function GETSORTEDCANDIDATES( $f$ )
(2)   create an empty list candidates
(3)   for all  $v \in V$  do
(4)     if ISRESOURCEENOUGH( $v, f$ ) then
(5)       add  $v$  to candidates
(6)     end if
(7)   end for
(8)   sort candidates by remaining resources descending
(9)   if ISRESOURCEENOUGH( $c, f$ ) then
(10)    add  $c$  to candidates
(11)  end if
(12)  return candidates
(13) end function

(14) function ISRESOURCEENOUGH( $v, f$ )
(15)  if  $v$  is  $c$  then                                     ▷ Check delay for cloud
(16)    link_bw_left  $\leftarrow B(l_c) + b$ 
(17)    for all  $f' \in F_c$  do
(18)      link_bw_left  $\leftarrow$  link_bw_left  $- B(f')$ 
(19)    end for
(20)    delay  $\leftarrow T_d /$  link_bw_left
(21)    if delay  $< T_{\max}(f)$  then
(22)      return false                                       ▷ Too much delay
(23)    end if
(24)  end if
(25)  if  $R_U(v) < U(f)$  then
(26)    return false                                       ▷ vCPU not enough
(27)  end if
(28)  if  $R_M(v) < M(f)$  then
(29)    return false                                       ▷ memory not enough
(30)  end if
(31)  if  $R_B(v) < B(f)$  then
(32)    return false                                       ▷ bandwidth not enough
(33)  end if
(34)  return true                                         ▷ validation passes
(35) end function

```

ALGORITHM 1: IoT-B&amp;B resource eligibility check algorithm.

remaining resource type, in a descending order. For example, if a pCPE node has 90% of vCPU left but only 20% of memory left, then the remaining memory will be used for sorting.

**3.2. Cost Estimation.** With the list of eligible candidate places for a VNF instance  $f$ , we can further estimate the cost of  $f$  deployed at each place. Algorithms 2 and 3 provide implementation of the cost model from Section 2. Algorithm 3 defines the function to choose the place for VNF instance  $f$  at the lowest cost, namely, CHOOSEPLACE( $f$ ). The function first calls GETCANDIDATES( $f$ ) in Algorithm 1 to get the places eligible for deploying  $f$ . Then, for each eligible place, the cost is checked based on the type of the place based on Algorithm 2. If the place is the cloud, CLOUDCOST( $f$ ) is invoked for cost; if the place is a pCPE node, the function BNBCOST( $f, v$ ) is called instead. After iterating all eligible places, the place with the lowest cost is selected and returned.

**3.3. Time Complexity.** Algorithm 1 has the time complexity of  $O(n \log(n))$  because of sorting the pCPE nodes by remaining

resources (assuming merge sort is used). Algorithm 2 has time complexity of  $O(1)$ . Algorithm 3 will always compare the first candidate pCPE node with the cloud and choose the destination with the lower cost, which has the time complexity of  $O(1)$ . Combining the three algorithms, the time complexity of IoT-B&B Algorithm is  $O(n \log(n))$ .

If the exhaustive algorithm is used, which does not sort the candidate pCPE nodes, it would have to check all candidates and find out the one with the lowest cost. Such algorithm would increase the time complexity to  $O(n^2)$ . Figure 5 shows the time consumed using the two different algorithms (Algorithms 2 and 3) with up to 50 pCPE nodes. From the results, we can see that VNF-B&B algorithm scales well compared to the exhaustive algorithm, where the time consumed is less than 1000 ms for 50 nodes, while the exhaustive algorithm takes more than 9000 ms.

**3.4. Access to IoT-B&B Algorithms.** We have implemented the Java version of the IoT-B&B Algorithm library. The library source code is published under the MIT License and is

```

(1) function CLOUDCOST( $f$ )
(2)    $\text{cpu\_left} \leftarrow U(c) + \delta$ 
(3)    $\text{memory\_left} \leftarrow M(c) + \delta$ 
(4)    $\text{link\_bw\_left} \leftarrow B(l_c) + b$ 
(5)    $\text{cost} \leftarrow 0$ 
(6)   for all  $f' \in F_c$  do
(7)      $\text{cpu\_left} \leftarrow \text{cpu\_left} - U(f')$ 
(8)      $\text{memory\_left} \leftarrow \text{memory\_left} - M(f')$ 
(9)      $\text{link\_bw\_left} \leftarrow \text{link\_bw\_left} - B(f')$ 
(10)  end for
(11)   $\text{cost} \leftarrow \text{cost} + W_U U(f) / \text{cpu\_left}$ 
(12)   $\text{cost} \leftarrow \text{cost} + W_M M(f) / \text{memory\_left}$ 
(13)   $\text{cost} \leftarrow \text{cost} + W_B T_d B(f) / \text{link\_bw\_left}$ 
(14)  return  $\text{cost}$ 
(15) end function

(16) function BNBCOST( $f, v$ )
(17)   $\text{cost} \leftarrow 0$ 
(18)  if user of  $f$  does not own  $v$  then
(19)     $\text{cost} \leftarrow \text{cost} + w_U(v)U(f)$ 
(20)     $\text{cost} \leftarrow \text{cost} + w_M(v)M(f)$ 
(21)     $\text{cost} \leftarrow \text{cost} + w_B(v)B(f)$ 
(22)     $\text{cost} \leftarrow \text{cost} \times (1 + \gamma)$ 
(23)  end if
(24)  return  $\text{cost}$ 
(25) end function

```

ALGORITHM 2: IoT-B&amp;B cost estimation algorithm.

downloadable from the following URL: <https://github.com/zhuheec/iot-bnb>.

## 4. System Implementation

We have implemented a system following the architecture illustrated in the previous section. The system provides a platform to practice and evaluate the IoT-B&B algorithm.

**4.1. Hardware Configuration of pCPE Nodes.** For flexibility and scalability, we use VMs instead of bare metal machines as pCPE nodes. Up to 99 VMs are deployed, and each acts as a pCPE node with 8 Cores of CPU, 16 GB of memory, and 40 GB of disk space. Every pCPE node can communicate with any other one via a private virtual network, to mimic that these pCPE nodes are at the same network edge.

**4.2. NFVI Setup.** Each pCPE node has CentOS7 [14] installed as its operating system. It has its essential functionalities running as CPE. We use the OpenStack Kolla Project [15] to deploy OpenStack services across multiple pCPE nodes as well as PE, such that

- (a) the OpenStack services on a pCPE node runs as Docker containers. They can be spun up and torn down with minimal overhead;
- (b) with container-based OpenStack services, a pCPE node can be converted into an OpenStack compute

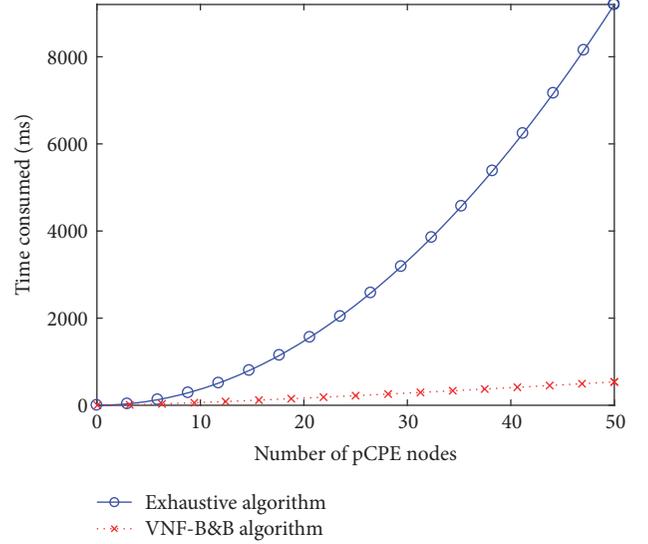


FIGURE 5: Time complexity comparison between IoT-B&B heuristic algorithm and the exhaustive algorithm checking all pCPE candidate nodes.

node and then register to the controller to be one of the available hosts;

- (c) when the pCPE is no longer available to be a compute node due to high usage, the OpenStack services on it can be stopped to free up resources.

Figure 6 reflects the architecture we use to provision IoT-B&B service upon container-based OpenStack.

**4.3. IoT-B&B Algorithm as Filter Scheduler.** To leverage the proposed IoT-B&B algorithm in the system, we implement it as a *filter scheduler* used by nova service, with the name `VnfBnbFilter`. The IoT-B&B algorithm matches the filtering-weighting mechanism of the *filter scheduler*.

Figure 7 shows how IoT-B&B algorithm works as a filter scheduler to rank places to deploy. Suppose there are four places to choose:  $v_1$ ,  $v_2$ ,  $v_3$ , and  $c$ . Algorithm 1 is first invoked to filter out ineligible places that do not have enough resources. In the example,  $v_2$  is filtered out as a result of resource shortage. Then, Algorithms 2 and 3 are used to rank the places according to the cost to deploy the VNF instance. They determine that  $v_3$  has the lowest cost to deploy the instance.

**4.4. System Life Cycle.** We define a set of system states and events for IoT-B&B. The system transitions its state based on the events according to the actual demand, in order to adjust the scaling of the VNF. As Figure 8 shows, the states and events are listed as follows.

**4.4.1. Up State.** The state indicates the system is up and running as expected. System load level is acceptable to satisfy the needs. The system is expected to stay in this state if it runs properly.

```

(1) function CHOOSEPLACE( $f$ )
(2)   candidates  $\leftarrow$  GETSORTEDCANDIDATES( $f$ )
(3)   for all candidate in candidates do
(4)     if candidate is  $c$  then
(5)       current_cost  $\leftarrow$  CLOUDCOST( $f$ )
(6)     else
(7)       current_cost  $\leftarrow$  BNBCOST( $f$ , candidate)
(8)     end if
(9)     if current_cost < cost then
(10)      return place
(11)    end if
(12)  end for
(13)  return none
(14) end function
    
```

ALGORITHM 3: IoT-B&B place selection algorithm.

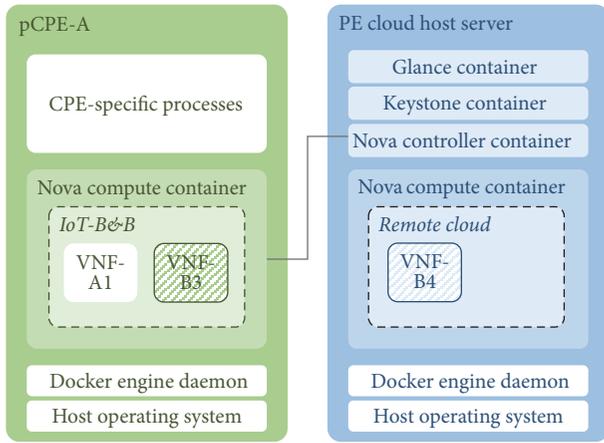


FIGURE 6: IoT-B&B service backed by container-based OpenStack.

4.4.2. *Load Check Event.* This is the event to update the system load level. If the updated load level triggers a change, the system may enter Overloaded or Underloaded state or remain in Up state, depending on the threshold to determine them.

4.4.3. *Overloaded State.* The state indicates the system is overloaded by a higher volume of requests. A scale-out is pending. The placement scheduler will be invoked to determine the place to scale out: B&B or the cloud and then triggers the actual event to scale out.

4.4.4. *Scale-out to B&B Event.* This is the event to trigger a new VM to be deployed on a B&B, that is, a CPE deployment.

4.4.5. *Scale-out to Cloud Event.* This is the event to trigger a new VM to be scaled out in the remote cloud environment.

4.4.6. *Underloaded State.* The state indicates the system is underloaded because of a lower volume of requests. A scale-in is pending. If the number of the VMs has already reached

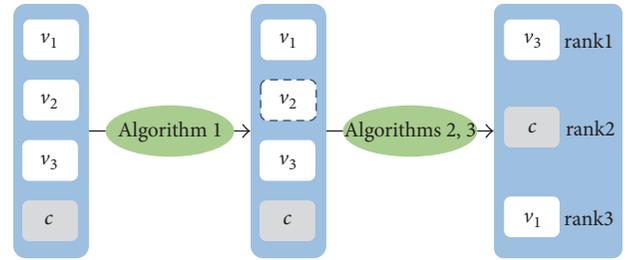


FIGURE 7: IoT-B&B algorithm in OpenStack as a filter scheduler.

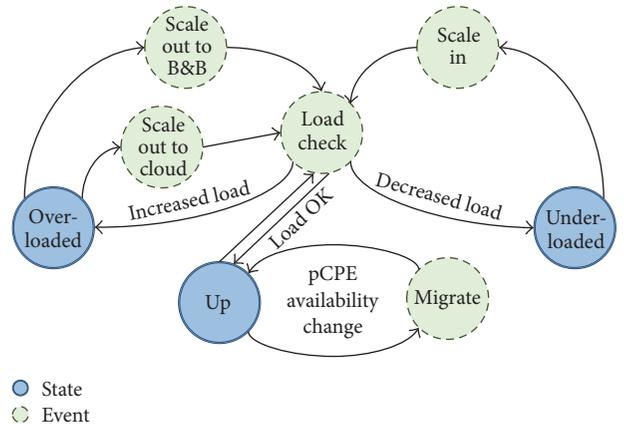


FIGURE 8: IoT-B&B service system life cycle.

the minimum number required, then the system would not enter this state.

4.4.7. *Scale-in Event.* This is the event to trigger an existing VM to be scaled in from either the B&B or the remote cloud.

4.4.8. *Migration Event.* This is the event that is triggered by pCPE availability changes. When a pCPE is no longer capable of hosting a VM because of higher usage from its user, it will cease to be a B&B and be removed from the list of available

TABLE 1: Constant configurations.

Constant	Value
$T_d$	50000 ms
$W_U$	1000
$W_M$	1000
$W_B$	1000
$\delta$	1
$b$	1
$\gamma$	1

hosts. Meanwhile, the migration event will be added to the system to move the existing VMs deployed.

*4.5. Typical Use Cases.* With the definition of the system life cycle, we describe typical uses cases leveraging the IoT-B&B algorithm.

*4.5.1. Launch of New Application.* The dynamic nature of IoT-based services allows the user to launch new applications which are processed by new VNF instances. A new VNF instance does not automatically get deployed on-site, that is, the pCPE node of its user. The reason can range from lack of enough resources to higher cost being deployed on-site. The IoT-B&B algorithm will be called to determine the place to deploy the new VNF instances.

*4.5.2. Scaling out due to Higher Load.* When the user applications have more significant activities, resulting in a higher load of the existing instances, the system's periodical load check daemon will detect the load increase. If the load is above the threshold raising flags of performance, extra VNF instances are needed for processing the larger amount of requests. The IoT-B&B algorithm will be called to determine the place to scale out new VNF instances.

*4.5.3. Migration for Lower Cost.* The VNF instances in the cloud may start with low cost. However, it does not last forever. As more VNF instances are deployed to the remote cloud, fewer resources are available and the unit resource becomes more expensive. At some point, a migration from the cloud to B&B nodes, or vice versa, is reasonable to lower considerable cost.

## 5. Numerical Results

The numerical results based on simulations are shown in this section. From the numerical results, our goals are to verify the benefits of leveraging B&B nodes, compared to using a centralized cloud alone. Table 1 lists the values of the constants used in the algorithms.

*5.1. Host Nodes Setup.* We create 99 pCPE nodes with random levels of initial resources. As all pCPE nodes and the cloud are used to deploy VNF instances, in our configuration, there is a total of 100 nodes for deployments. As seen in

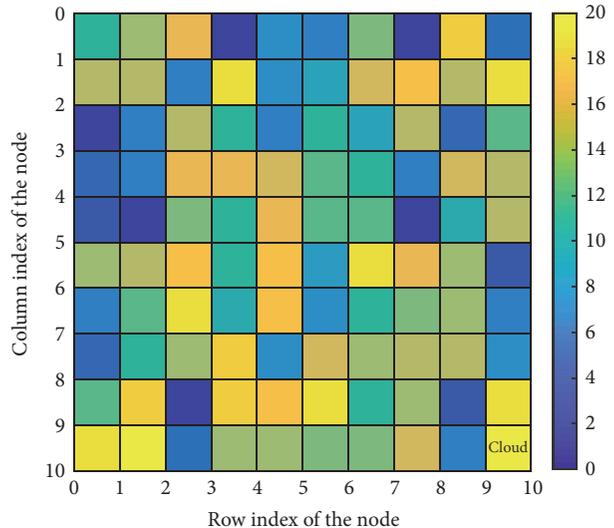


FIGURE 9: Initial resource levels of the 100 nodes used for experiments. The nodes are arranged as a  $10 \times 10$  matrix, where the last element represents the cloud.

Figure 9, the nodes are arranged as a  $10 \times 10$  matrix. Each node is represented by a cell, and is given a horizontal and a vertical coordinate from 1 to 10. The last element, which has the coordinates (10, 10), represents the remote cloud. The colors of the cells reflect the remaining resource levels of the nodes. For the ease of demonstration, the CPU, memory, and bandwidth resources are all broken into 20 levels ranging from 1 to 20. Deploying VNF instances on a pCPE node follows the Law of the Minimum [16], meaning that the capacity of a pCPE node to host instances is determined by its scarcest resource. Therefore, we color the cells according to the resource type of the lowest level of a node. For example, if the remaining CPU level of a node is 20, while the memory level is 3, the cell representing the node will be colored at Level 3. From the initial resource levels, it can be learned that the pCPE nodes have various levels of resources available. Meanwhile, the cloud starts with the maximum level of resources for deployment.

*5.2. VNF Resource Requirement Profile (Flavor) Types.* We predefine 10 types of VNF resource requirement profiles (flavors), as shown in Table 2, with different requirements of resources and max delays allowed. A VNF instance to be deployed will have a flavor from the 10 predefined ones. Templating VNF flavors is based on the real use cases as users will need VNF instances from limited kinds of images for serving known functionalities.

*5.3. Placement Configuration Modes.* In order to compare the effectiveness of the IoT-B&B algorithm, we configure the simulated system to keep deploying new VNF instances of a specific flavor with one of the three modes below, until the resources are depleted:

- (i) Local mode: deploying only on the pCPE node the user owns. The cloud and B&B nodes are not allowed.

TABLE 2: Predefined flavor types for simulation. Resource requirements in units.

Name	CPU	Memory	Bandwidth	Max delay
F1	1	1	1	1000 ms
F2	2	2	2	100 ms
F3	2	2	2	1000 ms
F4	2	2	2	10000 ms
F5	4	4	4	100 ms
F6	4	4	4	1000 ms
F7	4	4	4	10000 ms
F8	8	8	8	100 ms
F9	8	8	8	1000 ms
F10	8	8	8	10000 ms

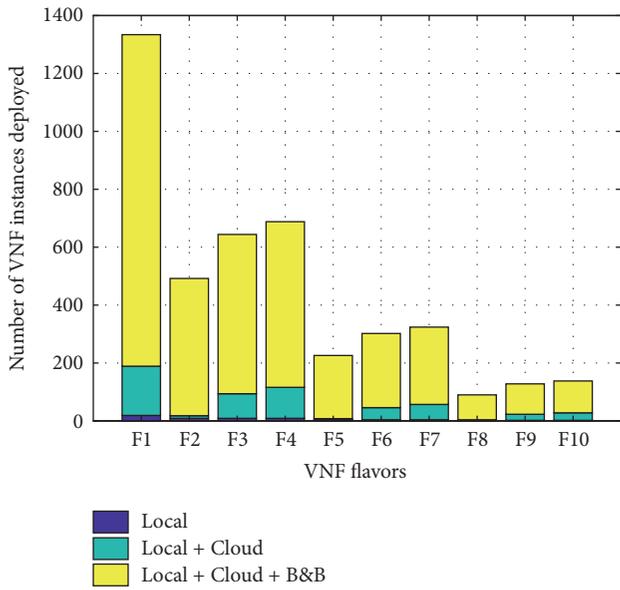


FIGURE 10: Total number of VNF instances deployed for the network edge with various flavors and placement configuration modes.

- (ii) Local + cloud mode: deploying locally on the pCPE node the user owns, as well as on the remote cloud.
- (iii) Local + cloud + B&B mode: local, cloud, and B&B deployments.

5.4. *Extended VNF Instance Capacity.* Figure 10 shows the numbers of instances of deploying each of the 10 predefined flavors with the 3 modes. From the results of Figure 10, we learn that the numbers of instances deployed for all 10 flavors have dramatically increased. Taking F3 as an example, in *Local Mode*, only 9 instances are deployed. When using *Local + Cloud Mode*, the number jumps to 85. For *Local + Cloud + B&B Mode*, the number skyrockets to 550. Therefore, the most beneficial part of the system is to extend the total capacity of hosting VNF instances. If using the remote cloud alone, the capacity of the cloud for VNF instances is limited by the core network delay, even if other resources are assumed to be unlimited. This bottleneck is greatly relieved by the B&B

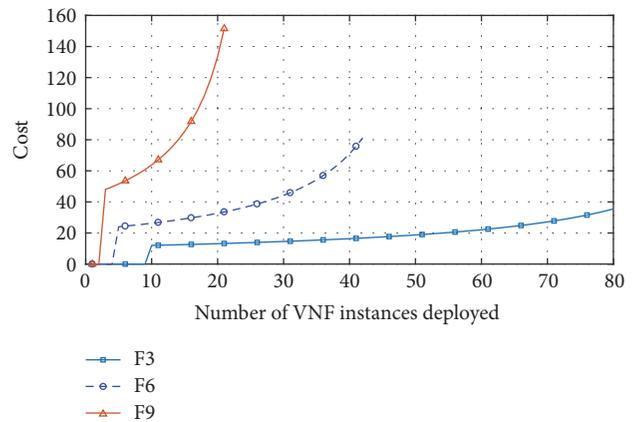


FIGURE 11: Cost hikes when the cloud load increases: *Local + Cloud Mode*.

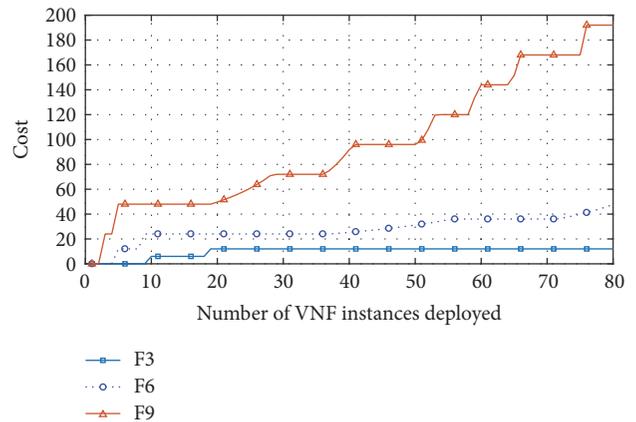


FIGURE 12: Cost hikes when the cloud load increases: *Local + Cloud + B&B Mode*.

nodes hosting instances, because the instances on B&B nodes do not put extra traffic to the core network.

5.5. *Cost Hike by Cloud Load Increase.* We pick the three flavors: F3, F6, and F9, to investigate the trends of cost increase as more VNF instances are deployed in the system. Figures 11 and 12 demonstrate the changes of costs to deploy

a new VNF instance on the cloud in two different modes, as the numbers of deployed instances go up.

Using *Local + Cloud Mode*, the cost is first 0 as the instances are deployed on the local pCPE nodes. As the loads increase, the remote cloud starts to be picked and the cost to deploy an instance increases as the numbers of deployed instances climb. For F6 and F9, the numbers of deployed VNF instances stop at 42 and 21, respectively.

Comparing Figure 12 with Figure 11, in *Local + Cloud + B&B Mode*, the costs are lower when deploying the same numbers of instances in the system. For instance, when deploying 20 instances with flavor F9, the cost using *Local + Cloud Mode* is about 140. Meanwhile, when deploying the same number of instances with the same flavor, the cost under *Local + Cloud + B&B Mode* is only around 50.

The results above have demonstrated the ability of the B&B nodes to redirect the load off the cloud and to reduce the overall cost, even if offering incentives to the users.

**5.6. Impact from outside the Network Edge.** As discussed in Section 2, when the cloud load from outside the network edge gets higher, that is, the value of  $T_d$  is higher, the ability of the cloud hosting VNF instances may be reduced. To verify how much the impact will be, under *Local + Cloud + B&B Mode* and for the three flavors F3, F6, and F9, we increase the level of  $T_d$  by 1 each time and repeat the deployment for 10 times. The numbers of VNF instances deployed are shown in Figure 13. From the results, the capacity of the system is affected by the increase of  $T_d$ . However, the impact becomes less significant as the level of  $T_d$  increases. With the considerable buffer of the B&B nodes, the impact from  $T_d$  is reduced.

**5.7. Remaining Resource Levels.** In *Local + Cloud + B&B Mode*, all B&B nodes participate in hosting VNF instances. We examine the resource levels after the system resources are depleted. When all VNF instances deployed are of flavor F1, the resource levels after the maximum number of instances is deployed are displayed in Figure 14.

The dark colors of all cells indicate that the remaining resource levels are low across all pCPE nodes. The cloud resource levels are also low because of the link/delay bottleneck from the network edge to the core network. The results have demonstrated the ability of the IoT-B&B algorithm to extract the resources to deploy more instances following the best-effort basis.

## 6. Related Work

NFV has been a key role to accelerate usage-based changes and to reduce OPEX by both SPs and vendor. Much of the recent NFV research relies on cloud computing as the underlying infrastructure [17].

By leveraging generic cloud computing Infrastructure-as-a-Service (IaaS) frameworks, such as OpenStack [18] and VMWare [19], research on cloud-based NFV has been done to ensure that VNFs run at optimum levels in the cloud [20]. Soares et al. presented a platform for VNFs called Cloud4NFV [3], which is compliant with the ETSI [9] NFV

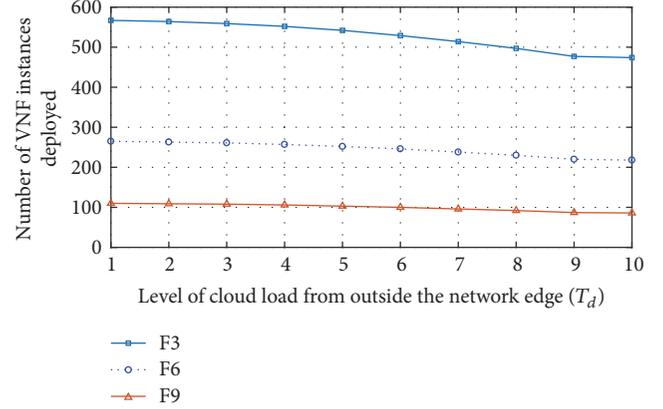


FIGURE 13: Cost changes when the cloud load increases due to tasks outside the target network edge.

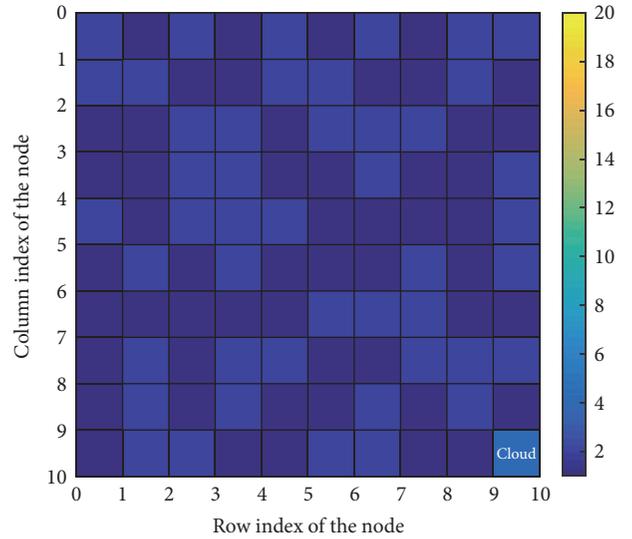


FIGURE 14: Remaining resource levels after all B&B nodes have been used.

architectural specification. Two approaches were discussed to virtualize NF: full virtualization moved all control and user plane functional entities to the cloud, while partial virtualization still forwarded user traffic to physical hardware. With the implementation of service provisioning and end-to-end inventory management, vConductor [21] was presented by Shen et al. that enabled users to plan the virtual network services using its data model. The systems and architectures above focus on deploying VNF instances into the generic cloud infrastructure, rather than the edge of the network.

Due to the nature of varying cost of resources in the cloud, cloud-based resource allocation problems have been studied to reduce the cost and to help evenly distribute the workload. We have analyzed vulnerability of mobile apps in [22] to keep sensitive information in local mobile device, while offloading secured computing-intensive modules to the cloud. Xiao et al. [23] presented a system leveraging virtualization to dynamically allocate resources in datacenter and to optimize the number of servers in use. While these

solutions did help better use cloud resources, they keep the computing remotely in the cloud and will not move it to the edge of the network.

The concept of fog computing was proposed in [7] and was anticipated to become an essential part of cloud computing with the volume of Internet-of-Things (IoT) growing explosively. Vaquero and Rodero-Merino [24] proposed a comprehensive definition of the fog covering its features and impact, including device ubiquity, challenges on service and fog-based network management, levels of device connectivity, and privacy. Edge clouds were presented as entry points for IoT, which could be parts of the Enhanced Packet Core (EPC). The scenarios of fog computing in several domains were discussed in [25], including Smart Grid, IoT, and SDN, with topics about security, privacy, trust, and service migration. The work above has pointed the research direction of leveraging the edge of the network from high levels. Based on fog computing, crowdsourcing becomes an option as fog nodes can be updated dynamically with the participation of the third party. The security and privacy challenges were illustrated in [26], where a general architecture was presented to model crowdsourcing networks, including crowdsourcing sensing and crowdsourcing computing. The security concerns were captured from the characteristics of the architecture.

Virtualization in edge networks as a form of fog computing, including NFV, have been given a close look. Manzalini et al. visioned potential value chain shifts and business opportunities in [27] by emerging paradigms such as SDN and NFV. The paper pictured a massive number of virtualized network and service functions running at the edge of the network, making the processing power more distributed globally. The service chaining in the cloud-based edge networks was analyzed in [28] by programming actions into OpenFlow switches to achieve dynamic service chaining. A platform called Network Functions At The Edge (NetFATE) was proposed in [29] as a proof of concept (PoC) of an NFV framework at the edge of a telco operator networks. Each CPE node was realized with a generic-purpose computer installed with a hypervisor and virtual switches. This made the CPE node capable of deploying VNFs on itself. The focus of this paper was to prove that deploying VNFs on the edge of the network is feasible. However, the benefits of resource sharing across different CPE nodes are not mentioned.

## 7. Conclusions

In this paper, we have presented the architecture and the algorithms to share resources of pCPE nodes across the network edge. When a sharable pCPE node has enough resources, SP will utilize its free resources as a bed-and-breakfast place to deploy VNFs of other users from the same network edge for a certain period. The users can get incentives by allowing SP to leverage the free resources.

By applying the VNF-B&B architecture, the capacity of VNF instances for the network edge is greatly increased. The cost of offloading to the centralized cloud is reduced. By keeping the VNFs at the network edge, the delay is reduced for better processing of real-time data burst from IoT devices.

Meanwhile, the traffic load to the core network is substantially reduced with the same number of VNF instances deployed.

Making better use of the network edge is an interesting topic and has a massive potential. While the paper ends here, we are continuing to beef up the architecture, including the following:

- (i) Explore the availability to factor in the service up and down of B&B nodes. This paper has used a constant factor to model the backup VNF instances. The modeling can be improved so that it is closer to the real-world scenario.
- (ii) Consider more factors impacting the deployment placement besides vCPUs, memory, and network bandwidth. Also, consider detailed factors that can indirectly impact the cost and the core network delay of the remote cloud.

Future work of this paper will consider the items listed above with the aim of obtaining a practical and effective framework of virtualizing and utilizing the network edge.

## Notations Used in Problem Formulation

$v, v_i, n_V$ :	$v$ is a pCPE node. $v_i$ is a specific pCPE node by its index, where $i \in [1 \cdots n_V]$ . $n_V$ is the total number of pCPE nodes in the network edge
$c$ :	The remote cloud location to deploy
$u, u_i$ :	$u$ is a user. Each user owns one pCPE node. $u_i$ is a specific user by the pCPE index, where $i \in [1 \cdots n_V]$
$l_{ij}, l_c, R(l_c)$ :	$l_{ij}$ is the link between pCPE nodes $v_i$ and $v_j$ , where $i, j \in [1 \cdots n_V]$ , $i \neq j$ . $l_c$ is the link between the network edge and the core network. $R(l_c)$ is the remaining network bandwidth of $l_c$
$a, n_a$ :	$a$ is a VNF type/ flavor. $n_a$ is the total number of VNF types
$f, f(a_k)$ :	$f$ is a VNF instance. $f(a_k)$ is an instance of type $a_k$ , $k \in [1 \cdots n_a]$
$f(a_k, u_i)$ :	A VNF instance of type $a_k$ and user $u_i$ , $k \in [1 \cdots n_a]$ , $i \in [1 \cdots n_V]$
$f(a_k, u_i, v_j)$ :	A VNF instance of type $a_k$ , user $u_i$ , and deployed on pCPE node $v_j$ , $k \in [1 \cdots n_a]$ , $i, j \in [1 \cdots n_V]$
$F_v, F_c$ :	$F_v$ is the set of all VNF instances deployed on pCPE node $v$ . $F_c$ is the set of all VNF instances deployed on the cloud
$U(v), M(v), B(v)$ :	The number of vCPUs, the amount of memory, and the network bandwidth capacity that can be provided by the pCPE node $v$ , respectively

$U(f), M(f), B(f)$ : The number of vCPUs, the amount of memory, and the network bandwidth required by  $f$ , respectively

$t(f), T_{\max}(f), T_d$ :  $t(f)$  is the core network delay by offloading  $f$  to the cloud.  $T_{\max}(f)$  is the maximum delay allowed by  $f$ .  $T_d$  is the core network delay not caused by the network edge

$b, \delta$ :  $b$  is the maximum core network delay when the bandwidth of  $l_c$  is depleted.  $\delta$  is a very small positive number used as part of the denominators in (9), (10), and (12) to avoid dividing by 0

$S(f, c), S(f, v)$ : Cost of  $f$  deployed on the cloud  $c$  and the pCPE node  $v$ , respectively

$X(f, c), X(f, v)$ : They equal 1 if  $f$  is on  $c/v$  and 0 if not.

## Conflicts of Interest

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

## Acknowledgments

The authors would like to thank Yiwen Wang who kindly reviewed the manuscript of this paper and shared comments.

## References

- [1] Customer premises equipment (CPE), 2017, <http://www.thenet-workencyclopedia.com/entry/customer-premises-equipment-cpe/>.
- [2] S. Beereddy and K. Sirupa, "NFV use case—delivering virtual CPE with multi-vendor VNF orchestration," in *Proceedings of the IEEE Conference on Network Function Virtualization and Software Defined Network (NFV-SDN '15)*, pp. 25–27, San Francisco, Calif, USA, November 2015.
- [3] J. Soares, M. Dias, J. Carapinha, B. Parreira, and S. Sargento, "Cloud4NFV: a platform for Virtual Network Functions," in *Proceedings of the 3rd IEEE International Conference on Cloud Networking (CloudNet '14)*, pp. 288–293, Luxembourg, Luxembourg, October 2014.
- [4] E. Telecom, White Paper: The Definitive Guide to vCPE, 2017, <https://www.ecitele.com/media/1703/white-paper-the-definitive-guide-to-vcpe.pdf>.
- [5] T. Taleb, M. Corici, C. Parada et al., "EASE: EPC as a service to ease mobile core network deployment over cloud," *IEEE Network*, vol. 29, no. 2, pp. 78–88, 2015.
- [6] N. Herbaut, D. Negru, G. Xilouris, and Y. Chen, "Migrating to a NFV-based Home Gateway: Introducing a Surrogate vNF approach," in *Proceedings of the 6th International Conference on the Network of the Future (NOF '15)*, pp. 1–7, Montreal, Canada, October 2015.
- [7] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the 1st Mobile Cloud Computing Workshop on Mobile Cloud Computing (MCC '12)*, pp. 13–16, Helsinki, Finland, August 2012.
- [8] M. Satyanarayanan, "The emergence of edge computing," *The Computer Journal*, vol. 50, no. 1, pp. 30–39, 2017.
- [9] ETSI Industry Specification Group (ISG) NFV, ETSI GS NFV 002 V1.2.1: Network Functions Virtualisation (NFV): Architectural Framework, 2014.
- [10] K. Jang, J. Sherry, H. Ballani, and T. Moncaster, "Silo: predictable message latency in the cloud," in *Proceedings of the ACM Conference on Special Interest Group on Data Communication (SIGCOMM '15)*, pp. 435–448, London, UK, August 2015.
- [11] OpenStack: Overcommitting CPU and RAM, 2017, <https://docs.openstack.org/arch-design/design-compute/design-compute-overcommit.html>.
- [12] S. A. Baset, L. Wang, and C. Tang, "Towards an understanding of oversubscription in cloud," in *Proceedings of the 2nd USENIX Conference on Hot Topics in Management of Internet, Cloud, and Enterprise Networks and Services (Hot-ICE '12)*, San Jose, Calif, USA, 2012.
- [13] Amazon EC2, 2017, <https://aws.amazon.com/ec2/>.
- [14] The CentOS Project, 2017, <https://www.centos.org/>.
- [15] OpenStack Kolla Project, 2017, <https://wiki.openstack.org/wiki/Kolla/>.
- [16] A. N. Gorban, L. I. Pokidysheva, E. V. Smirnova, and T. A. Tyukina, "Law of the minimum paradoxes," *Bulletin of Mathematical Biology*, vol. 73, no. 9, pp. 2013–2044, 2011.
- [17] R. Jain and S. Paul, "Network virtualization and software defined networking for cloud computing: a survey," *IEEE Communications Magazine*, vol. 51, no. 11, pp. 24–31, 2013.
- [18] OpenStack, 2017, <http://www.openstack.org/>.
- [19] VMWare, 2017, <http://www.vmware.com/>.
- [20] R. Mijumbi, J. Serrat, J. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network function virtualization: state-of-the-art and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 236–262, 2016.
- [21] W. Shen, M. Yoshida, T. Kawabata, K. Minato, and W. Imajuku, "VConductor: an NFV management solution for realizing end-to-end virtual network services," in *Proceedings of the 16th Asia-Pacific Network Operations and Management Symposium (APNOMS '14)*, pp. 1–6, Hsinchu, Taiwan, September 2014.
- [22] H. Zhu, C. Huang, and J. Yan, "Vulnerability evaluation for securely offloading mobile apps in the cloud," in *Proceedings of the IEEE 2nd International Conference on Cloud Networking (CloudNet '13)*, pp. 108–116, San Francisco, Calif, USA, November 2013.
- [23] Z. Xiao, W. Song, and Q. Chen, "Dynamic resource allocation using virtual machines for cloud computing environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 6, pp. 1107–1117, 2013.
- [24] L. M. Vaquero and L. Rodero-Merino, "Finding your way in the fog: towards a comprehensive definition of fog computing," *ACM SIGCOMM Computer Communication Review Archive*, vol. 44, no. 5, pp. 27–32, 2014.
- [25] I. Stojmenovic and S. Wen, "The fog computing paradigm: scenarios and security issues," in *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS '14)*, pp. 1–8, IEEE, Warsaw, Poland, September 2014.
- [26] K. Yang, K. Zhang, J. Ren, and X. Shen, "Security and privacy in mobile crowdsourcing networks: challenges and opportunities," *IEEE Communications Magazine*, vol. 53, no. 8, pp. 75–81, 2015.
- [27] A. Manzalini, R. Minerva, F. Callegati, W. Cerroni, and A. Campi, "Clouds of virtual machines in edge networks," *IEEE Communications Magazine*, vol. 51, no. 7, pp. 63–70, 2013.

- [28] F. Callegati, W. Cerroni, C. Contoli, and G. Santandrea, "Dynamic chaining of virtual network functions in cloud-based edge networks," in *Proceedings of the 1st IEEE Conference on Network Softwarization (NETSOFT '15)*, IEEE, London, UK, April 2015.
- [29] A. Lombardo, A. Manzalini, G. Schembra, G. Faraci, C. Rametta, and V. Riccobene, "An open framework to enable NetFATE (Network Functions at the edge)," in *Proceedings of the 1st IEEE Conference on Network Softwarization (NETSOFT '15)*, pp. 1–6, London, UK, April 2015.

## Research Article

# An SAT-Based Method to Multithreaded Program Verification for Mobile Crowdsourcing Networks

Long Zhang <sup>1</sup>, Wanxia Qu <sup>1</sup>, Yinjia Huo,<sup>2</sup> Yang Guo <sup>1</sup> and Sikun Li<sup>1</sup>

<sup>1</sup>College of Computer, National University of Defense Technology, Changsha, China

<sup>2</sup>The University of British Columbia, Vancouver, BC, Canada V6T 1Z4

Correspondence should be addressed to Wanxia Qu; quwanxia@nudt.edu.cn

Received 23 September 2017; Revised 24 December 2017; Accepted 1 January 2018; Published 28 January 2018

Academic Editor: Edith Ngai

Copyright © 2018 Long 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.

This paper focused on the safety verification of the multithreaded programs for mobile crowdsourcing networks. A novel algorithm was proposed to find a way to apply IC3, which is typically the fastest algorithm for SAT-based finite state model checking, in a very clever manner to solve the safety problem of multithreaded programs. By computing a series of overapproximation reachability, the safety properties can be verified by the SAT-based model checking algorithms. The results show that the new algorithm outperforms all the recently published works, especially on memory consumption (an advantage that comes from IC3).

## 1. Introduction

The mobile crowdsourcing network is a promising network architecture to perform tasks with human involvement and numerous mobile devices but suffers from security and privacy concerns [1, 2]. Pthread-style multithreaded programs play an important role in crowdsourcing computing [3, 4] and crowdsourcing sensing [5, 6] for supporting concurrent programming. Multithreaded programming used much existing system-level code such as device drivers, operating system, and distributed computing. The mobile crowdsourcing networks [7, 8] distribute tasks and collect the results and must make sure safely access for the shared data. Therefore, it is important to verify the safety properties for multithreaded programs.

In this paper, we consider the multithreaded programs with an unbounded number of threads. Each thread executes a finite, nonrecursive state machine. Mutexes, which can be expressed using Boolean variables, are used for synchronization in this kinds of multithreaded programs. The illegal access of mutexes will lead to safety problem. We assume all mutexes are shared variables. The safety problems is to verify if the illegal access of mutexes exists.

The safety property for multithreaded programs which was expressed as upward-closed sets of the target (“bad”) states can be verified by reduction to the coverability problem

of *well-structured transition systems* (WSTS) [9, 10]. WSTS are a very broad class of infinite-state systems, including thread transition system (TTS) [11], Petri nets and their monotonic extensions [12–16], broadcast protocols [17, 18], lossy channel systems [19] and context-free grammars [10]. Coverability is the base verification task for WSTS: the question is whether the system can reach an unsafe or illegal configuration among some subset of its (possibly unbounded number of) components. There have been several algorithms published for WSTS coverability problem [9–11, 20–22], but none perform as efficiently as finite state model checking.

The IC3 algorithm [23] is an SAT-based model checking algorithm and introduced as an efficient technique for safety properties verification of finite state systems, especially in hardware verification. It computes an inductive invariant by maintaining a sequence of overapproximation of reachability from initial states and strengthens them incrementally. An efficient implementation of the procedure shows good performance on hardware benchmarks [24].

This paper focuses on the multithreaded programs for mobile crowdsourcing networks. All multithreaded programs are pthread-style ANSI-C source code and transformed into TTS by using predicate abstraction [25, 26]. We introduce a novel, highly efficient algorithm for the coverability problem of TTS. The new algorithm is to find a way to apply conventional, finite state IC3, which is typically

the fastest algorithm for finite state model checking, in a very clever manner to solve the coverability problem. IC3 algorithm is a finite state model checking algorithm, and the original input is an FSM. We try to use the finite state model checker to solve the infinite-state systems. The bounded TTS is transformed into an FSM and described as the inputs format for IC3 engine. The significant contributions of this paper are as follows:

- (1) Our approach requires very novel and intricate reasoning because IC3 produces a series of overapproximation reachability results. A novel algorithm which is based on IC3 engine is proposed to solve the coverability problem of TTS.
- (2) We introduce new encoding techniques to make the verification of infinite-state systems possible by using finite state algorithms.
- (3) We implement tools combination, which is a good way to improve the total rate of successfully solved instances.

The experimental results show that our new algorithm outperforms all the recently published works, uses far less memory (an advantage that comes from IC3), and can solve more benchmarks successfully. The new method can solve 97.2% instances within 1 GB.

The rest of this paper is organized as follows. In Section 2, we review the related work. Section 3 presents necessary preliminaries used in this paper. In Section 4, we propose our new method based on IC3 and give more details of the implementation. Section 5 shows the experimental evaluation on multithreaded programs. Section 6 concludes this paper and discusses future works.

## 2. Related Works

A general decidability result showed that the coverability problem is decidable for WSTS [9], which backward-explore states starting from the target states. Bingham and Hu [20] proposed a new algorithm to compute fix-points over a series of finite state systems of increasing size. A new subclass of WSTS, named Nice Sliceable WSTS, was introduced. Starting from the target states, it computed the exact backward reachability by using finite state symbolic model checking [27] based on BDDs [28] to solve the coverability problem of NSW. Kaiser et al. [11, 22] introduced a new algorithm to solve the safety properties of multithreaded programs with an unbounded number of threads executing a finite state, nonrecursive procedure. By using many inexpensive uncoverability proofs, this new approach combined forward propagation under-approximations with backward propagation of overapproximations to the coverability problem in TTS. Inspired by the success of IC3 algorithm in finite state model checking, Kloos et al. [21] proposed an incremental, inductive procedure to check coverability of downward-finite WSTS, which contains Petri nets, broadcast protocols, and lossy channel systems. All those algorithms are based on the backward reachability and suffered from complex computational consumption. Esparza et al. [29] introduced an incomplete but empirically efficient solution to the coverability

problem. The new approach was based on classical Petri nets analysis techniques, the marking equation and traps [30, 31], and utilized an SMT solver to implement the constraint approach. Inspired by Esparza's work, Athanasiou et al. [32] introduced an approximate coverability method by using thread state equations and implemented it in a tool named TSE. TSE is very capable on Boolean programs but theoretically incomplete.

## 3. Preliminaries

### 3.1. Nicely Sliceable WSTS

*Definition 1* (well-quasi-ordering). A well-quasi-ordering (wqo) is a reflexive and transitive binary relation  $\leq$  over set  $X$ , and for every infinite sequence  $x_0, x_1, x_2, \dots$  of elements from  $X$ , there exists  $i < j$  such that  $x_i \leq x_j$ .

For  $Y \subseteq X$ , the upward-closure of  $Y$  is the set  $\uparrow Y = \{x \mid \exists y \in Y, y \leq x\}$ . A basis of an upward-closed set  $Y$  is a set  $B \subseteq Y$  such that  $Y = \uparrow B$ . A set  $U$  is said to be  $\leq$ -upward-closed (or simply upward-closed if  $\leq$  is clear from the context) if  $U = \uparrow U$ . It is known that if  $\leq$  is a wqo, then any  $\leq$ -upward-closed set has a unique finite basis  $B$  such that for all  $x, y \in B$  we have  $x \not\leq y$  and  $y \not\leq x$  [33]. Given upward-closed  $U$ , we let  $\text{basis}(U)$  denote the unique finite basis of  $U$ . Moreover, it is known that any infinite increasing sequence  $S_0 \subseteq S_1 \subseteq S_2 \subseteq \dots$  of upward-closed sets eventually stabilizes; that is, there exists  $k \in \mathbb{N}$  such that  $S_k = S_{k+1} = S_{k+2} = \dots$ .

*Definition 2* (discrete wqo). A wqo is a discrete wqo (dwqo) over  $X$  if for all  $x \in X$  there exists  $k \in \mathbb{N}$  such that for any sequence  $x_0 < x_1 < \dots < x_l = x$ , we have  $l \leq k$ . The weight function  $w : X \rightarrow \mathbb{N}$  maps each  $x$  to the minimum such as  $k$ . For the  $\leq$ -upward-closed set  $U$ , the base weight of  $U$  is  $\text{bw}(U) = \max\{w(x) \mid x \in \text{basis}(U)\}$ .

The weight function  $w(x)$  slices the state space into a countable number of finite sets  $S_0, S_1, S_2, \dots$ , where  $S_i = \{x \in S \mid w(x) = i\}$ . This property allows for finite state model checking techniques to be used to the reachability for each weighted bounded  $S_i$ .

*Definition 3* (nicely sliceable well-structured transition systems). A nicely sliceable well-structured transition system (NSW)  $M = (S, \rightarrow, \leq)$  is a transition system equipped with a dwqo on its states that satisfies the following properties:

- (1)  $S$  is the (possibly infinite) state space.
- (2)  $\rightarrow \subseteq S \times S$  is transition relation.
- (3)  $\leq$  is a dwqo over  $S$ .
- (4) For all  $x, x', y \in S$ , if  $x \rightarrow x'$  and  $x \leq y$ , there exists  $y'$  such that  $y \rightarrow y'$  and  $x' \leq y'$ .
- (5) Weight-respecting: for all  $x, x', y \in S$ ,  $w(x') - w(x) = w(y') - w(y)$ .
- (6)  $\delta$ -deflatable: for  $\delta \in \mathbb{N}$  if whenever  $x \rightarrow x'$  and  $z \leq x'$ , there exists  $y$  and  $y'$  such that the following properties hold: (1)  $y \leq x$ , (2)  $y \rightarrow y'$ , (3)  $z \leq y'$ , (4)  $w(y) \leq w(z) + \delta$ , and (5)  $w(y') \leq w(z) + \delta$ .

**3.2. Thread Transition Systems.** Thread transition systems (TTS) are motivated by the verification task of multithread asynchronous programs, which is the subset of NSW. Let  $L$  and  $S$  be finite sets for *local* and *shared* states, respectively. The elements of  $T = S \times L$  are called *thread states*.

**Definition 4** (thread transition system). A thread transition system (TTS) is a pair  $(T, R)$ , where  $R \subseteq T \times T$  is a binary relation on  $T$ , partitioned into  $R \Rightarrow \cup \Leftarrow$ .

Let  $V = \bigcup_{n=1}^{\infty} (S \times L^n)$ . The elements of  $V$  are called *states*. We write them in the form  $(s \mid l_1, l_2, \dots, l_n)$ . A TTS gives rise to a transition system  $M = (V, \rightarrow)$  with

$$(s \mid l_1, l_2, \dots, l_n) \rightarrow (s' \mid l'_1, l'_2, \dots, l'_n), \quad (1)$$

if one of the following conditions holds.

**Thread Transitions.**  $n' = n$  and there exists  $(s, l) \mapsto (s', l') \in R$  and  $i$  such that  $l_i = l, l'_i = l'$ , and, for all  $j \neq i, l'_j = l_j$ .

**Spawn Transitions.**  $n' = n+1$  and there exists  $(s, l) \Leftarrow (s', l') \in R$  and  $i$  such that  $l_i = l, l'_i = l'$ , and for all  $j < n', l'_j = l_j$ .

Let  $I_l \subseteq L$  be a set of initial local states and  $I_s \subseteq S$  be a set of initial shared states. We define the set of initial states to be  $I = I_s \times \bigcup_{n=1}^{\infty} I_l^n$ . An execution of the transition system  $M$  is a finite or infinite sequence of states in  $V$  whose adjacent states are related by  $\rightarrow$ , which started at initial states. A state is reachable if it appears in some execution.

In order to state the coverability problem, define the *cover* relation  $\preceq$  over  $V$  as  $(s \mid l_1, l_2, \dots, l_n) \preceq (s' \mid l'_1, l'_2, \dots, l'_n)$  if  $s = s'$  and  $[l_1, l_n, \dots, l_n] \subseteq [l'_1, l'_2, \dots, l'_n]$ , where  $[\cdot]$  denotes a *multiset*.

Give target states  $v_F \in V$ , if  $v_F$  is coverable; that is, does there exists a path in  $M$  leading to a state  $v$  that covers  $v_F$ :  $v \succeq v_F$ ? The safety property is described as the upward-closed set of  $v_F$  and converts into the coverability analysis problem.

A *cover* relation  $\preceq$  is neither symmetric nor antisymmetric, thus a quasi-order, and in fact a well-quasi-order (wqo) on  $V$ : any infinite sequence  $v_1, v_2, \dots$  of elements from  $V$  contains an increasing pair  $v_i \preceq v_j$  with  $i < j$ . It is easy to see that  $(M, \preceq)$  fulfills the definition of WSTS. A TTS with standard thread and spawn transition can be expressed as plain Petri nets [22] and is the subset class of NSW [20].

## 4. Multithreaded Programs Safety Verification

In this section, we introduce a new method for the safety verification of multithread programs. The input source code is translated into TTS by using SATABS [34]. Then, we propose a novel TTS coverability analysis algorithm to verify the safety properties. Finally, the implementation details are described.

**4.1. The Input Languages.** Most popular programming languages such as Java and C/C++ embrace concurrent programming via their *pthread* or *thread* class APIs, respectively. In this paper, we focus on the pthread-style multithreaded ANSI-C programs. ANSI-C is one of the most popular programming languages for safety critical embedded software.

The mobile crowdsourcing networks contains most embedded devices which are based on multithreaded programs to support the crowdsourcing computing and communications. SATABS is an SAT-based model checker by predicate abstraction, and can be used to model the ANSI-C programs into TTS format. We follow the introduction from the SATABS website (<http://www.cprover.org/satabs/>) to transform the ANSI-C programs into TTS. All ANSI-C programs can be translated into Boolean programs by SATABS, completely. The safety properties can be reserved during the formalization process.

**4.2. IC3-Based Thread Transition System Coverability Analysis Algorithm.** IC3 is SAT-based and computes inductive overapproximations of reachable sets. Let  $I$  and  $P$  be initial states and the property states, respectively. Also let  $T$  denote the transition relation over the current and the next states. IC3 maintains a trace:  $[R_0, R_1, \dots, R_N]$ . The first element  $R_0$  is the initial states. For  $i > 0, R_i$  is a set of clauses that AND-ed together and represent an overapproximation of the states reachable from the initial states in  $k$  steps or less.  $R_i \rightarrow R_{i+1}$ , and the clauses  $R_{i+1}$  are a subset of  $R_i$ , except for  $i = 0$ . The IC3 algorithm will terminate if a counterexample is found or an inductive proof  $R_N$  is got.

This section develops our new algorithm, TTSCov, which is based on the IC3 algorithm. For a target set  $v_F$  and  $i \in \mathbb{N}$ ,  $wl(v_F, i)$  presents the set of weight limited  $v_F$  by  $i$ . From the base weight of  $v_F$ , the algorithm computes the overapproximation for the backward reachable set  $O_i$ .  $O_i$  is an inductive overapproximation of the states from  $v_F$  which is reachable along a path that never exceeds weight  $i$ . We use IC3, an SAT-based finite state model checker, to compute this weight limited and inductive overapproximation of  $v_F$ .

As shown in Algorithm 1, the input is a TTS  $M$ , a set of initial states  $I$ , and an  $\preceq$ -upward-closed set of target states  $v_F$ . The variable  $i$  is the current weight boundary, which is initially the base weight of  $v_F$  and increases by 1 each loop iteration.  $O_i$  is an overapproximation of  $br(v_F, i)$ , which initially is set as  $wl(v_F, bw(v_F))$ .  $O_i$  is an overapproximation of  $O_{i-1}$  bounded by the weight  $i$ , which is computed by IC3 engine. If  $O_i$  intersected with the initial states  $I$ , the counterexample was found, and the algorithm terminated. In line (6), we check if  $O_i$  and  $O_{i-1}$  are equal, if not, the variable  $n$  was assigned as the current  $i$ . If the condition of line (6) fails  $\delta$  times consecutively, we have  $O_{n+\delta} = O_{n+\delta-1} = \dots = O_n$ , and thus the verification is successful.

Actually, the condition in line (8) can be replaced by  $O_i \neq O_{i-1}$ , and the algorithm works well under the syntactically equal check. But the syntactic checking is more efficient, as the IC3 algorithm builds the frame incrementally, and the clause in  $O_i$  is the subset in  $O_{i-1}$ . In order to take this feature, a Boolean variable can be used to check if new clauses add to  $O_i$  when computing the overapproximation at line (3). This speeds up the algorithm much more.

The main routine in Algorithm 1 is the while-loop. For each loop, the IC3 engine computes the overapproximation by using the SAT solver. The algorithm terminates when it finds a counterexample at line (5) or proves safety at line (13).

```

Input:
   $M = (S, \rightarrow, \preceq)$ : a TTS model to describe a multi-threaded program
   $I \subseteq S$ : the initial states
   $\delta$ : a constant which depend on the input model  $M$ 
   $v_F$ : the target states, which is described as an upward-closed set
Output:
  safe or unsafe
(1)  $i := bw(v_F), n := i, O_{i-1} := wl(v_F, i)$  // initialization
(2) while  $n \geq i - \delta$  do
(3)    $O_i := IC3(O_{i-1}, I, i)$ 
(4)   if  $O_i \cap I \neq \emptyset$  then
(5)     print CEX
(6)     RETURN unsafe
(7)   end if
(8)   if  $O_i \neq O_{i-1}$  then
(9)      $n := i$ 
(10)  end if
(11)   $i := i + 1$ 
(12) end while
(13) RETURN safe

```

ALGORITHM 1: TTSCov.

**4.3. Implementation.** A TTS with thread and spawn transitions is expressive as a plain Petri net. Zhang et al. [35] introduced a method to cut off a Petri net into a finite state machine (FSM). Inspired by Zhang’s work, this section introduces the details of how to bound the TTS into FSM.

*The TTS Format.* The input multithreaded programs model are encoded in the TTS format (<http://www.cprover.org/bfc/>). Each *shared* or *local* is mapped to a *shared/local* variable. Just one *shared* variable can be assigned to “1,” and all *local* variables could be assigned to arbitrary natural number. A transition is a thread transition or spawn transition, which described how the thread state changed.

*The FSM Format: AIGER.* AIGER is a format, library and set of utilities for And-Inverter Graphs (AIGs) (<http://fmv.jku.at/aiger/>). AIGs are a good way to describe a FSM and can be translated into a propositional logic for a SAT solver. The bounded TTS is encoded as an AIG model, where each *shared* and *local* state corresponds to state variables. Extra input variables are introduced to select which rule to be fired, and then update the state variables’ value to set up the transition relations equally.

*Shared and Local Variables.* An  $|S|$ -bit vector was used to encode all *shared* variables, as just one shared variable was assigned with “1” at the same time. For each *local* variable, the unary encoding was used to encode the natural numbers  $n$ , as literature [35] shows that one-hot encoding is one possible unary encoding. As shown in Table 1, thermometer encoding is another unary encoding and performs well when using the incremental SAT solvers. In this paper, thermometer encoding is used to encode the *local* variables.

A full adder is used to bound the total thread number, and the logic is the same as described in [35]. The total thread number is the sum of all *local* variables’ value. As

TABLE 1: Encoding: binary, one-hot, and thermometer.

$n$	Binary	One-hot	Thermometer
0	000	00000001	00000001
1	001	00000010	00000011
2	010	00000100	00000111
3	011	00001000	00001111
4	100	00010000	00011111
5	101	00100000	00111111
6	110	01000000	01111111
7	111	10000000	11111111

the thermometer encoding technique is used to encode the local variables, the structure information is also added as the constraint to the AIG model.

## 5. Experimental Evaluation

We have implemented our algorithm in a tool named TTSCov. TTSCov is implemented with C++, and all input instances are encoded in TTS format. Petri nets tools are used by converting TTS instances into MIST format (<https://github.com/pierreganty/mist>). Most crowdsourcing programs are described in pthread-style multithreaded ANSI-C. SATABS is the front-end of TTSCov, which translates the input ANSI-C programs in to TTS.

To measure TTSCov’s performance, we compare with the state-of-the-art tools: MIST, IIC [21], BFC [11], Petrinizer [29], and TSE [32]. All experiments are performed on an Intel 3.4 GHz Intel, and 16 GB of memory, running Linux OS in 64-bit. The CPU time is limited to 1 hour, and memory to 10 GB.

**5.1. Benchmarks.** We collect 178 Petri nets examples from the Petrinizer repository (<https://github.com/cryptica/pnerf>), in

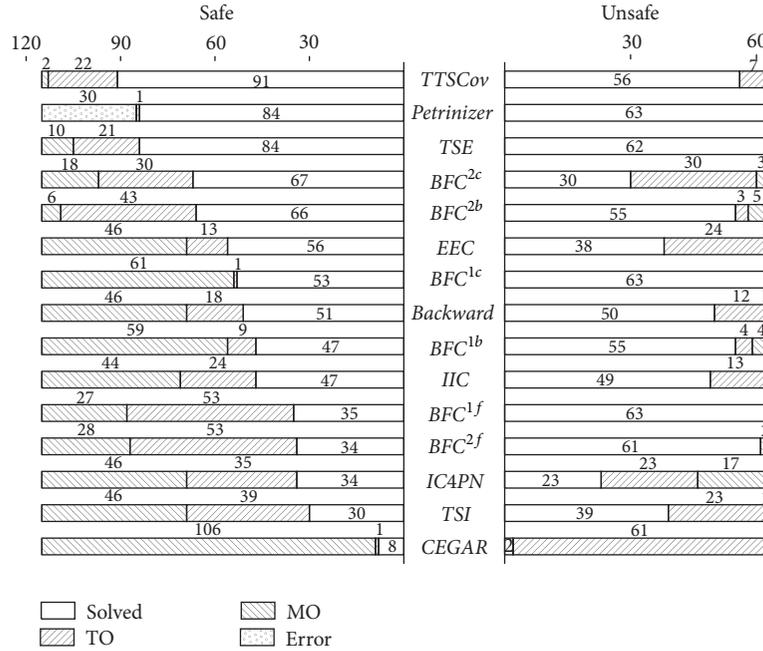


FIGURE 1: Total instances. All 178 instances are separated into safe and unsafe by the real results. There are 115 safe instances, and 63 unsafe instances. TTSCov compare with the state-of-the-art tools: MIST, IIC, BFC, Petrinizer, and TSE. There are two versions of BFC tool, and for each version, the tool has three modes. BFC<sup>2c</sup> means the BFC v2.0 run in concurrent mode. *f* and *b* respect forward and backward, respectively. MIST toolkit implemented five algorithms: *Backward*, *EEC*, *TSI*, *ICAPN*, and *CEGAR*.

which 115 instances are safe and 63 instances are unsafe. All examples are organized into five suites. The first suite is a collection of plain Petri nets from the MIST toolkit. This suite contains 23 Petri nets (17 instances are safe, and the rest are unsafe) and 6 bounded Petri nets which are all safe. The second suite comes from the provenance analysis of message in a medical system, which contains 12 safe instances. The third suite also comes from the provenance analysis of message in bug tracking application, which contains 40 safe instances and 1 unsafe instance. The fourth suite contains 46 instances that are used to evaluate the BFC tool (<http://www.cprover.org/bfc>). Those instances are generated from concurrent C program in TTS format. They are mostly unsafe, and just 2 instances are safe. The fifth suite contains 50 instances that comes from the Erlang verification tool called Soter [36], and those examples can be found on Soter's Website (<http://mjolnir.cs.ox.ac.uk/soter>). Out of 50 instances in this suite, 38 are safe. This suite contains the largest example in the collection, with 66,950 places and 213,635 transitions.

**5.2. Rate of Success on All Instances.** We run two different version of BFC, named BFC v1.0 and BFC v2.0, respectively. The BFC tool has 3 modes: backward, forward, and concurrent. In concurrent mode, two threads are used to run backward and forward parallel. But we find there is a potential bug in BFC v2.0 with concurrent mode. The tool got stuck in concurrent mode when switching the thread. We run 10 times in concurrent mode and show the median results in Figure 1. Petrinizer is an SMT-based tool, which has 8 parameters, and the tool may return wrong answer for some examples. If

Petrinizer returns wrong result in one configuration, we say the result is wrong. All five algorithms in MIST toolkit were compared.

Figure 1 shows that TTSCov performs better than the other complete tools, which solve 147 instances in total the same as Petrinizer, which is incomplete. For safe instances, TTSCov solves 91 out of 115, and 22 instances timeout. Most importantly, there are just 2 instances over the memory limit, which support the IC3 less memory usage. Petrinizer solves 84 instances, and 1 timeout. But 30 instances return unsafe results for those safe instances. Out of 30 wrong results examples, 19 instances return wrong answer with all 8 configurations, and the others are partly wrong. TSE solves 84 instances, 10 out of memory and 21 timeout. BFC<sup>2c</sup> solves 67 examples in total, whereas 30 examples time out and 18 instances are out of memory. BFC<sup>2b</sup> performs same as BFC<sup>2c</sup>, in which 66 instances are solved, but with more out of time instances. BFC<sup>1c</sup> solves 53 out of 115 safe instances, and 61 examples out of memory limit, with just one timeout. BFC<sup>1b</sup> solves 47 safe instances, and 9 out of time limit and 59 out of memory limit. The forward algorithm performs worst in both BFC v1.0 and BFC v2.0, where 34 instances and 35 instances are solved by BFC<sup>2f</sup> and BFC<sup>1f</sup>, respectively. There are 53 instances of timeout for both tools. The BFC<sup>2f</sup> has 28 examples out of memory, and the BFC<sup>1f</sup> runs out of memory limit for 27 examples. IIC solves 47 instances, 24 instances timeout and 44 instances out of memory. For five algorithms in MIST toolkit, EEC solves 56 instances, 13 instances timeout, and 46 out of memory. Backward algorithm performs the same as EEC, 5 more instances

TABLE 2: Petrinizer and TSE combine with the other tools. For each suite, we focus on the instance that Petrinizer or TSE are unsolved, but the combined tool is solved, then present the total number of solved instances. All data are under taking the sum of the instances that Petrinizer or TSE can solve and the disjoint instances that the other tools can solve.

	Petri (29)	Medical (12)	Bug-tr (41)	Wa-kr (46)	Soter (50)	Total (178)
Petrinizer	21	4	33	46	43	147
Petrinizer + MIST	29	12	33	46	47	167
Petrinizer + BFC	27	4	33	46	49	159
Petrinizer + IIC	29	9	33	46	47	164
Petrinizer + TTSCov	29	12	41	46	49	177
TSE	19	4	33	46	44	146
TSE + MIST	29	12	33	46	47	167
TSE + BFC	27	4	33	46	49	159
TSE + IIC	29	9	33	46	47	164
TSE + TTSCov	29	12	41	46	49	177

timeout. TSI and IC4PN solve about 30 safe instances, but CEGAR just solves 8.

For unsafe instances, TTSCov solves 56 examples out of 63, no out of memory case, but 7 timeout. Petrinizer and BFC perform well in those suite cases. Petrinizer, BFC<sup>1c</sup>, and BFC<sup>1f</sup> solve all 63 unsafe instances. TSE solves 62 instances, and just one out of memory. BFC<sup>1b</sup> has 4 timeout and 4 out of memory, respectively. BFC<sup>2b</sup> has 8 instances out of time or memory limitation, and BFC<sup>2f</sup> just has 2 instances out of memory. BFC<sup>2c</sup> performs worse, where 30 instances are solved, because of the potential bugs in concurrent mode. For some unsafe instances in the fourth suite, the forward thread has found the counterexample, but got stuck when switching the thread until timeout. What is more, there are 3 examples out of memory when running BFC v2.0 in concurrent mode. IIC and Backward performs almost the same, which solve about 50 instances, 1 out of memory and the rest are timeout. EEC, TSI, IC4PN, and CEGAR are not good on unsafe instances, especially CEGAR, which only solves 2 unsafe instances.

In brief, TTSCov performs well both for safe and unsafe instances, especially in memory usage. Petrinizer performs well in time and memory usage for all instances, but reports wrong answer for safe instances. TSE performs nearly the same as TTSCov, but incomplete the same as Petrinizer. BFC v1.0 and v2.0 are good at unsafe cases, excluding the potential bugs in BFC<sup>2c</sup>. IIC and MIST perform the same, but take more memory usage.

**5.3. Tools Combination.** Petrinizer and TSE are incomplete, but perform excellently on time and memory usage. We combine the Petrinizer and TSE with the other tools, and the total solved instances number is shown in Table 2.

Petrinizer works out 147 instances alone, but 30 instances return wrong or partly wrong result. For BFC, we compared the BFC v1.0 and BFC v2.0 in all three modes and then chose the best one. MIST stands for *Backward* algorithm, which performs the best of five algorithms in the MIST toolkit. When combining BFC with Petrinizer, 159 instances are solved. IIC and MIST solve 164 and 167, respectively, when

working together with Petrinizer. Our tool TTSCov can solve 177 instances when combined with Petrinizer.

TSE solves 146 instances in total. When combined with the other tools, the total solved instances number is the same as Petrinizer. More importantly, TTSCov can solve all collected instances when combined with Petrinizer or TSE, except one instance from the Soter suite, which all tools can not deal with.

**5.4. Memory Usage Evaluation.** To show the memory usage of TTSCov, we compare with MIST, IIC, and BFC. Figure 2 shows that TTSCov is an efficient tool in memory usage, due to the use of IC3 as the back-end engine. TTSCov solves nearly 97.3% instances within 1GB memory. About two-thirds instances can be solved within 2 GB for all tools, but TTSCov and BFC perform better than MIST and IIC in large instances. We find a bug when running BFC v2.0 for some instances from the fourth instance suite. The tool has a segmentation fault for two instances, and we have get the bug confirmation from the author. The segmentation fault instances are marked as out of memory. Petrinizer and TSE are indeed efficient for memory usage, but incomplete. Therefore, we do not compare with those. TTSCov is based on the IC3 engine, which solves the verification problem without unrolling the transition relations. This is the main reason for why TTSCov performs well in memory usage. In conclusion, TTSCov is an efficient tool in memory usage, especially for huge instances.

## 6. Conclusion and Future Works

This paper introduce an IC3-based algorithm to verify the safety properties of multithreaded programs in mobile crowdsourcing networks. The pthread-style multithreaded program is modeled as a TTS. Then the state-of-the-art SAT-based model checking algorithm is used to verify the safety properties, by computing a series of overapproximation reachability with IC3. The results show that our new approach can solve more instances compared favorably against several recently published approaches. Due to using IC3 as the back-end engine, our method is significant for its lower memory consumption. Tools combination is a good direction to

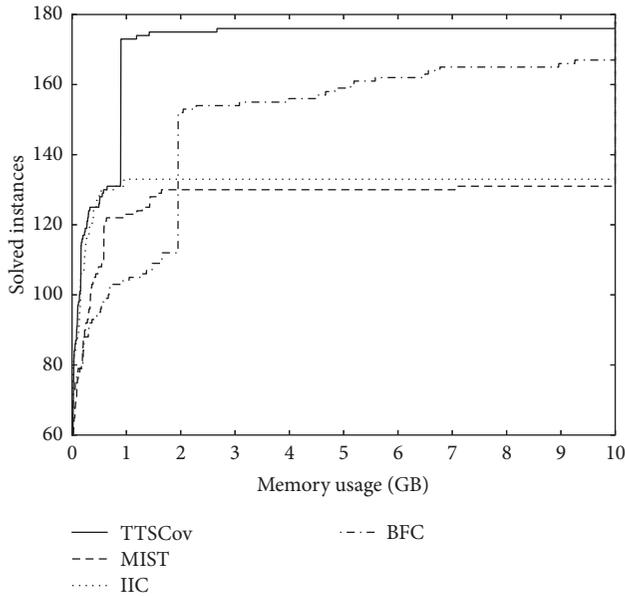


FIGURE 2: Comparison of TTSCov with MIST, IIC, and BFC for memory usage. BFC stands for BFC<sup>2b</sup>. We use two versions BFC in all three modes and find that the BFC<sup>2b</sup> performs the best in memory usage test. *Backward* algorithm is the best of five algorithms for memory usage in MIST toolkit, so MIST represented the *Backward* algorithm. For all timeout instances, we also show the memory usage in this figure.

solve the multithreaded programs for more complex mobile crowdsourcing networks. Parallel programming will be a good way to speed up the TTSCov algorithm.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

The authors would like to acknowledge that this work was supported by the National Natural Science Foundation of China (Grant no. 61133007). They also thank Carl Kwan for helpful and detailed comments and suggestions.

## References

- [1] K. Yang, K. Zhang, J. Ren, and X. Shen, "Security and privacy in mobile crowdsourcing networks: challenges and opportunities," *IEEE Communications Magazine*, vol. 53, no. 8, pp. 75–81, August 2015.
- [2] Y. Guo, X. Hu, B. Hu, J. Cheng, M. Zhou, and R. Y. Kwok, "Mobile cyber physical systems: current challenges and future networking applications," *IEEE Access*, pp. 1-1.
- [3] K. Parshotam, "Crowd computing: a literature review and definition," in *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*, pp. 121–130, the Association for Computing Machinery, East London, South Africa, October 2013.
- [4] Z. Ning, X. Hu, Z. Chen et al., "A cooperative quality-aware service access system for social internet of vehicles," *IEEE Internet of Things Journal*, pp. 1-1.
- [5] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: current state and future challenges," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 32–39, 2011.
- [6] X. Hu, T. H. S. Chu, H. C. B. Chan, and V. C. M. Leung, "Vita: a crowdsensing-oriented mobile cyber-physical system," *IEEE Transactions on Emerging Topics in Computing*, vol. 1, no. 1, pp. 148–165, 2013.
- [7] X. Hu, T. H. S. Chu, V. C. M. Leung, E. C.-H. Ngai, P. Kruchten, and H. C. B. Chan, "A survey on mobile social networks: applications, platforms, system architectures, and future research directions," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1557–1581, 2014.
- [8] X. Hu, J. Zhao, B.-C. Seet, V. C. M. Leung, T. H. S. Chu, and H. Chan, "S-afame: agent-based multilayer framework with context-aware semantic service for vehicular social networks," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 1, pp. 44–63, 2015.
- [9] P. A. Abdulla, K. Cerans, B. Jonsson, and Y.-K. Tsay, "General decidability theorems for infinite-state systems," in *Proceedings of the Eleventh Annual IEEE Symposium on Logic in Computer Science (LICS'96)*, pp. 313–321, 1996.
- [10] A. Finkel and P. Schnoebelen, "Well-structured transition systems everywhere!," *Theoretical Computer Science*, vol. 256, no. 1-2, pp. 63–92, 2001.
- [11] A. Kaiser, D. Kroening, and T. Wahl, "Efficient coverability analysis by proof minimization," in *CONCUR 2012—concurrency theory*, vol. 7454 of *Lecture Notes in Comput. Sci.*, pp. 500–515, Springer, Heidelberg, 2012.
- [12] G. Ciardo, "Petri nets with marking-dependent arc cardinality: properties and analysis," in *Application and theory of Petri nets 1994 (Zaragoza, 1994)*, vol. 815 of *Lecture Notes in Comput. Sci.*, pp. 179–198, Springer, Berlin, 1994.
- [13] C. Dufourd, A. Finkel, and P. Schnoebelen, "Reset nets between decidability and undecidability," in *International Colloquium on Automata, Languages, and Programming*, *Lecture Notes in Comput. Sci.*, pp. 103–115, Springer, Berlin, Germany, 1998.
- [14] J. Esparza and M. Nielsen, "Decidability Issues for Petri Nets," *BRICS Report Series*, vol. 1, no. 8, 1994.
- [15] S. Schmitz and P. Schnoebelen, "The power of well-structured systems," in *CONCUR 2013—concurrency theory*, vol. 8052 of *Lecture Notes in Comput. Sci.*, pp. 5–24, Springer, Heidelberg, 2013.
- [16] G. Geeraerts, J.-F. Raskin, and L. Van Begin, "Expand, enlarge, and check: new algorithms for the coverability problem of WSTS," in *FSTTCS 2004: Foundations of Software Technology and Theoretical Computer Science*, vol. 3328 of *Lecture Notes in Comput. Sci.*, pp. 287–298, Springer, Berlin, 2004.
- [17] E. A. Emerson and K. S. Namjoshi, "On model checking for non-deterministic infinite-state systems," in *Proceedings of the Thirteenth Annual IEEE Symposium on Logic in Computer Science*, pp. 70–80, IEEE, 1998.
- [18] J. Esparza, A. Finkel, and R. Mayr, "On the verification of broadcast protocols," in *Proceedings of the 14th Symposium on Logic in Computer Science*, pp. 352–359, IEEE, 1999.
- [19] P. A. Abdulla, A. Bouajjani, and B. Jonsson, "On-the-fly analysis of systems with unbounded, lossy fifo channels," in *Proceedings of the International Conference on Computer Aided Verification*, pp. 305–318, Springer, 1998.

- [20] J. Bingham and A. J. Hu, "Empirically Efficient Verification for a Class of Infinite-State Systems," in *Proceedings of the International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, vol. 3440, pp. 77–92, Springer, 2005.
- [21] J. Kloos, R. Majumdar, F. Niksic, and R. Piskac, "Incremental, inductive coverability," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 8044, pp. 158–173, 2013.
- [22] A. Kaiser, D. Kroening, and T. Wahl, "A widening approach to multithreaded program verification," *ACM Transactions on Programming Languages and Systems (TOPLAS)*, vol. 36, no. 4, 2014.
- [23] A. R. Bradley, "SAT-based model checking without unrolling," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 6538, pp. 70–87, 2011.
- [24] N. Een, A. Mishchenko, and R. Brayton, "Efficient implementation of property directed reachability," in *Proceedings of the Formal Methods in Computer-Aided Design FMCAD '11*, pp. 125–134, November 2011.
- [25] S. Graf and H. Saidi, "Construction of abstract state graphs with PVS," in *Computer Aided Verification*, vol. 1254 of *Lecture Notes in Computer Science*, pp. 72–83, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997.
- [26] M. A. Colón and T. E. Uribe, "Generating finite-state abstractions of reactive systems using decision procedures," in *Proceedings of the International Conference on Computer Aided Verification*, vol. 1998, pp. 293–304, Springer.
- [27] K. L. McMillan, *Symbolic Model Checking*, Springer US, Boston, Ma, USA, 1993, 25–60.
- [28] R. E. Bryant, "Graph-based algorithms for boolean function manipulation," *IEEE Transactions on Computers*, vol. C-35, no. 8, pp. 677–691, 1986.
- [29] J. Esparza, R. Ledesma-Garza, R. Majumdar, P. Meyer, and F. Niksic, "An SMT-based approach to coverability analysis," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 8559, pp. 603–619, 2014.
- [30] T. Murata, "Petri nets: properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.
- [31] W. Reisig, "Understanding Petri nets: Modeling techniques, analysis methods, case studies," *Understanding Petri Nets: Modeling Techniques, Analysis Methods, Case Studies*, pp. 1–230, 2013.
- [32] K. Athanasiou, P. Liu, and T. Wahl, "Unbounded-thread program verification using thread-state equations," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 9706, pp. 516–531, 2016.
- [33] G. Higman, "Ordering by Divisibility in Abstract Algebras," *Proceedings of the London Mathematical Society*, vol. 3-2, no. 1, pp. 326–336, 1952.
- [34] E. Clarke, D. Kroening, N. Sharygina, and K. Yorav, "SATABS: SAT-based predicate abstraction for ANSI-C," in *Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2005)*, vol. 3440 of *Lecture Notes in Computer Science*, pp. 570–574, Springer Verlag.
- [35] L. Zhang, W. Hu, W. Qu, Y. Guo, and S. Li, "A Formal Approach to Verify Parameterized Protocols in Mobile Cyber-Physical Systems," *Mobile Information Systems*, vol. 2017, Article ID 5731678, 2017.
- [36] E. D'Ousualdo, J. Kochems, and C.-H. L. Ong, "Automatic verification of Erlang-style concurrency," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 7935, pp. 454–476, 2013.

## Research Article

# HuAc: Human Activity Recognition Using Crowdsourced WiFi Signals and Skeleton Data

Linlin Guo , Lei Wang , Jialin Liu, Wei Zhou, and Bingxian Lu 

Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province School of Software,  
Dalian University of Technology, Dalian, China

Correspondence should be addressed to Lei Wang; lei.wang@ieee.org

Received 18 September 2017; Accepted 27 November 2017; Published 11 January 2018

Academic Editor: Kuan Zhang

Copyright © 2018 Linlin Guo 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 joint of WiFi-based and vision-based human activity recognition has attracted increasing attention in the human-computer interaction, smart home, and security monitoring fields. We propose HuAc, the combination of WiFi-based and Kinect-based activity recognition system, to sense human activity in an indoor environment with occlusion, weak light, and different perspectives. We first construct a WiFi-based activity recognition dataset named WiAR to provide a benchmark for WiFi-based activity recognition. Then, we design a mechanism of subcarrier selection according to the sensitivity of subcarriers to human activities. Moreover, we optimize the spatial relationship of adjacent skeleton joints and draw out a corresponding relationship between CSI and skeleton-based activity recognition. Finally, we explore the fusion information of CSI and crowdsourced skeleton joints to achieve the robustness of human activity recognition. We implemented HuAc using commercial WiFi devices and evaluated it in three kinds of scenarios. Our results show that HuAc achieves an average accuracy of greater than 93% using WiAR dataset.

## 1. Introduction

Human activity recognition is an important research problem in the social life, pervasive computing, and security monitoring fields [1–3]. Daily activities [4] were seen as an important means of communicating in our daily life, and we can communicate through body language like hands and head rather than speaking. Therefore, human activity recognition systems have been proposed in terms of application demand, technical support, and auxiliary devices.

Previous works related to activity recognition are roughly divided into three categories including wearable-based, vision-based, and WiFi-based. Wearable-based sensing behavior has been popular and widely used in elder healthcare, smart sensing, sports application, and tracking [1, 5, 6]. Researchers leverage the collecting information via sensors to recognize human behavior and analyze human health condition. However, it has several limitations such as *increasing the burden of users*, *the inconvenience of routine life*, and *sensors with limited power*. Vision-based activity recognition has been popular and achieves high accuracy. *The light, shadowing, privacy protection, and angle factors* increase the

difficulty of activity recognition and constrain the application fields. Microsoft released Kinect technology and Kinect can provide skeleton information using built-in sensors [7, 8]. Although Kinect-based activity recognition solves the light-environment problem and can track the skeleton joints of an activity with high accuracy, it cannot recognize the imperfect activity due to *the crowded room*, *the presence of obstacles*, and *out of the monitoring range*.

With the coverage of WiFi signals and the improvement of wireless infrastructures in public places, WiFi-based activity recognition systems [4, 9–11] leverage the change pattern of WiFi signals reflected by a human body to recognize the activity. WiFi-based activity recognition systems [12–14] not only ease the burden of wearable-based users, but also can sense the presence of obstacles in comparison with Kinect-based works. For example, WiVi [14] can sense the user's behavior through the wall, and RF-Capture [11] tracks the 3D positions of a human body when the person is occluded completely and captures the human figure without wearable devices.

We are interested in BodyScan system [15], and it is estimated on the idea of the combination of the wearable

sensors and WiFi signals. Moreover, it overcomes key limitations of existing wearable devices by providing a contactless and privacy-preserving approach to capture a rich variety of human activities. Based on this work, we explore the combination of CSI and skeleton data to sense human behavior. According to the works mentioned above, we explore three issues of activity recognition in this paper. First, we construct a WiFi-based activity recognition dataset named WiAR to provide a benchmark for previous works. Second, we design the mechanism of subcarrier selection to improve the robustness of activity recognition in the WiAR dataset. Third, we combine WiFi signals with crowdsourced skeleton data to improve the accuracy and robustness of activity recognition breaking the limitations of Kinect technology. The contributions of our work are summarized as follows:

- (i) We propose a HuAc system to recognize human activity and also construct a WiFi-based activity recognition dataset named WiAR as a benchmark to evaluate the performance of existing activity recognition systems. We use the kNN, Random Forest, and Decision Tree algorithms to verify the effectiveness of the WiAR dataset.
- (ii) We detect the start and end of the activity using the moving variance of CSI. Moreover, we leverage *K*-means algorithm to cluster effective subcarriers according to subcarrier's sensitivity and improve the robustness of activity recognition.
- (iii) We develop a selection method of skeleton joints based on KARD's work named SSJ, and it considers the spatial relationship and the angle of adjacent joints as auxiliary information of human activity recognition to improve the accuracy of tracking.
- (iv) We implement the fusion framework of CSI and skeleton data to sense the activity and solve the limitations of CSI-based and skeleton-based activity recognition, respectively. Experimental results show that HuAc achieves the accuracy of greater than 93%.

The rest of this paper is organized as follows. We introduce the related work in Section 2. Section 3 introduces preliminaries of WiFi-based activity recognition, and we describe the overview of HuAc in Section 4. Section 5 describes Kinect module, and WiFi module is shown in Section 6. Section 7 describes the process of human activity recognition. Section 8 evaluates the performance of HuAc system, and we give a case study about a motion-sensing game using WiFi signals in Section 9. Section 10 lists several discussions, and we give the conclusion of this paper in Section 11.

## 2. Related Work

In this section, related works on human activity recognition can be divided into two categories: Kinect-based, WiFi-based.

*2.1. Kinect-Based Activity Recognition.* Vision-based activity recognition has been proposed and developed in the computer vision field. With the release of Kinect, researchers

explore the human activity recognition using depth information and skeleton joints data provided by Kinect [7, 8, 16]. Biswas and Basu [8] leverage the histogram of depth information to recognize eight gestures. Moreover, the differences between continuous frames can obtain the motion profile to describe various gestures. Other works [7, 16] leverage depth information in combination with color image to improve the accuracy of gestures recognition. The limitations of Kinect-based activity recognition contain the restriction of sensing field, skeleton joints overlapping, and position-dependence factors. HuAc system explores the spatial relationship of skeleton joints to describe the trajectory of an activity and combines with CSI to improve the robustness of human activity recognition in a dynamic environment.

*2.2. WiFi-Based Activity Recognition.* Early works [17–19] explore the attenuation characteristics of WiFi signals to locate the position of someone and count the number of people in the indoor environment. Researchers study the signal pattern reflected by a human body to sense human behavior [11, 20–22]. These works describe human behavior recognition using coarse-grained RSSI information. For example, WiGest [18] studies the relationship between RSSI fluctuation and gestures to control media player actions without training. Therefore, we explore the relationship between RSSI fluctuation and human movement to detect the presence of an activity.

With the requirement of the practical application and the limitations of RSSI, an increasing number of researchers begin to explore fine-grained channel state information (CSI) to sense human behavior. Compared with RSSI, CSI can capture the tiny behavior [2, 9, 23–28] in terms of location, speed, and direction. WiFall system [2] detects a fall behavior by learning the specific CSI pattern. E-eyes [9] recognizes walking activity and in-place activity by adopting moving variance of CSI and fingerprint technique. Walking activity causes significant pattern changes of the CSI amplitude over time, since it involves significant body movements and location changes. In-place activity (watching TV) only involves relative smaller body movements and will not cause significant amplitude changes with repetitive patterns. The relationship between an activity and the place where an activity occurs motivates the novel idea on human activity recognition. CARM [10] shows the correlation between CSI value and human activity by constructing CSI-speed and CSI-activity model. WiDance [28] explores the Doppler shifts reflected by human behavior to predict the motion direction for the Exergames. We design the combination system of Kinect-based and WiFi-based methods to recognize an activity in different environments such as gaming system, supermarket, and elder health applications.

## 3. Preliminaries

*3.1. RSSI and CSI.* Received Signal Strength Indicator (RSSI) [29] in the level of packet represents signal-to-interference-plus-noise ratio (SINR) over the channel bandwidth as follows:

$$\text{RSSI} = 10 \lg (\|V\|^2), \quad (1)$$

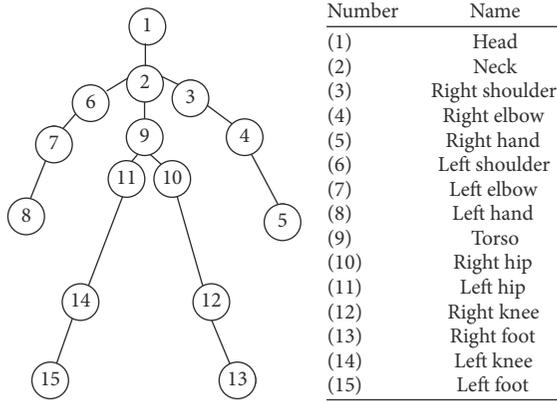


FIGURE 1: Skeleton joints.

where  $V$  is signal voltage. RSSI is the received signal strength in decibels (dB) and mapped into the distance according to Log-distance path loss model to roughly locate users or devices.

Channel State Information (CSI) depicts multipath propagation at the granularity of OFDM subcarrier in the frequency domain. It contains amplitude and phase measurements as follows:

$$h = |h| e^{j \sin \theta}, \quad (2)$$

where  $|h|$  and  $\theta$  are the amplitude and phase, respectively. The variable  $h$  shows CSI value of each subcarrier. We study the characteristics of each subcarrier to sense activity in the following work.

**3.2. Kinect Technology.** Kinect (RGB-D camera) refers to the advanced RGB/depth sensing, hardware, and the software-based technology that interprets the GRB/depth information. The hardware contains a normal RGB camera, a depth sensor (infrared projector and infrared camera), and a four-microphone array, which is able to provide depth signals, RGB images, and audio signals simultaneously. Kinect-based activity recognition algorithm frequently fails due to occlusions, overlapping joints (limbs close to the body), or clutter (other objects in the scene) [7]. A skeleton reported by Kinect contains 15 joints in Figure 1. We explore the corresponding relationship between skeleton joints and CSI to analyze the characteristics of an activity. Moreover, we explore the fusion information to improve the accuracy of human activity recognition. The details of Kinect-based activity recognition are listed in Section 5.

**3.3. WiAR: Constructing WiFi-Based Activity Dataset.** At present, there is no WiFi-based public activity dataset as well as vision-based public activity dataset. Due to the sensitivity of WiFi signals, it is hard for peer researchers to reproduce and evaluate previous works. Therefore, we construct the WiAR dataset which collects WiFi signals reflected by sixteen activities in three indoor environments such as empty room, meeting room, and office listed in Table 1. Each activity is performed 50 times by 10 volunteers which consist of five

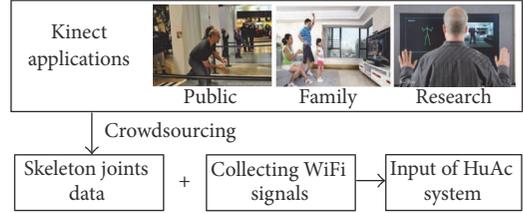


FIGURE 2: The framework of crowdsourced dataset.

females and five males, and the height of human body ranges from 150 cm to 185 cm.

The environmental complexity according to the room layout divides into three levels including empty environment, normal environment, and complex environment. First, empty environment describes no people and furniture around it. We obtain the high-quality WiFi signals from the empty room due to less noise and treat it as a baseline of WiAR dataset. Then, the normal environment contains furniture and working people. Compared with the empty environment, the multipath effect reflected by the furniture enriches collecting WiFi signals. Finally, a complex environment with furniture and moving people increases the difficulty of human activity recognition. The performance of WiAR dataset is given in Section 8.

**3.4. Crowdsourced WiFi Signals and Skeleton Joints.** Crowdsourced-based applications [30–37] have been increasingly developed by collecting data and reducing the cost in the Internet field. For the macrolevel network, the work [30] proposed a crowdsensing-oriented mobile cyber-physical system to provide the practical usage of the vita. For the microlevel wireless network, related works [38–41] leverage crowdsensing WiFi signals to detect the user’s location.

In our work, we attempt to collect WiFi signals and crowdsourced skeleton joints to reduce the training burden for collecting activity dataset. We obtain the activity label by leveraging the help from Kinect’s user. The framework of crowdsourced WiFi signals and skeleton joints are shown in Figure 2.

## 4. Overview of HuAc

**4.1. Observations.** The following observations come from the combination of our results and previous works [20, 42–44].

*The Impact of Indoor Environment on WiFi Signals Has a Difference with Time.* RSSI and CSI keep stability in the static indoor environment, and RSSI fluctuation ranges from 0 dB to 5 dB (empty environment: 0–3 dB; home environment: 0–7 dB; office: 0–5 dB; dynamic environment: 5–10 dB). Although RSSI sharply changes with environmental change, it cannot describe the fine-grained change of indoor environment due to the multipath effect. However, CSI is able to sense the change of fine-grained environment and detects what happened in an indoor environment. Specifically, RSSI only can find the environmental change and cannot sense how

TABLE I: WiFi-based activity recognition dataset (WiAR).

Granularity	Activities	Environments	Devices
Activity	Forward kick, side kick, bend, walk, phone, sit down, squat, drink water	Empty room, meeting room, office	Router, laptop with 5300 card
Gestures	Horizontal arm wave, two-hand wave, high throw, toss paper, draw tick, draw x, hand clap, high arm wave	Empty room, meeting room, office	Router, laptop with 5300 card

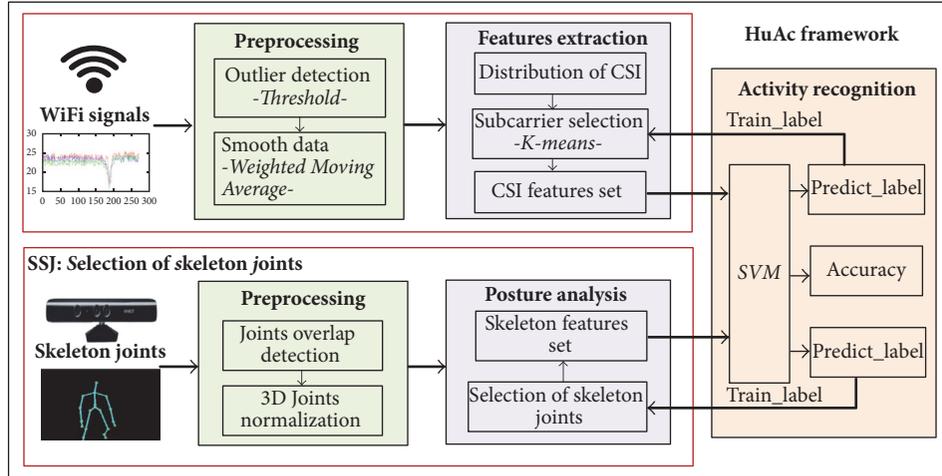


FIGURE 3: The framework of HuAc system.

the environment changes. CSI can find what causes environmental change and also can recognize how the environment changes such as tracking, sensing environment, and activity recognition.

*It Is Hard to Distinguish Similar Activities.* Existing works [2, 15, 45] explore the similar activity recognition. For example, WiFall [2] extracts seven features to describe fall behavior because similar activity causes the similar patterns of CSI, and it is difficult to distinguish them only using anomaly detection. The following RT-Fall system adopts the CSI phase difference to segment fall and fall-like activities because the phase difference of CSI is a more sensitive signature than CSI amplitude for activity recognition. The phase of CSI depends on the variation of LOS (Line-of-Sight) length. Therefore, the breakthrough point of the similar activity recognition rests on the physical difference between similar activities.

*The Same Activity Operated by Different People Has Various Signal Patterns.* According to our observations, the amplitude of CSI reflected by the same activity changes continuously in the different time and environments. Therefore, we cannot recognize activity with high accuracy according to the amplitude of CSI. The changing pattern of signals reflected by an activity can describe the characteristic of activity as verified by Smokey [25]. Therefore, we explore the changing pattern of signals to recognize an activity.

*The Impact of Activity with Different Directions on Activity Recognition.* In order to explore the impact of direction on activity recognition, we design a simple and clear experiment

on the playground because the playground does not have rich multipath effect and other wireless devices. We explore the impact of four directions including east, west, north, and south on the change pattern of signals, and the difference between face and back to the AP is biggest. Moreover, CSI data we collect in the playground contains less noise than that in an indoor environment.

*4.2. Framework of HuAc.* The HuAc framework consists of the Kinect-based module and WiFi-based module in Figure 3. We describe details of each module, respectively.

Kinect module consists of the preprocessing and posture analysis. We detect the overlap of skeleton joints using the statistical method and complete the normalization of skeleton joints. In order to obtain effective features of skeleton joints, we analyze postures of an activity according to the sequence of skeleton joints. Moreover, we design a selection method of skeleton joints named SSJ according to the result of posture analysis. Finally, we extract features of skeleton joints according to effective skeleton joints and also consider the spatial relationship of adjacent joints as auxiliary information to sense human activity.

WiFi module consists of the preprocessing and features extraction. In the preprocessing stage, we detect and remove the outlier data of an activity sequence according to the variance of RSSI reflected by an activity. After removing outlier data, we leverage the weighted moving average to smooth the activity data. For features extraction, we first analyze the amplitude distribution of CSI reflected by an activity to evaluate the sensitivity of the subcarrier on an

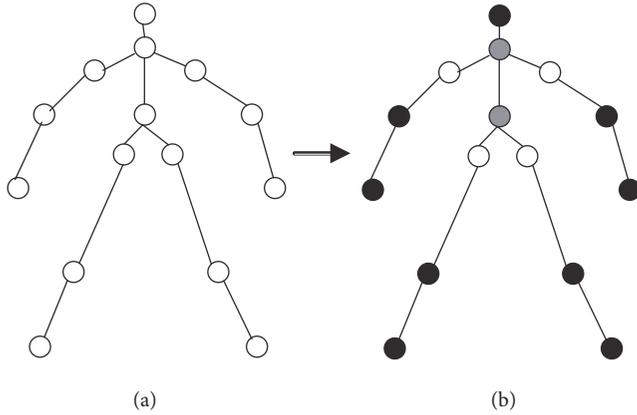


FIGURE 4: Skeleton structure [7]. (a) A skeleton structure contains 15 skeleton joints. (b) The white circle represents skeleton joints without direction such as shoulder and hip. The gray circle represents the neck and the torso which has a weak effect on the upper-body activity and the lower-body activity except the squat. The black circle represents normal skeleton joints.

activity. Then, we use  $K$ -means algorithm to cluster effective subcarriers. Finally, we extract important features from effective subcarriers to improve the stability of human activity recognition.

We use the combination information of CSI features set and the skeleton features set as an input of SVM to recognize human activity. Compared with the result of *predict\_label*, we give a feedback to the previous process of HuAc framework by using a *train\_label*, respectively.

## 5. Kinect Module

We mainly describe the details of Kinect module on the human activity recognition. Kinect module contains the preprocessing and posture analysis.

**5.1. Preprocessing.** The collected skeleton data contain empty values due to the overlap of skeleton joints or the occlusion in the motion-sensing game. Therefore, we need to detect the overlapping joints and replace the invalid values by recovering the true value of the overlapping joints. We leverage the relationship between the coordinates of adjacent joints to detect the overlapping joints. Certainly, we discard the sample of an activity when the percent of invalid joints exceeds the threshold.

After recovering the invalid data, we normalize the coordinates of skeleton joints due to the differences of people's height and the distance between the user and the sensor. The work [7] extracts 11 joints (except right shoulder, left shoulder, right hip, and left hip) from 15 joints in Figure 4, and we explore 30 subcarriers with the similar pattern reflected by a human body. Therefore, we select 15 joints to match the 15 subcarriers. Let  $J_i$  be one of the 15 joints detected by the Kinect, and the coordinates vector  $f$  is given by

$$f = \{j_1, j_2, \dots, j_i, \dots, j_{14}, j_{15}\}, \quad (3)$$

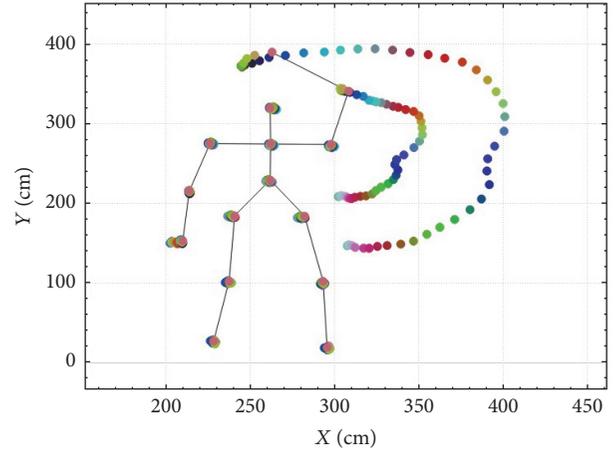


FIGURE 5: High arm wave tracking using skeleton data. The activity has two active joints (right hand, right elbow), and the direction changes with every clockwise movement. However, adjacent joints have the slight change in a certain range.

where  $j_i$  is the vector containing the 3D normalized coordinates of the  $i$ th joint  $J_i$  detected by Kinect. Thus,

$$j_i = \frac{J_i}{s} + T_i, \quad 1 \leq i \leq 15, \quad (4)$$

where  $s$  is the scale factor which normalizes the skeleton according to the distance  $h$ , between the neck and the torso joints of a reference skeleton, and

$$s = \frac{\|J_9 - J_2\|}{h}. \quad (5)$$

The translation matrix,  $T$ , needs to set the origin of the coordinate system to the torso. After preprocessing phase, we obtain high-quality skeleton data.

**5.2. Postures Analysis.** An activity consists of subactivity sequence over time. According to the skeleton structure, a human body is divided into two parts including upper body and lower body. Upper body contains five joints (right elbow, left elbow, right hand, and head) and two baseline joints (neck, torso) as in Figure 4. Lower body contains four joints (right foot, left foot, right knee, and left knee). We reproduce the tracking of skeleton joints using QT tool and plot the trajectory chart of each activity. We observe that the adjacent joints keep the similar track in Figure 5, and some joints have slight movement influenced by human activity. For example, when the right elbow and right hand move in the clockwise direction to complete the horizontal arm wave, we observe that right hip and left hip have slight movement.

According to the change of joints sequence, we can segment an activity into several subactivities in terms of direction and pause factor. Horizontal arm wave behavior consists of four postures (subactivities) as in Figure 6. Each subactivity roughly contains 14 frames and  $F_i$  represents the  $i$ th frame (packet) of the activity reported by Kinect. We can evaluate the rough activity according to the sequence of

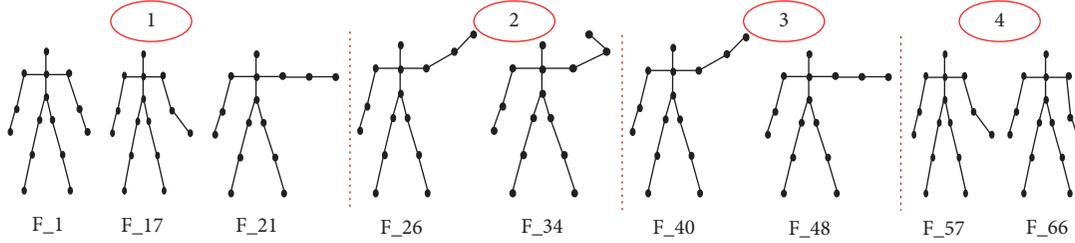


FIGURE 6: Postures of horizontal arm wave.

subactivity. Except for related joints of each subactivity, torso and hip joints have a weak swing. We neglect the impact of weak swing on the activity recognition. We pay more attention to the selection of skeleton joints in the following section.

**5.3. SSJ: Selecting Skeleton Joints.** We design a selection method of skeleton joints named SSJ to describe a fine-grained subactivity. After postures analysis, we know the relationship between a subactivity and key skeleton joints. We expand the coordinated system of human skeleton to miniature coordinated system of subactivity skeleton by the above-mentioned relationship. The miniature coordinated system needs to determine a fixed skeleton joint and different subactivities have different fixed skeleton joints. For example, we observe that shoulder joint is a fixed joint from the process of high arm wave behavior. Therefore, we determine the starting point coordinate of the miniature coordinated system corresponding to the subactivity.

## 6. WiFi Module

We introduce the design details of WiFi module on the human activity recognition. WiFi module consists of the preprocessing and features extraction.

**6.1. Preprocessing.** The collected data with noises increases the difficulty of activity recognition due to the tiny differences between noises and WiFi signals reflected by a fine-grained activity. Outlier data also weaken the quality of collecting data. Therefore, we detect outlier using the variance-based method and remove high-frequency signals using the low-pass filter. Moreover, we reduce the sawtooth wave of the filtered signal by using the weighted moving average.

**6.1.1. Outlier Detection and Removing High Frequency.** Outlier has an important impact on the quality of collecting data because outlier increases or decreases the fluctuation strength of WiFi signals. We analyze the RSSI distribution of an activity to evaluate the possible experience-threshold. Then, we combine the variance of RSSI and the experience-threshold to detect outlier. After removing outlier data, the activity corresponds to the low-frequency change of CSI according to the waveform of CSI reflected by an activity. Therefore, we adopt the low-pass filter to remove the high-frequency data in Figure 7.

**6.1.2. Weighted Moving Average.** For filtered signal, signal data still contain sawtooth wave. Because CSI is sensitive to indoor layout or human movement, and the received CSI fluctuation caused by the environment is hard to distinguish from the fluctuation caused by a fine-grained activity. Therefore, we smooth the CSI data using the weighted moving average as proposed in WiFall [2]. We randomly select 15 subcarriers from 30 subcarriers which correspond to 15 skeleton joints of Kinect technology. Each CSI stream contains 15 subcarriers as  $\{CSI_1, CSI_2, \dots, CSI_{15}\}$ .  $CSI_{t,1}$  is the first subcarrier of CSI at time  $t$ .  $\{CSI_{1,1}, \dots, CSI_{t,1}\}$  indicates the CSI sequence of first subcarrier in the time period  $t$ . The latest CSI has weight  $m$ , the second latest  $m - 1$ , and so on. The expression of CSI series is shown as follows:

$$CSI_{t,1} = \frac{1}{m + (m - 1) + \dots + 1} \times (m \times CSI_{t,1} + (m - 1) \times CSI_{t-1,1} + \dots + 1 \times CSI_{t-m,1}), \quad (6)$$

where  $CSI_{t,1}$  is the averaged new CSI. The value of  $m$  decides in what degree the current value is related to historical records. In our study, we select  $m$  according to the experience and trial method. We first set  $m$  as 5 which means the length of 5 packets. A weighted moving average algorithm and median filter have the similar effect on the original signals recorded by the receiver in Figure 7. They can remove the galling of signals and alleviate the sharp change of signals. With the  $m$  increasing, the weighted moving average algorithm becomes more smooth than the low-pass filter and the median filter. Finally, we set  $m$  to 10 because each activity produces a sharp change in 10 packet periods.

**6.2. Feature Extraction.** Plenty of related works summarize the importance of features extraction for human activity recognition in a dynamic indoor environment. We segment activity after smoothing CSI and extract features of each activity according to activity characteristics. Kinect-based features extraction quotes the work [3].

**6.2.1. Activity Segmentation.** Activity segmentation mainly detects the start and end of an activity and removes the nonactivity packets from a sample which corresponds to the whole activity. We propose two methods to detect the start and end of an activity and improve the robustness of segmentation algorithm. First, we remove the first second and the last-second data sequence of an activity to reduce

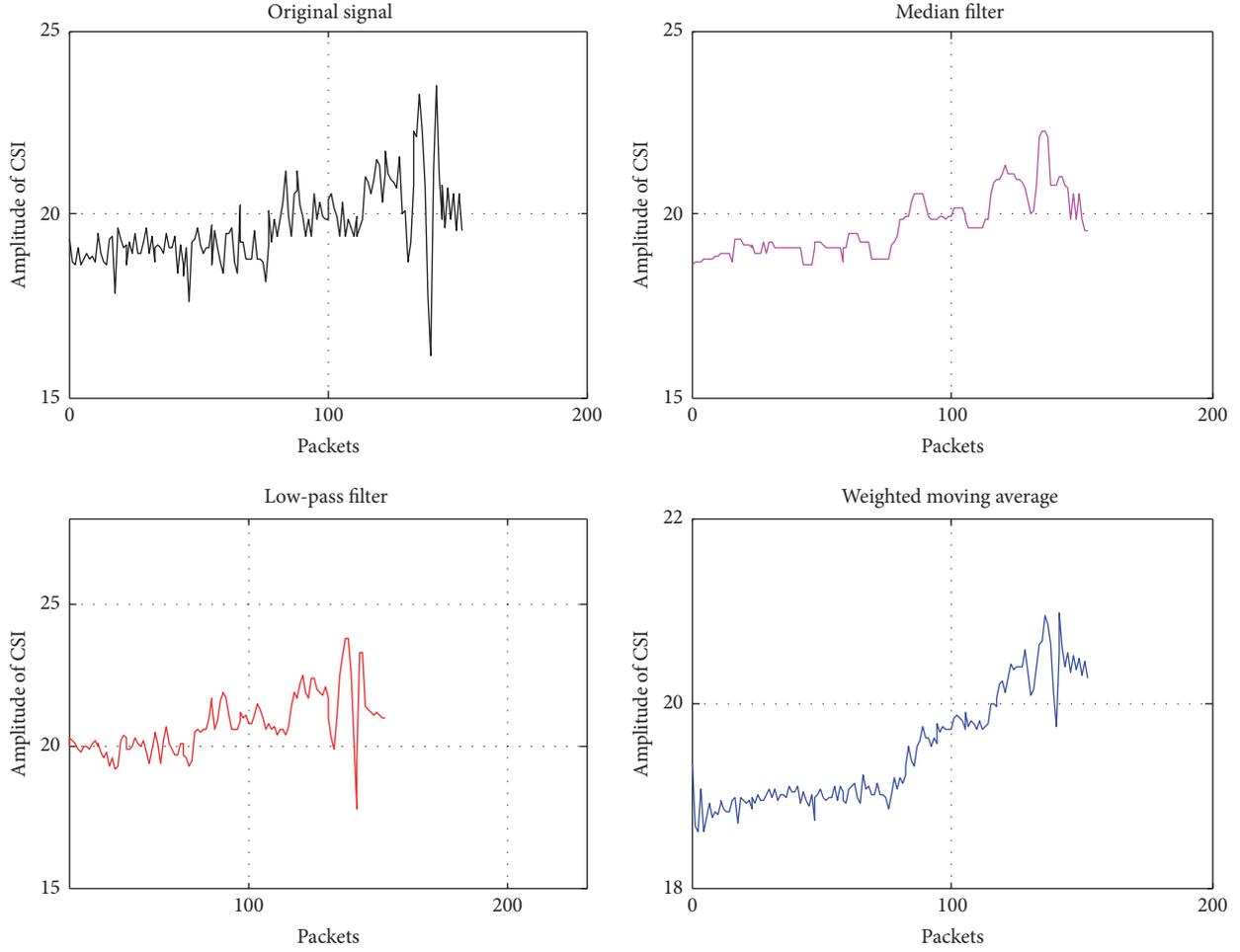


FIGURE 7: Methods of signal filtering.

the error of true activity sequence in our experimental environment. But this method is invalid in the practical environment due to the unknown time which each activity starts. Therefore, we leverage moving variance of CSI to detect the start and end of each activity. Moving variance of CSI describes the difference of the local packets reflected by the activity. Packet sequences on the corresponding activity are defined as  $X = \{x_1, x_2, \dots, x_n\}$ .  $X$  represents data sequence (a sample) of an activity, and  $x_i$  represents the  $i$ th packet in the data sequence. We often use the standard deviation instead of the variance of CSI as follows:

$$\sigma_i = \sqrt{\frac{\sum_1^m (x_{i+j-1} - \bar{x})^2}{m}}, \quad (i = 1, 2, \dots, n - m), \quad (7)$$

where  $m$  represents step-size and  $\bar{x}$  is the mean value of samples.

We construct a window per 10 packets from the packet sequence of each sample and compute the variance of the window. Then, we construct the moving variance histogram and compare with other strength windows. Finally, we can detect the sharp points of each activity and roughly recognize

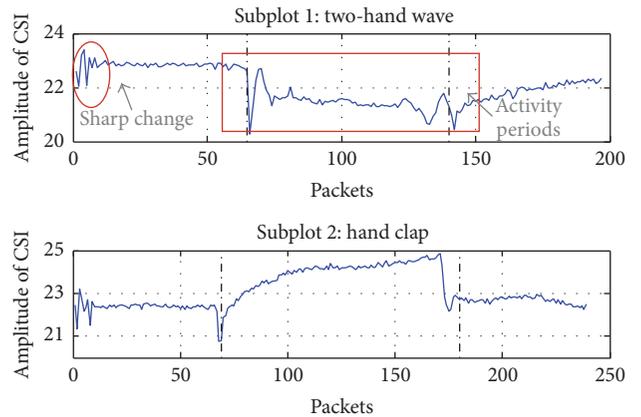


FIGURE 8: Segmentation point of similar activity.

the start and end of each activity from the data sequence. The start and end of the activity period are shown in Figure 8. The red circle describes a sharp change of CSI at the start point of collecting data, but it is not the true start of an activity. The red

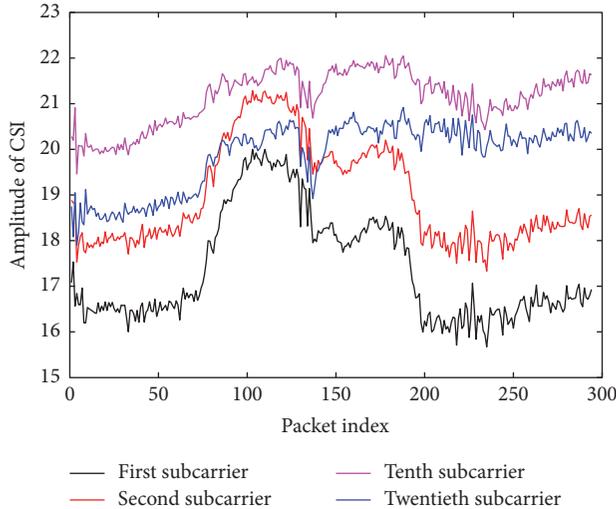


FIGURE 9: The fluctuation of different subcarriers reflected by the horizontal arm wave behavior.

rectangle represents the duration of activity. Moreover, the black dotted line roughly represents the true start and end of the activity. According to our experimental results, detecting the start and end of the activity still causes a small error due to the sensitivity of signals.

**6.2.2. Subcarrier Selection and Feature Detection.** According to our observation, subcarriers have the similar tendency for the same activity in Figure 9, but they have different sensitivity. Therefore, we select the obvious subcarriers reflected by an activity using  $K$ -means to achieve the robustness of human activity recognition. Thirty subcarriers are divided into 3 clusters using  $K$ -means algorithm in Figure 10. According to the output of  $K$ -means algorithm on subcarriers, CSI features we extract include variance, the envelope of CSI, signal entropy, the velocity of signal change, median absolute deviation, the period of motion, and normalized standard deviation. Finally, we construct the features set of CSI.

## 7. HuAc: Activity Recognition

We explore the relationship between CSI-based and skeleton-based methods on human activity recognition in Figure 11. The CSI-based method leverages the signal pattern to recognize an activity. The skeleton-based method uses the coordinate change of skeleton joints to recognize the same activity. From the opinion of experiment results, an activity with back to the AP has more complex CSI pattern and has the smaller amplitude than that with face to AP.

We mainly introduce several classification algorithms used by the human activity recognition field including kNN, Random Forest, Decision Tree, and SVM. In the following sections, we verify that the performance of SVM outperforms others. We select SVM classification algorithm to recognize sixteen activities in the WiAR dataset. CSI features set and skeleton features set as the inputs of SVM train the optimal model to achieve the stable accuracy of activity recognition. The outputs of SVM contain the

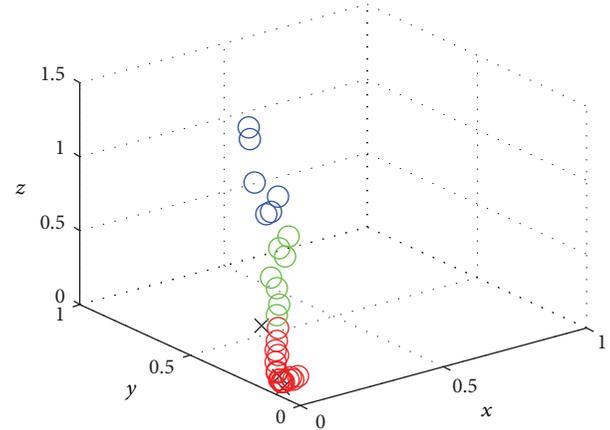


FIGURE 10: Clustering subcarriers.

*accuracy*, *predict\_label*, and *prob\_estimates*. We evaluate the performance of classification algorithm according to the accuracy and achieve the accuracy of activity recognition using the *predict\_label*. According to the match level between *train\_label* and *predict\_label*, we obtain the false positive rate and the false negative rate. We analyze the result and give a feedback on the previous step. According to the feedback, we pay more attention to the activity with low accuracy.

## 8. Implementation and Evaluation

### 8.1. Implementation

**8.1.1. Experimental Setup.** We use a commercial TP-Link wireless router as the transmitter operating in IEEE 802.11n AP mode at 2.4 GHz. A Thinkpad 400 laptop running Ubuntu 10.04 is used as a receiver, which is equipped with off-the-shelf Intel 5300 card and a modified firmware. During the process of receiving WiFi signals, the receiver pings 30 pkts/s from the router and records the RSSI and CSI from each packet. Three experimental environments including empty room, meeting room, and office are shown in Figure 12.

**8.1.2. Experimental Data.** We deal with data from three cases: For *WiFi-based activity data*, we collect activity data in different indoor environment. For *skeleton data*, we directly leverage the KARD dataset [3] to get the skeleton data. For *environmental data*, we mainly collect data from the empty room, meeting room, and office with the human. Our goal is to explore the impact of the environmental factor on the WiFi signals and analyze the differences between an activity and environmental change on WiFi signals according to the above-mentioned three kinds of data.

We collect WiFi signals to construct a new dataset named WiAR which contains 16 activities with 50 times performed by ten volunteers. The details of WiAR have been introduced in Section 3. The KARD contains RGB video (.avi), depth video (.avi), and 15 skeleton points (.txt). Each volunteer performs 18 activities 3 times each with ages ranging from 20–30 years and height from 150–180 cm. In this paper, we only select 16 activities as target activity listed in Table 1.

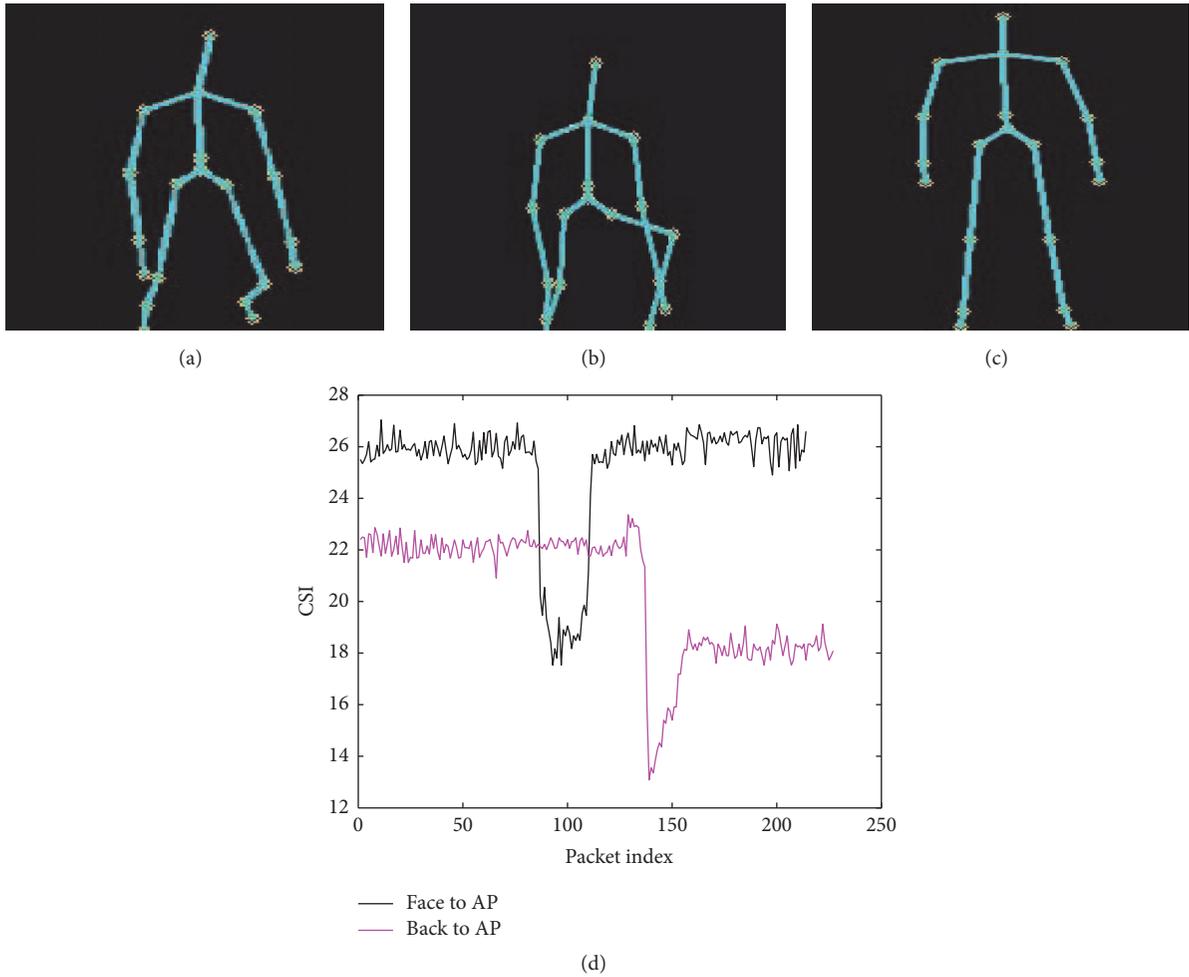


FIGURE 11: Skeleton joints sequence and CSI change of squat behavior. (a)–(c) represent the skeleton sequence of squat behavior. (d) is the CSI change reflected by squat behavior in terms of face to AP and back to AP.

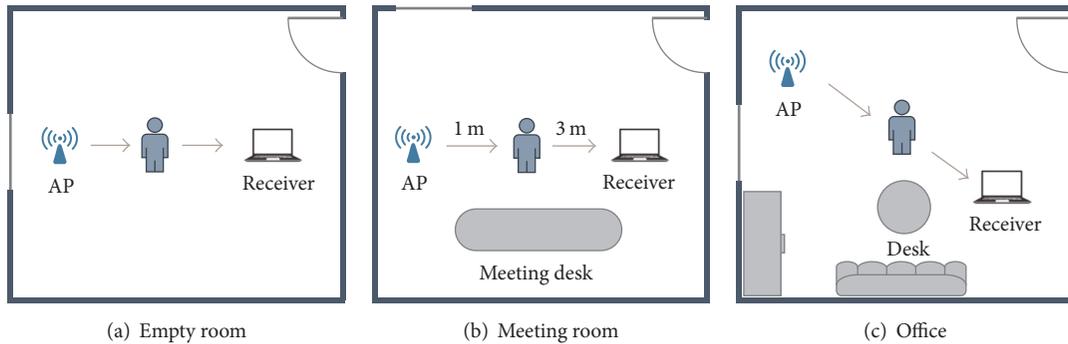


FIGURE 12: Experimental scenarios.

We design three experimental schemes to analyze the accuracy of activity recognition. First, we collect RSSI and CSI to recognize an activity as the reference point. Second, we leverage the skeleton data of KARD to recognize an activity by using our method and previous method [3] in the similar indoor environment. Third, we propose a fusion scheme

which CSI combines with skeleton data to recognize an activity. Moreover, we design another experimental scheme in which volunteer performs an activity with repeating 10 times. The goal of the experimental scheme is to investigate the periodic regularity of CSI change influenced by the same activity.

TABLE 2: Performance comparison by four classification algorithms.

Method	10 subcarriers			30 subcarriers		
	A	B	C	A	B	C
kNN	0.875	0.916	0.947	0.916	0.895	0.947
Random Forest	0.885	0.906	0.958	0.906	0.895	0.948
Decision Tree	0.8542	0.822	0.916	0.865	0.834	0.917
SVM	<b>0.9625</b>	<b>0.9688</b>	<b>0.975</b>	<b>0.94375</b>	<b>0.90625</b>	<b>0.9375</b>

8.2. *Evaluation of WiAR Dataset.* We analyze activity data of all volunteers to evaluate the performance of WiAR dataset using kNN with voting, Random Forest, and Decision Tree algorithms.

We study the impact of subcarriers and antennae on the performance of activity recognition by using four classification algorithms shown in Table 2. It shows that the accuracy using SVM outperforms other classification algorithms and 10 subcarriers obtained by subcarrier selection mechanism increase 4.26% when compared with activity recognition using 30 subcarriers. Three antennae such as A, B, and C increase the diversity of CSI data and keep more than 80% of activity recognition accuracy. The four algorithms verify the effectiveness of WiAR dataset.

### 8.3. Evaluation of Activity Recognition

8.3.1. *Performance of Activity Recognition Using RSSI.* The section evaluates the performance of RSSI on the human activity recognition. The difficulty we encounter in the process of activity recognition using RSSI is how to deal with the multipath effect caused by indoor environment and reflection effect caused by human behavior. We select an indoor environment as a reference environment which keeps static and only contains a volunteer and an operator. We leverage RSSI variance as an input of SVM to obtain the 89% of average recognition accuracy in the static environment. When other people move and are close to the control area of WiFi signals, the accuracy of activity recognition decreases to 77% with the high stability. Several activities face the low accuracy such as two-hand wave, forward kick, side kick, and high throw. The average false positive rate is 8.9% and increases to 15.3% in a dynamic environment. Therefore, human activity recognition using RSSI needs the help of CSI-based method to improve the accuracy and the robustness of human activity recognition.

8.3.2. *Performance of Activity Recognition Using CSI.* This section elaborates the impact of interference factors on human activity recognition using CSI in the following four aspects: human diversity, similar activities, different indoor environments, and the size of a training set. Moreover, we keep the fixed position of volunteers and the distance between receiver device and transmitter device in the whole experiment.

*The Impact of Human Diversity on the Accuracy.* Human diversity not only increases the diversity information of CSI but also raises the difficulty of activity recognition because

different people have different motion styles such as speed, height, and strength. We achieve 93.42% of average recognition accuracy for all volunteers in Figure 13(a). We select two volunteers including volunteer A and volunteer B to verify the impact of human diversity on the accuracy. Volunteer A which often regularly exercises obtains 97.1% of average recognition accuracy. Volunteer B which rarely exercises in the routine lives achieves 92.3% of average recognition accuracy. Therefore, the exercise experience increases the differences between activities due to standard activity and improves the recognition accuracy.

*The Impact of Similar Activity on the Accuracy.* We explore two group similar activities including high arm wave, horizontal arm wave, high throw, and toss paper in Figure 13(b). The first group activity achieves 92.5% of average recognition accuracy and 94.6% for the second group. The false positive for similar activity is higher than independent activity. For example, forward kick and side kick also belong to the similar activity, and the difference between them is the moving direction. In order to obtain the better accuracy, we will consider the impact of moving direction on the signal change in the future work.

*The Impact of Indoor Environment on the Accuracy.* As shown in Figure 12, there are three experimental environments including empty room, meeting room, and office in terms of the complexity. The accuracy about three environments is shown in Figure 13(c). The accuracy of the meeting room with 94.7% outperforms the other two environments, and then accuracy was 93% for empty room and 87% for office due to multipath effect. The meeting room generates 2.6% of average error, and 9.8% of average error in the office due to paths excessively reflected by the body. We will deeply explore the multipath effect using the amplitude and phase of CSI in the future work.

*The Impact of Training Size on the Accuracy.* We design three proof schemes to analyze the accuracy of human activity recognition by using different training sizes in Figure 13(d). We first introduce three activity sets and three training sets. Activity set 1 consists of horizontal arm wave, high arm wave, high throw, and toss paper. Activity set 2 contains two-hand wave and handclap activity. Activity set 3 consists of phone, draw tick, draw x, and drink water. Moreover, these activity sets come from the same people. With the training size increasing, the accuracy of activity recognition is improved by about 10% for the activity set 1. Activity set 1 has a low accuracy because activity set 1 contains more

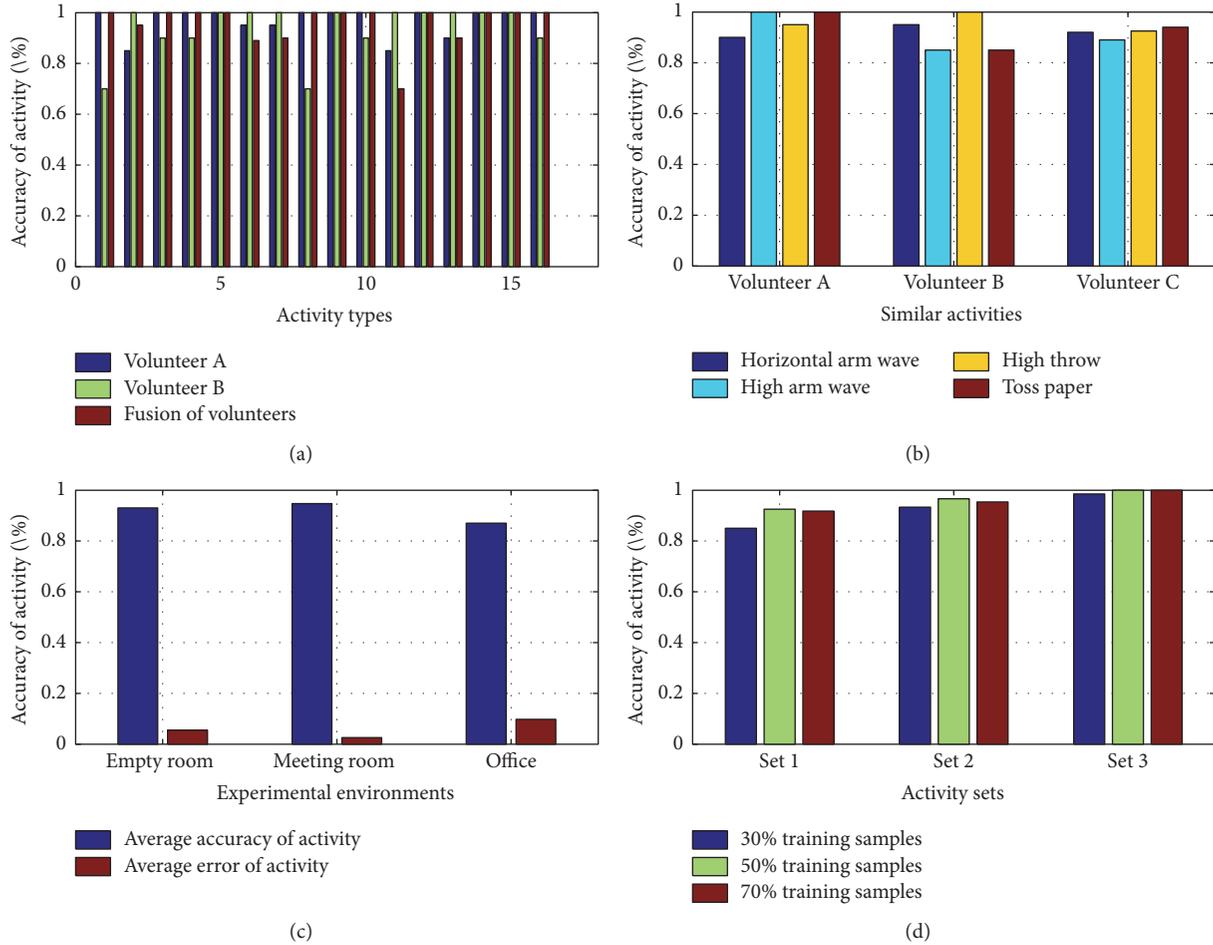


FIGURE 13: Performance analysis of activities using CSI. (a) Sixteen activities include horizontal arm wave, high arm wave, two-hand wave, high throw, draw x, draw tick, toss paper, forward kick, side kick, bend, handclap, walk, phone, drink water, sit down, and squat. (b) Four activities contain horizontal arm wave, high arm wave, high throw, and toss paper. (c) The impact of experimental environments on accuracy. (d) The impact of training samples on accuracy of three activity sets.

similar activities. Although activity set 3 also contains similar activities, the accuracy is better than activity set 1 due to the strength of activity.

**8.3.3. Performance between Kinect-Based and WiFi-Based Activity Recognition.** It is hard for the waveform of RSSI with noise to keep the stability when controlling area changes during collecting data. Therefore, we use waveform shape of RSSI to recognize an activity that is not a better choice for the current level of technology. Waveform pattern of CSI can describe an activity with credibility and fine-grained way. The mapping relationship between CSI-based and Kinect-based activity recognition for various activities is represented by using several parameters shown in Table 3. The environmental factor is evaluated by using the number of multipaths and the complexity of the indoor environment. In order to extend the application field of activity sensing, we construct the mapping relationship between CSI-based and Kinect-based activity recognition. The mapping relationship can avoid information loss. For example, once one of the two

datasets is lost, activity recognition system still works by using another dataset information.

We evaluate the performance of human activity recognition from KARD dataset [3]. The highest recognition rate is 100% (side kick, handclap), while the worst is 80% (high throw). We propose a selection method of skeleton joints named SSJ to improve the accuracy of activity recognition and reduce the computing cost. SSJ achieves 93.15% of the average recognition accuracy. Existing three activities, such as high arm wave, draw kick, and sit down, achieve the low accuracy of 80%, 75%, and 70%, respectively. Table 4 shows the performance of four methods including CSI-based, KARD-based (skeleton joints), SSJ-based, and HuAc. Table row of the bold font shows that skeleton-based method outperforms CSI-based method on the accuracy of activity recognition. Table row of the italic font shows that several activities are sensitive to CSI. HuAc improves the accuracy of activity recognition and increases the stability of activity recognition in a dynamic indoor environment. We focus

TABLE 3: Mapping relation between WiFi and Kinect.

	WiFi	Kinect
Techniques	CSI	Skeleton joints
Granularity	Subcarriers (15)	Joints (15)
Parameters	Similarity coefficient, median absolute deviation, variance, environment factor	Distance between joints, angle between adjacent joints, variance, sequence of key joints

TABLE 4: Accuracy of activity for CSI-based and Kinect-based.

Activities	WiFi	KARD [3]	SSJ	HuAc
<b>Horizontal arm wave</b>	<b>90%</b>	<b>92%</b>	<b>100%</b>	<b>100%</b>
High arm wave	100%	96%	80%	95%
<b>Two-hand wave</b>	<b>93.1%</b>	<b>96%</b>	<b>100%</b>	<b>100%</b>
<i>High throw</i>	90%	80%	100%	100%
Draw x	100%	96%	100%	93%
<i>Draw tick</i>	100%	90%	75%	93%
<i>Toss paper</i>	100%	90%	100%	100%
<b>Forward kick</b>	<b>87%</b>	<b>96%</b>	<b>100%</b>	<b>100%</b>
Side kick	100%	100%	90%	100%
Bend	95.7%	96%	100%	100%
<b>Hand clap</b>	<b>92%</b>	<b>100%</b>	<b>100%</b>	<b>100%</b>
Walk	100%	100%	100%	100%
Phone	100%	96%	100%	100%
<i>Drink water</i>	100%	86%	100%	100%
<b>Sit down</b>	<b>90%</b>	<b>100%</b>	<b>70%</b>	<b>91%</b>
Squat	96.7%	100%	90%	90%

attention on the stability of activity recognition algorithm or system in the future work.

## 9. Case Study: Motion-Sensing Game Using WiFi Signals

We introduce the application based on our work in the motion-sensing game. At present, Kinect provides the angle with limitations in which the horizontal viewing angle is  $57.5^\circ$  and  $43.5^\circ$  for vertical viewing angle, and distance with limitation ranges from 0.5 m to 4.5 m. Moreover, Kinect loses the sensing ability when barrier occurs and occludes game user in the control area. An interesting point of our work is that we pay more attention to the activity itself, and we do not care about the user location. However, Kinect needs to adjust the location of a user before activity recognition to achieve well sensing. Therefore, we will propose a framework instead of Kinect in the future when the accuracy of human activity recognition using WiFi can satisfy the requirement in an indoor environment.

We list a motion-sensing game using WiFi signals in Figure 14. One or two people are located in the middle of the transmission and receiving terminal and prolong the distance between the TV and user. The area below the blue dashed line represents the control area, and our work can sense human behavior within 10 m and achieve a better performance

in the range of black circle. The user operates the same activity as well as the TV set, and receiving terminal collects corresponding data. By the phase of signals processing, we achieve an activity with the probability and match it with the game of TV set. Once the matching result satisfies the threshold value, activity recognition matches success in the motion-sensing game using WiFi signals.

## 10. Discussion and Future Work

*10.1. Extending to Shadow Recognition.* In our research, we consider the relationship between the WiFi signals and skeleton data on the human activity recognition. Moreover, we describe the interesting topic of the shadow activity recognition. Shadow is an important issue to vision-based activity recognition or monitoring; however, WiFi-based activity recognition can sense human behavior through wall or shadow. First, we explore the characteristics of CSI to enhance the sensing ability by using the high-precision device. Second, WiFi signals can help vision-based activity recognition to improve the ability of sensing environment. In this study, we also need to consider the material attenuation. According to our observations, there is a little difference between the impact of wall reflection and body reflection on the WiFi signals. WiVi [14] leverages the nulling technique to explore the through-wall sensing behavior by using CSI and

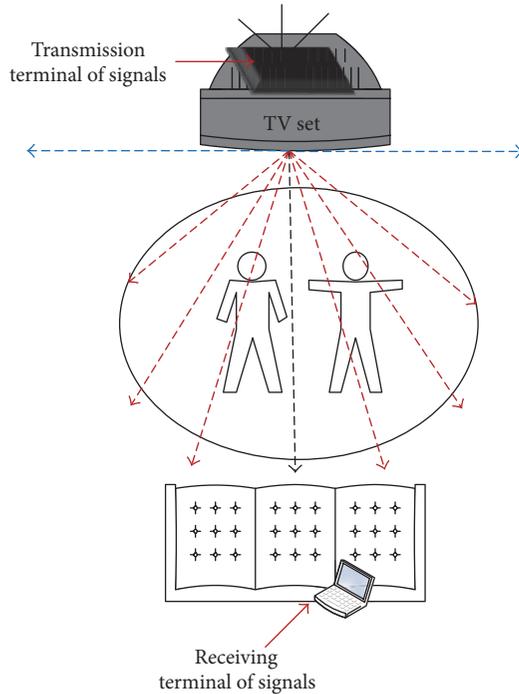


FIGURE 14: Motion-sensing game using WiFi signals.

analyzing the offset of signals from reflection and attenuation of the wall. We recommend researchers to read this paper and their following work [11].

**10.2. Extending to Multiple People Activity Recognition.** Multiple people activity recognition needs multiple APs to obtain more signals information reflected by a human body. At present, existing works can locate target location [46] and detect the number [19] of multiple people using CSI in the indoor environment. Kinect-based activity recognition system recognizes two skeletons (six skeletons for Kinect 2.0) and locates skeletons of six people. Therefore, the combination of WiFi signals and Kinect facilitates the development of multiple people activity recognition. In the future, our team wants to deeply research the character of WiFi signals and propose a novel framework to facilitate the practical application of human activity recognition in the social lives.

**10.3. Data Fusion.** Skeleton data detect the position of each joint for each activity and track the trajectory of human behavior. CSI can sense a fine-grained activity without attaching device in the complex indoor environment. The balance point between CSI and skeleton joints and the selection method of effective features are important factors for improving the quality of fusion information. Moreover, time synchronization of fusion information is also an important challenge in the human activity recognition field.

## 11. Conclusion

In our work, we construct a WiFi-based public activity dataset named WiAR and design HuAc, a novel framework of human activity recognition using CSI and crowdsourced skeleton

joints, to improve the robustness and accuracy of activity recognition. First, we leverage the moving variance of CSI to detect the rough start and end of an activity and adopt the distribution of CSI to describe the detail of each activity. Moreover, we also select several effective subcarriers by using  $K$ -means algorithm to improve the stability of activity recognition. Then, we design SSJ method on the basis of KARD to recognize similar activities by leveraging spatial relationship and the angle of adjacent joints. Finally, we solve the limitations of CSI-based and skeleton-based activity recognition using fusion information. Our results show that HuAc achieves 93% of average recognition accuracy in the WiAR dataset.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

The work is supported by National Natural Science Foundation of China with no. 61733002 and the Fundamental Research Funds for the Central University with no. DUT17LAB16 and no. DUT2017TB02. This work is also supported by Tianjin Key Laboratory of Advanced Networking (TANK), School of Computer Science and Technology, Tianjin University, Tianjin 300350, China.

## References

- [1] L. Chen, J. Hoey, C. D. Nugent, D. J. Cook, and Z. Yu, "Sensor-based activity recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 42, no. 6, pp. 790–808, 2012.
- [2] C. Han, K. Wu, Y. Wang, and L. M. Ni, "WiFall: device-free fall detection by wireless networks," in *Proceedings of the 33rd IEEE Conference on Computer Communications (IEEE INFOCOM '14)*, pp. 271–279, Toronto, Canada, May 2014.
- [3] S. Gaglio, G. Lo Re, and M. Morana, "Human activity recognition process using 3-D posture data," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 5, pp. 586–597, 2015.
- [4] H. Abdelnasser, K. A. Harras, and M. Youssef, "WiGest demo: a ubiquitous WiFi-based gesture recognition system," in *Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPs '15)*, pp. 17–18, IEEE, Hong Kong, May 2015.
- [5] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Computing Surveys*, vol. 46, no. 3, article 33, 2014.
- [6] A. Avci, S. Bosch, M. Marin-Perianu, R. Marin-Perianu, and P. Havinga, "Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: A survey," in *Proceedings of the ARCS 2010*.
- [7] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with microsoft kinect sensor: A review," *IEEE Transactions on cybernetics*, vol. 43, no. 5, pp. 1318–1334, 2013.
- [8] K. Biswas and S. Basu, "Gesture recognition using Microsoft Kinect," in *Proceedings of the 5th International Conference on Automation, Robotics and Applications (ICARA '11)*, pp. 100–103, December 2011.

- [9] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu, "E-eyes: Device-free location-oriented activity identification using fine-grained WiFi signatures," in *Proceedings of the 20th ACM Annual International Conference on Mobile Computing and Networking, MobiCom 2014*, pp. 617–628, USA, September 2014.
- [10] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Understanding and modeling of WiFi signal based human activity recognition," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, MobiCom 2015*, pp. 65–76, Paris, France, September 2015.
- [11] F. Adib, C.-Y. Hsu, H. Mao, D. Katabi, and F. Durand, "Capturing the human figure through a wall," *ACM Transactions on Graphics*, vol. 34, no. 6, article no. 219, 2015.
- [12] J. Wilson and N. Patwari, "See-through walls: motion tracking using variance-based radio tomography networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 5, pp. 612–621, 2011.
- [13] F. Adib, Z. Kabelac, and D. Katabi, "Multi-person localization via RF body reflections," in *Proceedings of the 12th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2015*, pp. 279–292, usa, May 2015.
- [14] F. Adib and D. Katabi, "See through walls with WiFi!," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (ACM SIGCOMM '13)*, pp. 75–86, August 2013.
- [15] B. Fang, N. D. Lane, M. Zhang, A. Boran, and F. Kawsar, "BodyScan: Enabling radio-based sensing on wearable devices for contactless activity and vital sign monitoring," in *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys 2016*, pp. 97–110, Singapore, June 2016.
- [16] Z. Cheng, L. Qin, Y. Ye, Q. Huang, and Q. Tian, "Human daily action analysis with multi-view and color-depth data," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 7584, no. 2, pp. 52–61, 2012.
- [17] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM '00)*, vol. 2, pp. 775–784, Tel Aviv, Israel, March 2000.
- [18] J. Gjengset, J. Xiong, G. McPhillips, and K. Jamieson, "Phaser: Enabling phased array signal processing on commodity WiFi access points," in *Proceedings of the 20th ACM Annual International Conference on Mobile Computing and Networking, MobiCom 2014*, pp. 153–163, USA, September 2014.
- [19] C. Xu, B. Firner, R. S. Moore et al., "Scpl: indoor device-free multi-subject counting and localization using radio signal strength," in *Proceedings of the 12th International Conference on Information Processing in Sensor Networks (IPSN '13)*, pp. 79–90, Philadelphia, Pa, USA, April 2013.
- [20] K. Kleisouris, B. Firner, R. Howard, Y. Zhang, and R. P. Martin, "Detecting intra-room mobility with signal strength descriptors," in *Proceedings of the 11th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2010*, pp. 71–80, USA, September 2010.
- [21] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: wireless indoor localization with little human intervention," in *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking (Mobicom '12)*, pp. 269–280, August 2012.
- [22] L. Sun, S. Sen, and D. Koutsonikolas, "Bringing mobility-awareness to WLANs using PHY layer information," in *Proceedings of the 10th ACM International Conference on Emerging Networking Experiments and Technologies, CoNEXT 2014*, pp. 53–65, Australia, December 2014.
- [23] Y. Zeng, P. H. Pathak, and P. Mohapatra, "Analyzing shopper's behavior through WiFi signals," in *Proceedings of the 2nd Workshop on Physical Analytics, WPA 2015*, pp. 13–18, Italy.
- [24] K. Ali, A. X. Liu, W. Wang, and M. Shahzad, "Keystroke recognition using WiFi signals," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, MobiCom 2015*, pp. 90–102, France, September 2015.
- [25] X. Zheng, J. Wang, L. Shanguan, Z. Zhou, and Y. Liu, "Smokey: Ubiquitous smoking detection with commercial WiFi infrastructures," in *Proceedings of the 35th Annual IEEE International Conference on Computer Communications, IEEE INFOCOM 2016*, USA, April 2016.
- [26] L. Sun, S. Sen, D. Koutsonikolas, and K.-H. Kim, "WiDraw: Enabling hands-free drawing in the air on commodity WiFi devices," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, MobiCom 2015*, pp. 77–89, France, September 2015.
- [27] G. Wang, Y. Zou, Z. Zhou, K. Wu, and L. M. Ni, "We can hear you with Wi-Fi!," in *Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom '14)*, pp. 593–604, Maui, Hawaii, USA, September 2014.
- [28] K. Qian, C. Wu, Z. Zhou, Y. Zheng, Z. Yang, and Y. Liu, "Inferring Motion Direction using Commodity Wi-Fi for Interactive Exergames," in *Proceedings of the the 2017 CHI Conference*, pp. 1961–1972, Denver, Colorado, USA, May 2017.
- [29] Z. Yang, Z. Zhou, and Y. Liu, "From RSSI to CSI: indoor localization via channel response," *ACM Computing Surveys*, vol. 46, no. 2, article 25, 2013.
- [30] X. Hu, T. H. S. Chu, H. C. B. Chan, and V. C. M. Leung, "Vita: a crowdsensing-oriented mobile cyber-physical system," *IEEE Transactions on Emerging Topics in Computing*, vol. 1, no. 1, pp. 148–165, 2013.
- [31] V. Radu and M. K. Marina, "HiMLoc: indoor smartphone localization via activity aware Pedestrian Dead Reckoning with selective crowdsourced WiFi fingerprinting," in *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN '13)*, pp. 1–10, Montbeliard, France, October 2013.
- [32] X. Hu, X. Li, E. C.-H. Ngai, V. C. M. Leung, and P. Kruchten, "Multidimensional context-aware social network architecture for mobile crowdsensing," *IEEE Communications Magazine*, vol. 52, no. 6, pp. 78–87, 2014.
- [33] X. Hu and V. C. M. Leung, "Towards context-aware mobile crowdsensing in vehicular social networks," of *IEEE ISCCGC*, 2015.
- [34] B. Lu, Z. Zeng, L. Wang, B. Peck, D. Qiao, and M. Segal, "Confining Wi-Fi coverage: A crowdsourced method using physical layer information," in *Proceedings of the 13th Annual IEEE International Conference on Sensing, Communication, and Networking, SECON 2016*, UK, June 2016.
- [35] Z. Ning, X. Hu, Z. Chen et al., "A Cooperative Quality-aware Service Access System for Social Internet of Vehicles," *IEEE Internet of Things Journal*, pp. 1-1.
- [36] Z. Ning, F. Xia, N. Ullah, X. Kong, and X. Hu, "Vehicular Social Networks: Enabling Smart Mobility," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 16–55, 2017.

- [37] Z. Ning, X. Wang, X. Kong, and W. Hou, "A Social-aware Group Formation Framework for Information Diffusion in Narrow-band Internet of Things," *IEEE Internet of Things Journal*, pp. 1-1.
- [38] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proceedings of the 18th annual international conference on Mobile computing and networking (Mobicom '12)*, pp. 293–304, August 2012.
- [39] Y. Sungwon, P. Dessai, M. Verma, and M. Gerla, "FreeLoc: calibration-free crowdsourced indoor localization," in *Proceedings of the 32nd IEEE Conference on Computer Communications (INFOCOM '13)*, pp. 2481–2489, Turin, Italy, April 2013.
- [40] Z.-P. Jiang, W. Xi, X. Li et al., "Communicating is crowdsourcing: Wi-Fi indoor localization with CSI-based speed estimation," *Journal of Computer Science and Technology*, vol. 29, no. 4, pp. 589–604, 2014.
- [41] Y. Chen, L. Shu, A. M. Ortiz, N. Crespi, and L. Lv, "Locating in crowdsourcing-based dataspace: Wireless indoor localization without special devices," *Mobile Networks and Applications*, vol. 19, no. 4, pp. 534–542, 2014.
- [42] L. Cheng and J. Wang, "How can I guard my AP? Non-intrusive user identification for mobile devices using WiFi signals," in *Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2016*, pp. 91–100, Germany, July 2016.
- [43] Z. Zhou, Z. Yang, C. Wu, W. Sun, and Y. Liu, "LiFi: Line-Of-Sight identification with WiFi," in *Proceedings of the 33rd IEEE Conference on Computer Communications (INFOCOM '14)*, pp. 2688–2696, May 2014.
- [44] Y. Wang, J. Yang, Y. Chen, H. Liu, M. Gruteser, and R. P. Martin, "Tracking human queues using single-point signal monitoring," in *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys 2014*, pp. 42–54, USA, June 2014.
- [45] H. Wang, D. Zhang, Y. Wang, J. Ma, Y. Wang, and S. Li, "RT-Fall: A Real-Time and Contactless Fall Detection System with Commodity WiFi Devices," *IEEE Transactions on Mobile Computing*, vol. 16, no. 2, pp. 511–526, 2017.
- [46] X. Guo, D. Zhang, K. Wu, and L. M. Ni, "MODLoc: Localizing multiple objects in dynamic indoor environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 11, pp. 2969–2980, 2014.

## Research Article

# A Time and Location Correlation Incentive Scheme for Deep Data Gathering in Crowdsourcing Networks

Fulong Ma,<sup>1</sup> Xiao Liu,<sup>1</sup> Anfeng Liu ,<sup>1</sup> Ming Zhao,<sup>2</sup> Changqin Huang,<sup>3</sup> and Tian Wang <sup>4</sup>

<sup>1</sup>School of Information Science and Engineering, Central South University, Changsha 410083, China

<sup>2</sup>School of Software, Central South University, Changsha 410083, China

<sup>3</sup>School of Information Technology in Education, South China Normal University, Guangzhou 510631, China

<sup>4</sup>College of Computer Science & Technology, Huaqiao University, Xiamen, Fujian Province 361021, China

Correspondence should be addressed to Anfeng Liu; [afengliu@mail.csu.edu.cn](mailto:afengliu@mail.csu.edu.cn)

Received 4 September 2017; Accepted 14 November 2017; Published 9 January 2018

Academic Editor: Zhaolong Ning

Copyright © 2018 Fulong Ma 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.

To tackle the issue in deep crowd sensing, a Time and Location Correlation Incentive (TLCI) scheme is proposed for deep data gathering in crowdsourcing networks. In TLCI scheme, a metric named “Quality of Information Satisfaction Degree” (QoISD) is to quantify how much collected sensing data can satisfy the application’s QoI requirements mainly in terms of data quantity and data coverage. Two incentive algorithms are proposed to satisfy QoISD with different view. The first algorithm is to ensure that the application gets the specified sensing data to maximize the QoISD. Thus, in the first incentive algorithm, the reward for data sensing is to maximize the QoISD. The second algorithm is to minimize the cost of the system while meeting the sensing data requirement and maximizing the QoISD. Thus, in the second incentive algorithm, the reward for data sensing is to maximize the QoISD per unit of reward. Finally, we compare our proposed scheme with existing schemes via extensive simulations. Extensive simulation results well justify the effectiveness of our scheme. The QoISD can be optimized by 81.92%, and the total cost can be reduced by 31.38%.

## 1. Introduction

Internet of Things (IoT) [1–5] and cloud computing [6–13] take advantage of the ubiquity of smart sensor-equipped devices such as smartphones, iPad, and vehicle sensor devices. All of these are to collect information with low cost and provide a new paradigm for solving the complex data sensing based applications from the significant demands of critical infrastructure such as surveillance systems [14–16], remote patient care systems in healthcare, intelligent traffic management, and automated vehicles in transportation environmental [17–19] and weather monitoring systems. In such applications, due to the need to collect a wide range and large amount of data even in a long-term continuous period, the cost of traditional method deploying sensing devices or specialized employee to collect the data is so high limiting the development of applications [20, 21]. Hence, a new method of data collection named crowd sensing (or participatory sensing) is adopted to collect data [21–26]. First, there are a

large number of users who can participate in data collection. Because of the development of electrical devices, for example, the smartphones have been very popular in the past few years [27]. In the final quarter of 2010, it is the first time that smartphone sales passed PCs [28]. Smartphones are equipped with rich sensors which are cheap but useful, such as accelerometer, digital compass, GPS, microphone, gyroscope, and camera. These sensors can monitor a diverse range of human activities and collect large amounts of useful information [23, 28]. These can sense data through their smartphones or smart sensing devices which are collectively referred to as users. Second, the data in this way to collect has a broad range and at a low cost, which can provide effective data for many applications. This data collection is generally carried out as shown in Figure 1. In such a crowdsourcing sensing system, there are four main components. (a) users (also known as crowdsourcing, or participants): they represent the smart sensing devices or people who hold sensing devices, which are to sense data samples, report to applications, and

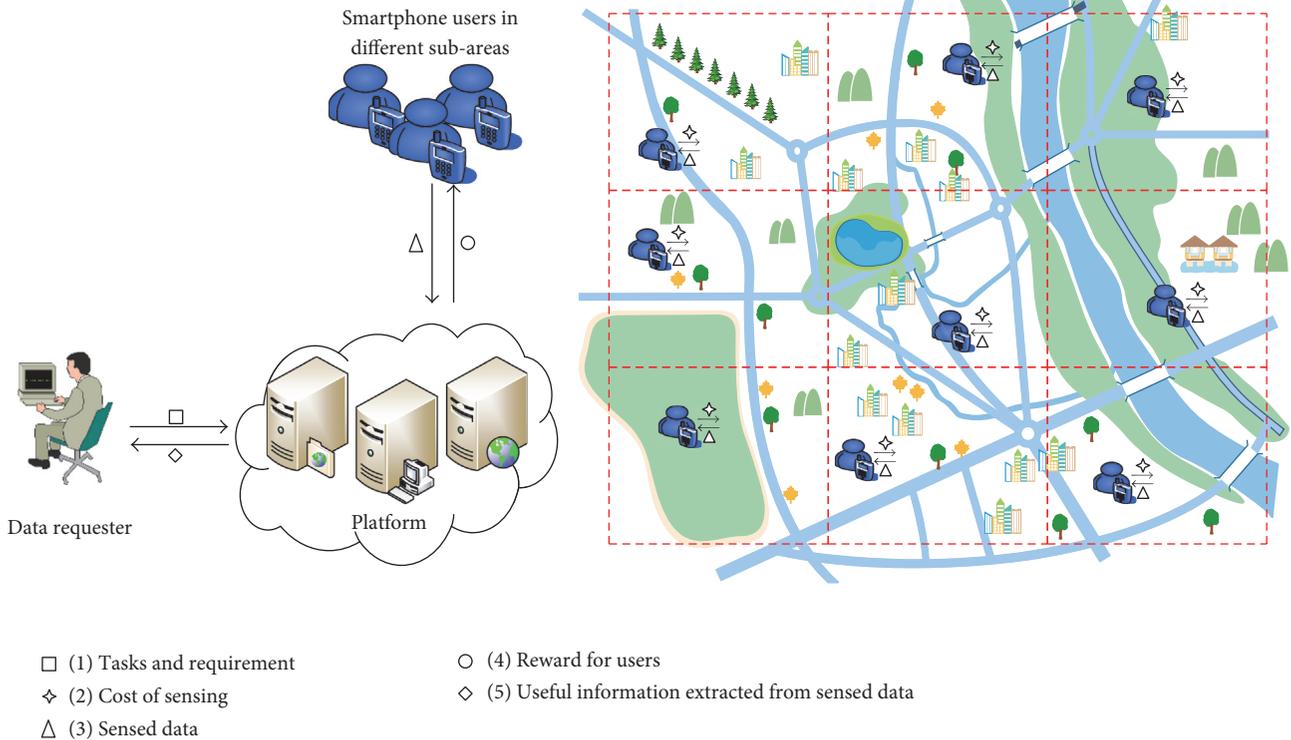


FIGURE 1: A crowdsourcing sensing system.

gain the reward, also known as reporter; (b) applications (or task publisher): application is the demander of data, which publishes the requirements for data and pays a certain reward to users, motivating the users to collect sensing data, and after users collect data and report to applications, applications provide the customers for advanced services based on the collected sensing data and charge a certain fee to make up for the data contribution; (c) customers: the people who use the application's services and pay a certain fee to applications; (d) the third-party process platform: the interaction between users, applications, and customers is not certainly direct and may be carried out through the third-party trusted platform and the third-party trusted interactive platform helping the interaction between the parties becomes more effective and convenient and it has been widely used in many researches and practices. The main body of this paper is the incentive mechanism and interaction between users and applications [23]. The interactive process of crowdsourcing sensing system is shown as follows: the applications publish the attributes, such as area, time period, storage quality, and the reward for reporting data samples [22]. The users in target area collect the sensing data, report the sensed data to applications, and gain the reward. Applications process the collected data, form the services, and provide the customers for services. Because a large number of users are involved in collecting the sensing data, it is fast and of low cost and high quality to complete the applications which cannot be achieved in the past. For example, in the wide-area bird ecology system, in order to obtain the biological habits of the entire bird ecosystem, migration, and the relationship between the food chains, a

large amount of data must be collected in a wide location and for long time to provide effective service. In these large data services, due to the large amount and range, long time period, and even high data acquisition frequency of data collection, the cost is very high if the applications deploy and maintain the data acquisition equipment (or employee) by itself. Thus, after the applications published needed data elements of bird data samples, time, locations, and the reward for the data, a large number of users volunteered to collect the data and report to applications, which can form the services which are of low cost and high quality and have rich sampling data compared to other methods.

It is foreseeable that, with the development of smart sensing devices, the sensing devices will become more powerful and have a wider sensing scope. And crowd sensing based applications will play an increasingly important role in every aspect of mankind in the future, so as to provide more convenient and better user experience for human. However, it depends on the high-quality data samples that the crowd sensing processes the data, provides customers for advanced services, and gets the payoff which can maintain a positive input to the data collection to ensure the sustainable development of applications. For data collection, it is necessary that rewards are introduced into the crowdsourcing sensing system because (1) it is possible that participating in a sensing task may incur monetary costs, the usage of network bandwidth, and the consumption of the power of the smartphones. Thus, the rewards could motivate them to tolerate these costs and contribute to the sensing task. (2) The smartphones are personal which are not controlled by others.

We cannot decide when and how to collect sensor data and it is necessary that rewards can be used to affect their behavior. As a general incentive mechanism, the users who participate in sensing will get monetary reward since they provide the sensed data.

There are many existing studies [29–32] that focus on the incentive mechanism. This paper summarizes the existing research into the following categories.

(1) The incentive mechanisms which are for the purpose of certain number of data samples: the main objective of this type of incentive mechanism is to obtain a sufficient number of data samples. There are many ways in such strategy, such as game based mechanism and demand and supply model based market incentive mechanism.

However, in this type of study, the main consideration is to collect a sufficient number of data samples with the least cost, but without considering or rarely considering the quality of data. It obviously no longer meets the current rapid development of the application demands.

(2) Incentive mechanism based on quality: this kind of research not only considers the quantity and cost of data collection but also takes quality of data collection into account. For example, Tham and Luo [30] propose a metric called Quality of Contributed Service (QCS), and the metric named Quality of Information (QoI) is also proposed by Reddy et al. [31] to evaluate the quality of the data samples. This type of incentive mechanism achieves efficient data collection through selecting high QCS reporters. Data coverage is also studied by Song et al. [32], which recruits the most matching participants to maximize coverage. But the actual performance of these strategies is not stratifying due to three reasons as follows. First of all, these studies tend to think that the participants can be controlled by the applications so that the applications can select or recruit the certain participants to meet the QoS, QoI, or coverage of applications. However, whether users participate in sensing task depends on themselves in practice. Hence, the participants cannot be chosen by the application. Secondly, it always assumes that the number of users involved in sensing data is very large in these applications so that the applications could only consider how to select the certain users meeting the needs of application, which is impossible in many cases. Even if in the downtown area where there are rich sensing devices, if there is no sufficient incentive ability, the users who can participate in sensing data are so few that there are little participants which can be chosen by applications. In this case, the focus of the study is to study how to make a suitable reward to motivate users to participate in sensing task rather than considering how to select certain users. Last but not least, in these studies, even if selecting the data with Quality of Information (QoI) in high priority, it cannot obtain high-quality applications from the overall perspective. The reason is that the quality of data is high from the partial perspective, but, in the overall perspective, there is no improvement on quality of applications while collecting data samples in the area where the collected data is enough, even if the data is of high quality. At the time, the data in the area where the collected data is very few can efficiently improve the quality of applications, even if the data is of low quality. Therefore,

there is local optimization but no overall optimization if data collection only depends on QoI.

In deep crowd sensing (DCS), there are more detailed requests for the data collection [33–38]. First of all, in this aspect of data quality, the demand of sensing data in target area is  $D$ , which means that the application is of high quality if the collected data is not less than  $D$ . Thus, the extra sensing data that is more than  $D$  is redundant. Hence, the system should collect the certain amount of sensor data meeting the demands and it is important that the collected data have a well coverage while the total amount of the collected data is  $D$ . Coverage generally refers to the case where data is distributed evenly in the data collection range. If the target area of data collection is divided into many subareas and every subarea is not large, the best coverage is that the amount of data collected by each subarea is equal. When the total amount of collected data is certain, if the data collected in some subareas is more than needed, it leads to the lacking of data collected in other subareas. As a result, the QoI is low in the subareas that the collected data is less than needed. As the task mentioned above for monitoring bird migration, data collection is needed in specified monitoring area. If there is no collected data in some subareas, it means that the monitoring of bird migration is not comprehensive. And as the monitoring of haze, if there is no collected data in some subareas, it leads to the disappearance of these subareas of haze in the data report.

On the other hand, the data demand  $D$  of system is changing over the time. In order to reduce the cost of data collection, making sure the data demand  $D$  is to ensure that the collected data can reflect the actual situation. Thus, the data demand  $D$  is often changing in different cases. As the haze monitoring mentioned above, if the weather is stable, the haze of the whole city is stable, and the decrease of actual total collected data has little impact on the high-quality services provided by applications, but it can reduce a lot of cost. And in the case of large changes in the weather, it needs data collection with high frequency and large amount of data to ensure that applications meet the requirements of customers. Obviously, this issue also exists in other applications and it lacks the consideration of these cases in previous studies.

For users, it is different between the users in different subareas, which seriously affect the effectiveness of previous studies. Specifically, there are significant differences in the number of users in different subareas, which leads to a serious shortage of incentive mechanisms that provide all users with the same incentive reward. The main reason is two points as follows: (a) the impact on application quality. Generally speaking, the probability of users involved sensing data is higher when gaining more reward, and, on the contrast, it is the same. Thus, in the case of gaining the same reward, the less subareas where the users are, the less available data samples are, which cannot meet the requirements of applications. And data samples in the subareas where there are more users are so many leading to the data redundancy and it even seriously affects the quality of services provided by applications, which has a negative impact on the development of applications. (b) Because the cost in different subareas is different, it has more damage to the quality of services provided by applications.

In practice, the cost in different subareas is different. The general rule is that the cost of sensing data is low in the subareas where there are adequate infrastructures because the convenience of communication bandwidth, net speed, communication costs, and physical traffic reduce the cost of system in these subareas. On the contrary, at the edge of the subareas where network, communication, network, and traffic are inconvenient, the cost of system is high. At the time, combined with the previous reasons, the number of users is large in the low-cost subareas. Thus, in the same incentive reward strategy, it will result in the rich is the richer, and the poor is the poorer, which further deteriorates the performance of the applications. It can be seen that the previous studies cannot solve the problem better, which results in deep crowd sensing.

To tackle those issues in deep crowd sensing, a Time and Location Correlation Incentive (TLCI) scheme is proposed for deep data gathering in crowdsourcing networks in this paper. The goal of TLCI scheme is to collect the appropriate data and to satisfy the Quality of Information (QoI) requirements of applications with minimum budget as much as possible. In TLCI scheme, the data sensing is fine-grained and the reward for data samples is dynamically changing based on the time and location of data sensing, which can effectively improve the system performance. The main contributions of this paper are summarized as follows.

(1) A Time and Location Correlation Incentive (TLCI) scheme is proposed for deep data gathering in crowdsourcing networks. The TLCI scheme is a fine-grained data gathering mechanism which is more efficient than previous strategies to provide high-quality data gathering with lower cost, because our scheme takes the number of users in different time and locations, the cost for sensing data, and the willingness of participating in sensing data into consideration.

(2) A concept named “QoI satisfaction degree” (QoISD) is introduced in this paper to quantify the degree of how collected sensing data can satisfy whole requirements of applications mainly in terms of data quantity and data coverage. The applications of our proposed algorithm show “QoI satisfaction degree” is more effective in evaluating the overall performance contribution of the data samples to the system.

(3) Two incentive algorithms are proposed to satisfy QoISD with different view. The first algorithm is to ensure that the application gets the specified sensing data to maximize the QoISD. So in the first incentive algorithm, the reward for data sensing is to maximize the QoISD. The second algorithm is to minimize the cost of the system while meeting the sensing data requirement and maximizing the QoISD. So the reward for data sensing is to maximize the QoISD per unit of reward in the second incentive algorithm.

(4) Finally, we compare our proposed scheme with existing schemes via extensive simulations. Experimental results show that our distributed incentive mechanism can successfully attain our aim in this work, which is more suitable in real world. Extensive simulation results well justify the effectiveness of our scheme. The QoISD can be optimized by 81.92%, and the total cost can be reduced by 31.38%.

The rest of the paper is organized as follows. We review related work in Section 2. In Section 3, we describe the system model and formulate the problem of our incentive mechanism. Section 4 present the details of our distributed algorithm for achieving the aim. We evaluate the proposed algorithm via simulations in Section 5. Our paper is concluded in Section 6.

## 2. Related Work

Crowd sensing (CS) refers to sensing a wide range of human activities and their surrounding environment by kinds of sensing devices such as smartphones, iPad, or sensor nodes [32, 39–42]. Because the sensing devices is so vast, it is given a new name crowdsourcing networks (CN) [43]. Due to quick development of mobile sensing devices, collecting the data by these devices is so rich, real-time, and low cost that the traditional way of data collection has a fundamental change [29–32, 43]. As a result, there is a huge change in the network [43]. The center of the network computing migrates from the network center to the network edge [44], which is from cloud computing to edge computing [44], then to big data network [45]. The root of these technological developments lies in the collection of high-quality data [29–32]. Thus, how to obtain high-quality data becomes the focus of research.

Due to the cost of sensing, process, and report data, such as storage, energy, and communication, the incentive mechanisms are a key part of these systems and have been studied extensively [29–32, 43]. The incentive mechanism has been applied to crowd sensing with some specialties that are not available in other applications [43]. First, in CS, the number of reporter is large, and most of them do not have identity authentication features. Thus, it does not have the verification functionality that is available in most interactive systems. In traditional crowdsourcing, such as Amazon Mechanical Turk [44], the crowdsourcing platform publishes their tasks and the reward for subtask (or microtask). The worker is required to submit proof of the task after completing the task and crowdsourcing platform will pay the worker for the promised reward. In this way, through the large number of workers it can achieve the huge task which can be broken down into many microtasks. However, this method is difficult to be implemented in the CS because of the following reason. In CS, the task for data collection is called microtask, in which each data sample is so micro so that if using a complex authentication mechanism, its cost and time consumption are far more than the cost of the task itself. Thus, it is not suitable for CS adopting such certification mechanism. Therefore, an effective incentive mechanism to motivate the users to collect data samples is very important for CS [29].

There are many incentive mechanisms that are designed to attract users to participate in sensing task. This paper divides it into the following categories.

(1) The incentive mechanisms which are for the purpose of certain number of data samples: the incentive mechanism is to motivate the users to collect the data through adjusting the reward. The general process is as follows: we assume the amount of data expected to be collected is  $D$ . It will decrease the reward for each data sample to reduce the cost

of applications while the actual collected data is more than the expected. On the contrary, if the current collected data is less than  $D$ , then the current incentive reward is not enough and the enthusiasm of the users to sense data is not high. At the time, the system improves the enthusiasm and makes the collected data reach  $D$  through increasing the reward for data samples. There are many differences for the specific applications. A game based on services price decision (GSPD) model to depict the process of price competition between Services Organizers (SOs, i.e., applications) and entities (users) as well as internal entities (users), which leads to a Pareto-optimal equilibrium point. The auction incentive mechanism is another incentive mechanism which is applied in CS. A reverse auction-based dynamic price (RADP) incentive mechanism was proposed by Lee and Hoh [46]. And a subset of users with lowest reward (prices) are chosen by the service provider that needs sensing data, and their sensor data are purchased at their bid prices. It shows that the dynamic price incentive mechanism can reduce the incentive cost compared with the fixed price. Another reverse auction-based incentive mechanism was designed by Jaimes et al. [47] which considers the locations of the users, the budget constraints, and the sensed coverage. Their incentive schemes can improve the covered area. The Stackelberg game joint auction-based mechanism is also proposed by Yang et al. [48].

(2) Incentive mechanism based on quality: this kind of research not only considers the quantity and cost of data collection but also takes quality of data collection into account. In this type of research, some evaluation norms of quality of data samples are proposed to evaluate the quality of data samples. In some studies to collect data samples for service composition, the concept of Quality of Contributed Service (QCS) is proposed by Tham and Luo [30] to measure the contribution of collected data to the combined service. In addition, in some applications, such as noise mapping, traffic condition reporting, and environmental impact monitoring, the metric named Quality of Information (QoI) is proposed by Wang et al. [29] to evaluate the quality of the data samples. In such studies, it is generally considered that the number of participants involved sensing data is very large. Hence, the goal of the incentive strategy design is to select the participants with highest level of performance assessment (e.g., QCS and QoI) from these users. At the time, these studies focus on minimizing the overall cost, profit, or energy consumption while selecting the high-quality participants, shown by Liu et al. [49–52].

(3) In addition to the above two types of research, recent studies think that only the reward based incentive mechanism does not necessarily achieve good results. One important reason for this is as follows: these studies did not consider the users and collected data credibility. Thus, the recent researchers have proposed a comprehensive consideration of the QoI of data and the collected data credibility. These studies think that the users' reputation represents the collected data credibility. Therefore, the applications selecting users with high reputation on priority can improve the performance of the system. Such a reputation-based incentive

mechanism can be found in Zhang and Van Der Schaar [53] as well as Wang et al. [54].

### 3. The System Model and Problem Statement

*3.1. System Model.* It is considered that a crowdsourcing sensor data gathering system is made up of a platform in the cloud and a large number of smartphone users connecting to the platform. The applications (data requesters) publish the task on the platform to collect the sensor data. In addition, the smartphone users  $U$  who are possibly to participate in the sensing task are denoted as follows:

$$U = \{u_1, u_2, \dots, u_n\}. \quad (1)$$

The participants  $\mathcal{U}$  represent the users who are satisfied with reward for their sensed data contribution and participating in the sensing task. So the set  $\mathcal{U}$  is the subset of the set  $U$ .

$$\mathcal{U} = \{u_1, u_2, \dots, u_n\}. \quad (2)$$

The participants take a cost when they use their smartphone to collect the data, such as the electricity consumption when opening the sensor or their consumed time, which have an impact on whether to participate in task. In order to formulate the users' behavior in the reality, it is necessary to stimulate the user's cost and the probability of participating in sensing task.

*Definition 1* (participants cost). The sensing time  $t_i$  represents the consumed time of participants  $u_i$  for participating in the sensing task. In the sensing time, participants collect data continuously. However, the cost is different in different circumstance and different time. The cost of the users is involved in the consumption of time and power of phone. For example, the cost is simpler in cities than mountains and daytime in the night. And the cost is different in the same location and time even if the sensing time is same. It is mainly because the difference of cost of consumption of phone power is due to the difference of smartphones and sensors. Thus, the cost function of participants is involved in the time and sensing circumstance.

We assume that we need the sensor data in a place  $L$  and in a time period  $T$ . If the sensed data is evenly distributed on the target area and time, it makes sensed data representative and is of high quality. The targeted area  $L$  and the task's period  $T$  will be divided equally, such as  $L = \{L_1, L_2, \dots, L_n\}$ ,  $T = \{T_1, T_2, \dots, T_m\}$ , which form many grids. The size of the grid is related to applications. In general, the number of grids should be enough, and the grid area should be small enough. In this way, the collected data from participants  $u_i$  could represent the data situation for the grid.  $T_i$  represents a short period of time during which the monitored value of the object is stationary. The data collected by sensor nodes during this time period can represent the physical values for this time period. The granularity of time  $T_i$  division is also related to applications. If the application changes substantially, then it requires that  $T_i$  is small. And if the application changes within

a narrow range, then  $T_i$  can take a larger value. For example, in the case of smooth weather, the temperature in an hour has almost no change; then the time granularity of sensing temperature can take a larger value. In this period of time, any data collected during this time period can represent the temperature during this time period. And when the weather changes substantially,  $T_i$  should take a smaller value so that the temperature value sensed in a very short time period is basically equal and can represent the temperature in this time period. If taking a larger value, the difference of the sensed value is very large in different time. Thus, the sensed value in different time cannot represent the temperature in the time period.

And we use  $c_i^j$  to represent the cost in subperiod  $T_j$  and in subarea  $L_i$  caused by circumstance involved in location and time.  $\mathcal{C}_i^j$  represents the cost matrix in the entire sensing time and area.

$$\mathcal{C}_i^j = \begin{bmatrix} c_1^1 & c_1^2 & \cdots & c_1^m \\ c_2^1 & c_2^2 & \cdots & c_2^m \\ \vdots & \vdots & \ddots & \vdots \\ c_n^1 & c_n^2 & \cdots & c_n^m \end{bmatrix} \quad i \in [1, n], j \in [1, m]. \quad (3)$$

We assume that the cost involved in power consumption has a linear relationship with the sensing time, denoted as  $\alpha_i \tau$ , and  $\tau$  is sensing time. In addition,  $\alpha_i$  is user parameter, which depends on the phone of participant and different between different users. Hence, the cost function  $C_i(\tau)$  of sensing time  $\tau$  for participant  $u_i$  is formulated as

$$C_i(\tau) = \alpha_i \tau + c_i^j. \quad (4)$$

*Definition 2* (the probability of participation). The probability of participation  $\delta_i(p)$  of user  $u_i$  represents the probability that the user  $u_i$  participates in the sensing task when the reward is  $r_i$  for the sensing time  $\tau$ . We assume that the users are rational and strategic so that whether they participate in the task depends on the reward and cost. The probability of participation is increasing as the ratio of economic benefits increases. We use the power function  $\delta(r_i)$  of ratio of economic benefits to represent the probability that user  $u_i$  participate in data collection when the reward is  $r_i$ , which is shown in the following:

$$\delta(r_i) = \begin{cases} 0, & r_i \leq C_i(\tau) \\ \left( \frac{r_i - C_i(\tau)}{C_i(\tau)} \right)^\beta - 1, & C_i(\tau) < r_i \leq 2C_i(\tau) \\ 1, & r_i > 2C_i(\tau). \end{cases} \quad (5)$$

And  $\beta$  is the parameter to be measured.

The set of users in subarea  $L_i$  and subperiod  $T_j$  is denoted as follows:

$$U_i^j = \begin{bmatrix} u_1^1 & u_1^2 & \cdots & u_1^m \\ u_2^1 & u_2^2 & \cdots & u_2^m \\ \vdots & \vdots & \ddots & \vdots \\ u_n^1 & u_n^2 & \cdots & u_n^m \end{bmatrix} \quad i \in [1, n], j \in [1, m], \quad (6)$$

where  $u_i^j = \{u_1, u_2, \dots, u_z\} \mid u_k$  is the smartphone user. And  $u_i^j$  represent the set of users in the subperiod  $T_j$  and subarea  $L_i$ . In addition, we use  $|u_i^j|$  to represent the number of users in the subperiod  $T_j$  and subarea  $L_i$ .

$U_i^j$  in (6) represent the set of users in the subperiod  $T_j$  and subarea  $L_i$ . But all users in a grid not necessarily participate in sensing task. Hence, we use  $\mathcal{U}_i^j$  to represent the users who actually participate in the sensing as follows:

$$\mathcal{U}_i^j = \begin{bmatrix} u_1^1 & u_1^2 & \cdots & u_1^m \\ u_2^1 & u_2^2 & \cdots & u_2^m \\ \vdots & \vdots & \ddots & \vdots \\ u_n^1 & u_n^2 & \cdots & u_n^m \end{bmatrix} \quad i \in [1, n], j \in [1, m], \quad (7)$$

where  $u_i^j = \{u_1, u_2, \dots, u_z\} \mid u_k$  is the participants.  $u_i^j$  represent the set of participants in the subperiod  $T_j$  and subarea  $L_i$ . In addition, we use  $|u_i^j|$  to represent the number of participants in the subperiod  $T_j$  and subarea  $L_i$ .

We assume the reward  $R_i^j$  provided by applications is the metric shown in

$$R_i^j = \begin{bmatrix} r_1^1 & r_1^2 & \cdots & r_1^m \\ r_2^1 & r_2^2 & \cdots & r_2^m \\ \vdots & \vdots & \ddots & \vdots \\ r_n^1 & r_n^2 & \cdots & r_n^m \end{bmatrix} \quad i \in [1, n], j \in [1, m], \quad (8)$$

where  $r_i^j$  represent the reward of participants in the subperiod  $T_j$  and subarea  $L_i$ .

Equation (6) shows the users situation in the subperiod  $T_j$  and subarea  $L_i$ . But all users in a grid not necessarily participate in sensing task. In the incentive reward mechanism shown in (8), according to (5), the evaluated participation situation is shown in

$$\mathcal{W}_i^j = \begin{bmatrix} w_1^1 & w_1^2 & \cdots & w_1^m \\ w_2^1 & w_2^2 & \cdots & w_2^m \\ \vdots & \vdots & \ddots & \vdots \\ w_n^1 & w_n^2 & \cdots & w_n^m \end{bmatrix} \quad i \in [1, n], j \in [1, m], \quad (9)$$

where  $w_i^j = \delta(r_i^j) \times u_i^j$ .

The amount of data expected to collect data is shown in

$$D_i^j = \begin{bmatrix} d_1^1 & d_1^2 & \cdots & d_1^m \\ d_2^1 & d_2^2 & \cdots & d_2^m \\ \vdots & \vdots & \ddots & \vdots \\ d_n^1 & d_n^2 & \cdots & d_n^m \end{bmatrix} \quad i \in [1, n], \quad j \in [1, m], \quad (10)$$

where  $d_i^j = |u_i^j| \tau$

The actual amount of data obtained is shown in

$$\mathbb{D}_i^j = \begin{bmatrix} dl_1^1 & dl_1^2 & \cdots & dl_1^m \\ dl_2^1 & dl_2^2 & \cdots & dl_2^m \\ \vdots & \vdots & \ddots & \vdots \\ dl_n^1 & dl_n^2 & \cdots & dl_n^m \end{bmatrix} \quad i \in [1, n], \quad j \in [1, m], \quad (11)$$

where  $d_i^j = |u_i^j| \tau$

*Definition 3* (QoI satisfaction degree, QoISD). In the crowd-sourcing gathering system, the data requester (i.e., applications) publishes the task on the platform to collect the sensor data. Thus, the quality of sensed is very important for data requester. In order to make the sensed data of high quality, it is obvious that the data should have a good distribution on locations and time so that the data is more representative.

In this paper, a metric named ‘‘QoI satisfaction degree’’ (QoISD) is to quantify how much collected sensing data can satisfy the application’s QoI requirements mainly in terms of data coverage.

We assume that the data demand of application is shown in (12). If the actual data  $d_i^j$  is exactly the same as the data demand  $\mathfrak{D}_i^j$ , it shows that the incentive mechanism is efficient, which means there is no redundant data. It is the best situation, and, at the time, the QoISD is highest.

$$\mathfrak{D}_i^j = \begin{bmatrix} \mathfrak{D}_1^1 & \mathfrak{D}_1^2 & \cdots & \mathfrak{D}_1^m \\ \mathfrak{D}_2^1 & \mathfrak{D}_2^2 & \cdots & \mathfrak{D}_2^m \\ \vdots & \vdots & \ddots & \vdots \\ \mathfrak{D}_n^1 & \mathfrak{D}_n^2 & \cdots & \mathfrak{D}_n^m \end{bmatrix} \quad i \in [1, n], \quad j \in [1, m]. \quad (12)$$

However, in practice, the incentive mechanism is difficult to make  $d_i^j = \mathfrak{D}_i^j \mid \forall i \in [1, n], \quad j \in [1, m]$ . Therefore, in this paper, we use  $\mathfrak{E}^j$  in (13) to represent the difference between the collected data and the data demand, which is QoISD. Obviously, when QoISD  $\mathfrak{E}^j \rightarrow 0$ , it is the best situation.

$$\mathfrak{E}^j = \sum_{k=1}^n (d_k^j - \mathfrak{D}_k^j)^2. \quad (13)$$

If the application requires the same amount of data for each grid, which is a constant  $\mathfrak{D}$ , then (13) is transformed into the following:

$$\mathfrak{E}^j = \sum_{k=1}^n (d_k^j - \mathfrak{D})^2. \quad (14)$$

### 3.2. Problem Statement

(1) *Sensor Data to Meet the Requirement.* Sensing data to meet the applications requirement contains two aspects of meaning: (a) considering the total data requirement is  $D_{\text{total}}^j$  in subperiod  $T_j$ . Therefore, the total amount of data collected in the target area must be greater than  $D_{\text{total}}^j$ , as in (16):

$$\sum_{i=1}^n d_i^j \geq D_{\text{total}}^j \mid \forall j \in [1, m]. \quad (15)$$

(b) The amount of data collected for each grid  $L_i$  should be greater than the amount of data required for application, which is shown in

$$d_i^j \geq \mathfrak{D}_i^j \mid \forall i \in [1, n], \quad j \in [1, m]. \quad (16)$$

(2) *Maximize the Quality of Data.* As mentioned above, in order to get high quality of sensor data, we should make the data’s distribution best based on time and locations, which means optimizing the QoISD, as follows:

$$\min \quad (\mathfrak{Q}^j) = \min \left( \sum_{k=1}^n (d_k^j - \mathfrak{D}_k^j)^2 \right) \mid \forall j \in [1, m]. \quad (17)$$

(3) *Minimize the Cost for Collection Data.* The aim of incentive mechanism is to minimize the cost of system and optimize the quality of the collected data. Therefore, an important goal of Time and Location Correlation Incentive (TLCI) scheme is to minimize the cost for data collection.

$$\min \quad (\mathfrak{C}^j) = \min \left( \sum_{k=1}^n (r_k^j u_k^j) \right) \mid \forall j \in [1, m]. \quad (18)$$

Thus, the overall goal of TLCI scheme is as follows:

$$\begin{aligned} \min \quad (\mathfrak{Q}^j) &= \min \left( \sum_{k=1}^n (d_k^j - \mathfrak{D}_k^j)^2 \right) \mid \forall j \in [1, m], \\ \min \quad (\mathfrak{C}^j) &= \min \left( \sum_{k=1}^n (r_k^j u_k^j) \right) \mid \forall j \in [1, m], \end{aligned} \quad (19)$$

$$\text{s.t.} \quad \sum_{i=1}^n d_i^j \geq D_{\text{total}}^j \mid \forall j \in [1, m],$$

$$d_i^j \geq \mathfrak{D}_i^j \mid \forall i \in [1, n], \quad j \in [1, m].$$

## 4. Incentive Mechanism Design

In order to state the parameter of this paper clearly, the main notions introduced in this paper can be found in Parameters’ Description.

As shown in Figure 2, we propose an incentive mechanism in this work to get enough sensor data which is to optimize the QoISD. In our incentive mechanism, the platform sends the task’s information to every user, and

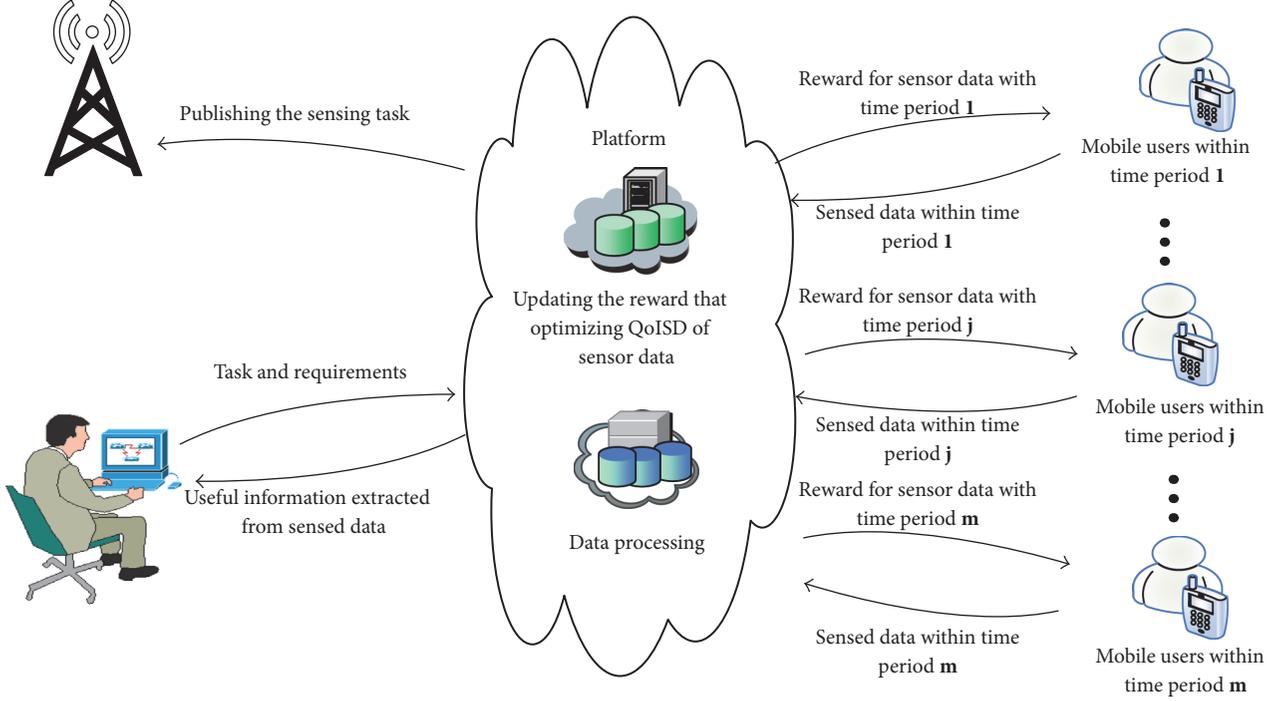


FIGURE 2: An overview of our incentive mechanism.

whether users participate in the sensing task depends on their own cost and the published reward. We assume that the total lasting time period of sensing task is divided into some time periods. And in every subperiod, the target area where data is expected to be obtained is divided into some subareas and the platform publishes a new reward for participants' contribution, which is different for participants in different subareas to optimize the QoISD. It is repeated until the sensing task is completed.

In the incentive mechanism, we use two reward allocation algorithms to optimize the QoISD and reduce the cost of system and both of the algorithms are based on greedy strategies. One algorithm is to solve the problem of optimizing the QoISD when attaining the task's requirement. The other algorithm is improved on the basis of the first algorithm, which is to take the less cost on rewarding the participants and has a better speed than the former. The two incentive algorithms are discussed as follows.

**4.1. Distributed Algorithm for Optimizing QoISD.** The aim in this work is to find a better way of reward allocation so that the system could gain enough high-quality sensed data. The aim in first algorithm is to optimize the QoISD while meeting data requirements that the sensed data in subperiod  $T^j$  reach the data demand  $D_{\text{total}}^j$ , which is formulated as

$$\begin{aligned} \min \quad & (\mathcal{E}^j) = \min \left( \sum_{k=1}^n (d_k^j - \mathfrak{D}_k^j)^2 \right) \mid \forall j \in [1, m] \\ \text{s.t.} \quad & \sum_{i=1}^n d_i^j \geq D_{\text{total}}^j \mid \forall j \in [1, m]. \end{aligned} \quad (20)$$

The algorithm mainly motivates the smartphone users to participate in sensing task through adjusting the reward allocation for users in every subarea  $R^j = \{r_1^j, r_2^j, \dots, r_n^j\}$ . We first analyze the optimal results of sensed data. If the total amount of sensor data needed for applications is  $D_{\text{total}}^j$ , and the amount of sensed data in every subarea is equal which is formulated as  $\mathcal{A}^j = \{\mathfrak{D}_1^j, \mathfrak{D}_2^j, \dots, \mathfrak{D}_n^j\}$ , it requires  $\mathfrak{D}_i^j \triangleq \mathfrak{D}_k^j \mid \forall i, k \in [1, n]$ . In this case, the optimal incentive mechanism is to adjust the reward vector  $R^j$  so that making the amount of sensed data in every subarea is  $D_{\text{total}}^j/n$  exact and the value of QoISD is zero which is optimal. Similarly, if there is no requirement that  $\mathfrak{D}_i^j \triangleq \mathfrak{D}_k^j \mid \forall i, k \in [1, n]$ , and the data demand is  $\mathcal{A}^j = \{\mathfrak{D}_1^j, \mathfrak{D}_2^j, \dots, \mathfrak{D}_n^j\}$ , we assume the actual sensed data is  $D^j = \{d_1^j, d_2^j, \dots, d_n^j\}$ . The optimal incentive strategy is to adjust the reward vector  $R^j$  to optimize the QoISD the same as above. Obviously, it should increase the reward in the subarea where sensed data is less than demand and the difference of both is rather large to motivate the users' participation and makes the QoISD optimal. On the contrary, it optimizes the QoISD decreasing the reward in the subarea where sensed data is more than demand. As shown in (21),  $\sigma_i^j$  represents that deviation rate between actual sensed data and demand in subarea  $L_i$  and subperiod  $T_j$ .

$$\sigma_i^j = \frac{(\mathfrak{D}_i^j - d_i^j)}{\mathfrak{D}_i^j}. \quad (21)$$

Obviously, when  $\sigma_i^j$  is positive it means the actual sensed data is less than demand and it needs to raise the reward  $r_i^j$ . In

addition, the larger  $\sigma_i^j$ , the more amount the lacked data and the increased reward  $r_i^j$ . On the contrary, when  $\sigma_i^j$  is negative, it means the actual sensed data is more than demand and it should decrease the reward  $r_i^j$  to reduce the cost of the system.

Our incentive algorithm is to find the optimal reward allocation for every subarea. The mechanism algorithm iteratively raises the reward to the users in the subarea that could optimize QoISD  $\mathfrak{C}^j$  until the sensed data is enough to the sensing system's demand and the QoISD is optimal. The detailed descriptions are provided as follows.

*Step 1.* First, according the information of sensing task, we initialize the reward of every subarea to  $r_0$  and calculate the new amount of sensor data  $D_{\text{real}}^j$ .

*Step 2.* Second, we iteratively adjust the reward vector to optimize the QoISD  $\mathfrak{C}^j$ . In each iteration, the algorithm adjusts the reward in a subarea which makes QoISD optimal. We use  $\xi$  to mark the target subarea and  $r_\xi$  to mark the increased reward, which is negative when the actual sensed data in target area is more than demand. In addition,  $\varepsilon_{\min}$  represents the optimal QoISD currently and the algorithm updates the value of  $\varepsilon_{\min}$  in every iteration. Hence, in the end of every iteration, it updates the reward  $r_\xi^j$  in target subarea and the amount of sensed data. In addition, the algorithm ignores the subarea where there is no increased amount of sensor data, as shown in Steps (19)-(20).

*Step 3.* The sensed data  $D_{\text{real}}^j$  is updated in every iteration, and the algorithm ends when  $D_{\text{real}}^j$  reaches the data demand  $D_{\text{total}}^j$ . In the end of this incentive algorithm, it returns the reward vector, which makes the QoISD optimal.

The proposed algorithm is given in Algorithm 1.

*4.2. Distributed Algorithm for Optimizing the QoISD and Reducing the Cost.* Based on Algorithm 1, in this subsection, we propose another algorithm for taking the platform's cost and QoISD into account. The aim in Algorithm 2 is to optimize the QoISD and reduce the cost of system while meeting data requirements that the sensed data within subperiod  $T^j$  reaches the data demand  $D_{\text{total}}^j$ , which is shown in

$$\begin{aligned} \min \quad & (\mathfrak{C}^j) = \min \left( \sum_{k=1}^n (d_k^j - \mathfrak{D}_k^j)^2 \right) \mid \forall j \in [1, m] \\ \min \quad & (\mathfrak{C}^j) = \min \left( \sum_{k=1}^n (r_k^j \mathbf{u}_k^j) \right) \mid \forall j \in [1, m] \\ \text{s.t.} \quad & \sum_{i=1}^n d_i^j \geq D_{\text{total}}^j \mid \forall j \in [1, m] \\ & d_i^j \geq \mathfrak{D}_i^j \mid \forall i \in [1, n], j \in [1, m]. \end{aligned} \quad (22)$$

It is apparent that we could solve it by weighted sum method since it is a multiobjective optimization problem. By

selecting scalar weights parameters  $\lambda$  and  $1-\lambda$  for  $\mathfrak{C}^j$  and  $\mathfrak{C}^j$ , the target function is shown in the following:

$$\begin{aligned} \min \quad & (\lambda \mathfrak{C}^j + (1-\lambda) \mathfrak{C}^j) \mid \forall j \in [1, m] \\ \text{s.t.} \quad & \sum_{i=1}^n d_i^j \geq D_{\text{total}}^j \mid \forall j \in [1, m] \\ & d_i^j \geq \mathfrak{D}_i^j \mid \forall i \in [1, n], j \in [1, m], \end{aligned} \quad (23)$$

where  $0 \leq \lambda \leq 1$ . And we could adjust the value of  $\lambda$  to achieve our purpose.

The idea of Algorithm 1 is to raise the reward to the users in the subarea that could optimize QoISD  $\mathfrak{C}^j$  iteratively until the sensed data is enough to the sensing system's demand and the QoISD is optimal. In order to reduce the cost of system, the reward should be allocated to the cost-effective subarea, where system could gain more sensor data with low cost. For each subarea  $L_k$ , we use the  $\Upsilon_k$  to represent how cost-effective it is, which is the ratio of increased data to increased cost and formulated as

$$\Upsilon_k = \frac{(\delta(r_k + r_\nabla) - \delta(r_k)) \mathbf{u}_k^j \tau}{\mathbf{u}_k^j \times (\delta(r_k + r_\nabla) \times (r_k + r_\nabla) - \delta(r_k) x r_k)}. \quad (24)$$

Hence, there is a main alternation in Algorithm 1:

$$\varepsilon_{\min} = -\lambda \Upsilon_k + (1-\lambda) \sum_{i=1}^n (d_i^j - \mathfrak{D}_i^j)^2. \quad (25)$$

The proposed algorithm is given in Algorithm 2.

## 5. Performance Analysis and Optimization

*5.1. Methodology and Setup.* In this section, we compare our two algorithms, Time and Location Correlation Incentive Algorithm 1 (TLCI1) and Time and Location Correlation Incentive Algorithm 2 (TLCI2) with the pricing based fixed price algorithm (PPA). The main character of this algorithm (PPA) is that the participants get the same reward for the same amount of sensor data. We assume the smartphone users are rational and strategic so that whether they participate in sensing task depends on their cost and reward. In order to show how they are similar or different, the parameters of these are the same.

We first study the situation of sense data and the cost of system for rewarding the participants at different locations in 5.2. Then in 5.3, we study the sensed data and the cost in different time in two situations and make a performance comparison of the three algorithms. In addition to the performance comparison based on different time and locations, we also study the quality of sensed data and the stability of algorithms in 5.4 and 5.5, respectively.

The default values of all parameters in our model are set as follows. The default units of reward and sensing time are dollar and minute, respectively. And the user-specific parameters  $\alpha_i$  are uniformly distributed random values within (1, 3). In addition, according to the data obtained from the

```

Input: The data demand  $\mathcal{D}^j = \{\mathfrak{D}_1^j, \mathfrak{D}_2^j, \dots, \mathfrak{D}_n^j\}$ ;
Output: The new vector of reward  $R^j = \{r_1^j, r_2^j, \dots, r_n^j\}$ .
(1) For each  $r_i^j \mid i \in [1, n]$ 
(2)    $r_i^j = r_0$  // to initialize the reward of every sub-area to  $r_0$ 
(3) End for
(4)  $D_{\text{real}}^j = \sum_{k=1}^n d_k^j$ 
(5) While  $D_{\text{real}}^j < D_{\text{total}}^j$  Do
(6)    $\mathfrak{C}_u^j = \sum_{k=1}^n (d_k^j - \mathfrak{D}_k^j)^2$ ;
      //  $\mathfrak{C}_u^j$  is the QoISD before the update
(7)    $\xi = 0$  //to mark the selected grid which is not determined
(8)    $\varepsilon_{\min} = +\infty$  //to store the minimum QoISD
(9)   For  $k = 1$  to  $n$  Do //each grid  $L_k$ 
(10)    If  $(d_k^j - \mathfrak{D}_k^j) < 0$ 
(11)      $r_{\nabla} = +\zeta$ 
           //to increase the reward when the sensed data doses not meet the demand
(12)    Else
(13)      $r_{\nabla} = -\zeta$ 
           //to decrease the reward when the sensed data doses not meet the demand
(14)    End if
(15)     $\overline{d}_k^j = d_k^j + (\delta(r_k + r_{\nabla}) - \delta(r_k)) u_k^j \tau$ 
           //the sensed data in grid  $L_k$  after updating the reward
(16)     $d_k^j \leftarrow \overline{d}_k^j$ 
(17)     $\mathfrak{C}_{\text{new}}^j = \sum_{i=1}^n (d_i^j - \mathfrak{D}_i^j)^2$ 
           //  $\mathfrak{C}_{\text{new}}^j$  is the updated QoISD
(18)    If  $\mathfrak{C}_{\text{new}}^j = \mathfrak{C}_u^j$  //if there is no improved QoISD
(19)     Goto Step (10)
(20)    End if
(21)    If  $\mathfrak{C}_{\text{new}}^j < \varepsilon_{\min}$ 
(22)      $\varepsilon_{\min} = \mathfrak{C}_{\text{new}}^j$ 
(23)      $\xi = k$ ; //to select the  $L_k$  that makes QoISD optimal
(24)      $r_{\xi} = r_{\nabla}$ 
(25)    End if
(26)    End for
(27)     $r_{\xi}^j = r_{\xi}^j + r_{\xi}$ 
(28)     $D_{\text{real}}^j = \sum_{k=1}^n d_k^j$  //to update the sensed data
(29) End while
(30) Return the new reward  $R^j$ 

```

ALGORITHM 1: The QoISD optimization algorithm.

questionnaire survey, the probability parameter  $\beta$  is 1.745. All simulations under the same setting are repeated ten times to get the average values.

**5.2. The Performance of Sensed Data and Cost at Different Locations.** We first study the performance of the three algorithms under different situations. We mainly study two scenarios, the active and the inactive period of users. In the active period, there are large number of users distributed in the target area. On the contrary, fewer people are involved in the sensing task in the inactive period. The study on two scenarios is shown as follows.

**5.2.1. The Active Period of Users.** In the active period of users, there are large number of users that could participate in the sensing task and have plenty of free time to collect the sensor data. We assume that the platform's demand is 4000 (Mb) and

the number of users of every subarea is uniformly distributed random values within (50, 100). The results are as follows. As shown in Figures 3, 4, and 5, the range of sensed data in the subareas in TLCI1 and TLCI2 are similar and the range of sensed data in our two TLCI algorithms varying from 45 to 55 is less than that in PPA, which is varying from 40 to 75 and has a negative impact on the quality of sensed data. The result shows that the variance of sensed data can be reduced by 68.85% and 52.50% in TLCI1 and TLCI2, respectively. We also stimulate the participants' reward in Figures 6 and 7 about our two TLCI algorithms. Compared to the same reward for every user in PPA, the reward varying from 3.7 to 7 is different for each user in our two TLCI algorithms.

**5.2.2. The Inactive Period of Users.** In the inactive period of users, the users do not have enough time to participate in the sensing task. Thus, the number of users is less than the

**Input:** The data demand  $\mathcal{A}^j = \{\mathfrak{D}_1^j, \mathfrak{D}_2^j, \dots, \mathfrak{D}_n^j\}$ ;  
**Output:** The new vector of reward  $R^j = \{r_1^j, r_2^j, \dots, r_n^j\}$ .

- (1) **For** each  $r_i^j \mid i \in [1, n]$
- (2)  $r_i^j = r_0$  // to initialize the reward of every sub-area to  $r_0$
- (3) **End for**
- (4)  $D_{\text{real}}^j = \sum_{k=1}^n d_k^j$
- (5) **While**  $D_{\text{real}}^j < D_{\text{total}}^j$  **Do**
- (6)  $\mathfrak{C}_u^j = \sum_{k=1}^n (d_k^j - \mathfrak{D}_k^j)^2$ ;  
//  $\mathfrak{C}_u^j$  is the QoISD before the update
- (7)  $\xi = 0$  //to mark the selected grid which is not determined
- (8)  $\varepsilon_{\text{min}} = +\infty$  //to store the minimum QoISD
- (9) **For**  $k = 1$  to  $n$  **Do** //each grid  $L_k$
- (10) **If**  $(d_k^j - \mathfrak{D}_k^j) < 0$
- (11)  $r_{\nabla} = +\varsigma$   
//to increase the reward when the sensed data doses not meet the demand
- (12) **Else**
- (13)  $r_{\nabla} = -\varsigma$   
//to decrease the reward when the sensed data doses not meet the demand
- (14) **End if**
- (15)  $\overline{d}_k^j = d_k^j + (\delta(r_k + r_{\nabla}) - \delta(r_k)) u_k^j \tau$   
//the sensed data in grid  $L_k$  after updating the reward
- (16)  $d_k^j \leftarrow \overline{d}_k^j$
- (17)  $\mathfrak{C}_{\text{new}}^j = \sum_{i=1}^n (d_i^j - \mathfrak{D}_i^j)^2$   
//  $\mathfrak{C}_{\text{new}}^j$  is the updated QoISD
- (18) **If**  $\mathfrak{C}_{\text{new}}^j = \mathfrak{C}_u^j$  //if there is no improved QoISD
- (19) **Goto** Step (10)
- (20) **End if**
- (21) **If**  $\mathfrak{C}_{\text{new}}^j < \varepsilon_{\text{min}}$
- (22)  $\varepsilon_{\text{min}} = -\lambda Y_k + (1 - \lambda) \mathfrak{C}_{\text{new}}^j$
- (23)  $\xi = k$ ; //to select the  $L_k$  that make QoISD optimal
- (24)  $r_{\xi} = r_{\nabla}$
- (25) **End if**
- (26) **End for**
- (27)  $r_{\xi}^j = r_{\xi}^j + r_{\xi}$
- (28)  $D_{\text{real}}^j = \sum_{k=1}^n d_k^j$  //to update the sensed data
- (29) **End while**
- (30) **Return** the new reward  $R^j$

ALGORITHM 2: The QoISD optimization and minimizing the cost algorithm.

number of users in the active period. We assume that the number of users in every subarea is uniformly distributed random values within (30, 60). As shown in Figures 8, 9, and 10, we could draw the same conclusion as that in active period. In the inactive period, the variance of sensed data can be reduced by 90.85% and 79.06% in TLCI1 and TLCI2, respectively. In addition, the situation of reward allocation is shown in Figures 11 and 12 that the reward allocation of two algorithms is similar.

*5.3. The Performance of Sensed Data and Cost at Different Time.* It is considered that the variation of the number of users in the whole day has an objective law, which is shown as Figure 13. The users prefer to participate in the sensing task in the daytime and the number of users in daylight is far less

than in daytime. For a more comprehensive consideration, we consider the whole QoISD that involves the data at different times rather than the QoISD at a time period. Therefore, when the impact of incentive reward on the whole QoISD is too little, it will reduce the data demand and increase to other time periods according to the number of users. It is to ensure that the collected data of the whole day meet the demand.

In this subsection, we study the performance of three algorithms in two situations: the random distribution of users and the concentrated distribution of users. In situation of the random distribution of users, the users are evenly distributed at different subareas. We assume that the total users at a time is  $n_r$ , and the users at a subarea are uniformly distributed random values within  $(n_r/m-30)$  and  $(n_r/m+30)$  when there are  $m$  subregions at the time. In the concentrated distribution of users, the 50% of users focus on the 30% of central area.

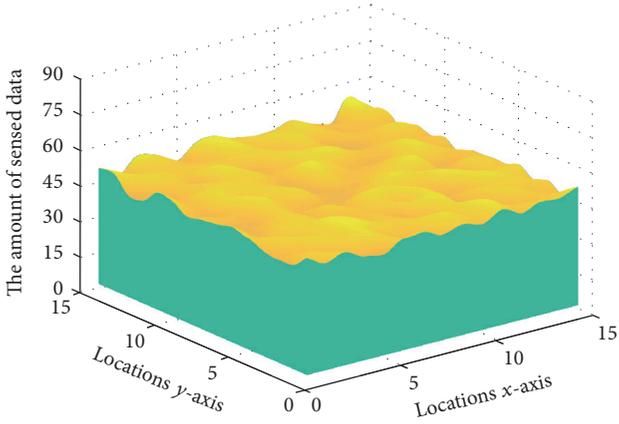


FIGURE 3: The distribution of sensed data of TLCI1 in active period of users.

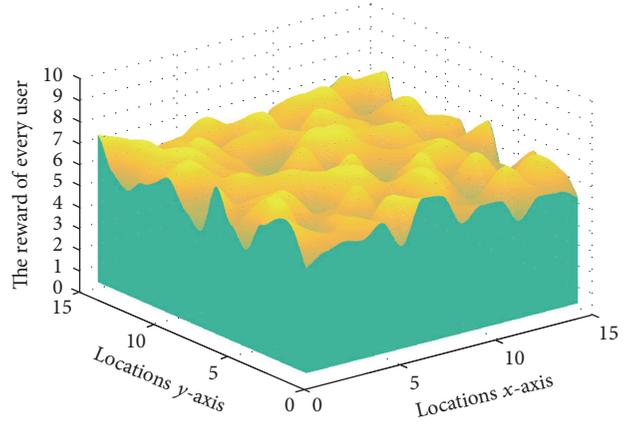


FIGURE 6: The distribution of users' reward of TLCI1 in active period of users.

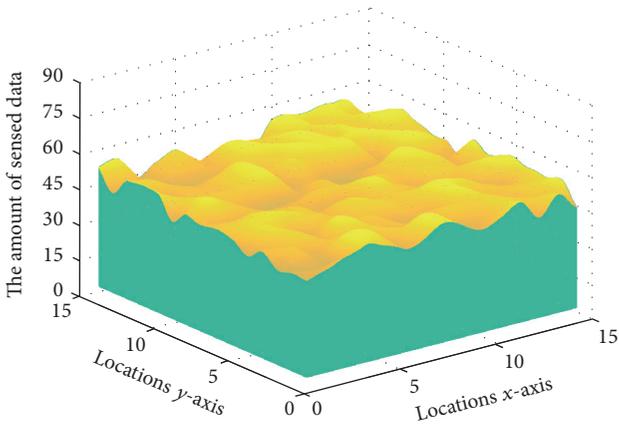


FIGURE 4: The distribution of sensed data of TLCI2 in active period of users.

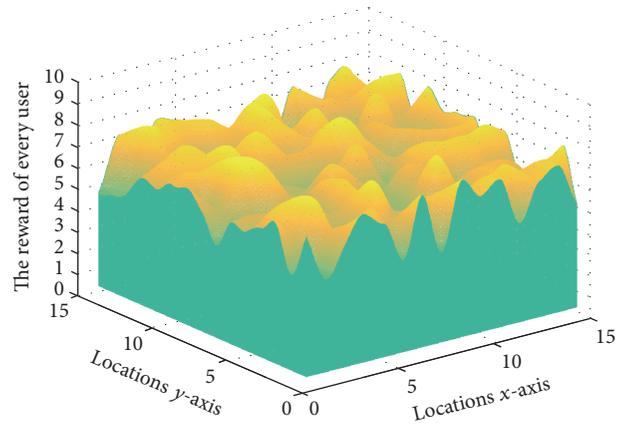


FIGURE 7: The distribution of users' reward of TLCI2 in active period of users.

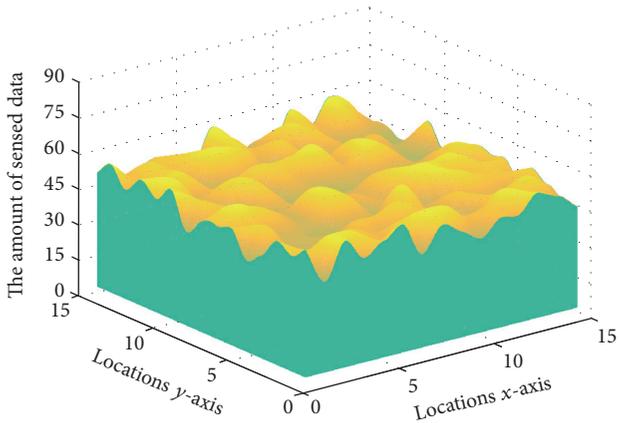


FIGURE 5: The distribution of sensed data of PPA in active period of users.

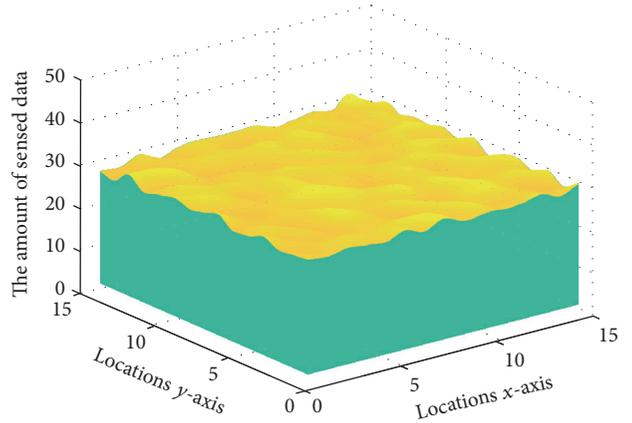


FIGURE 8: The distribution of sensed data of TLCI1 in inactive period of users.

5.3.1. *The Random Distribution of Users.* In the random distribution of users, the users are evenly distributed at the locations at a time. In this subsection, we stimulate the average reward for users and total reward budget for users' contribution at different time, as shown in Figures 14 and 16. It is obvious that, compared to fixed reward in PPA, our

two TLCI algorithms have a better adaptability, adjusting the user's reward according to the number of users. The overall trend is that the average of reward is declining with more users as shown in Figure 14, which can be interpreted as the payment in exchange for participation. As a result, the number of participants and the total reward at different time

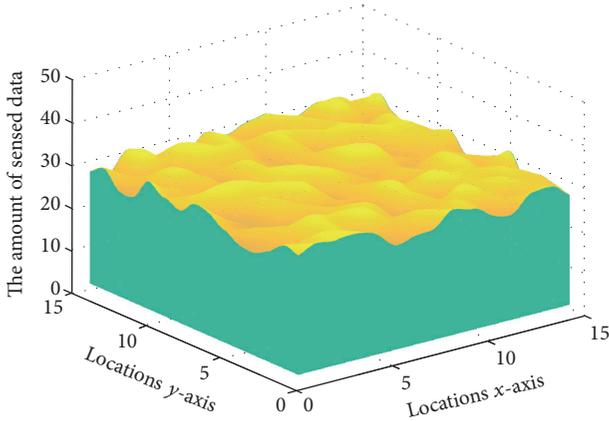


FIGURE 9: The distribution of sensed data of TLCI2 in inactive period of users.

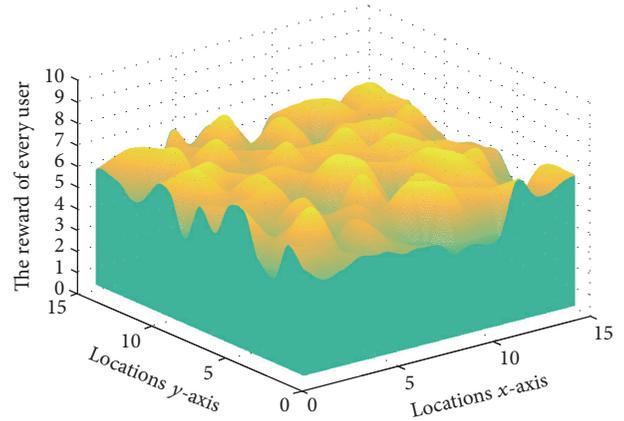


FIGURE 12: The distribution of users' reward of TLCI2 in inactive period of users.

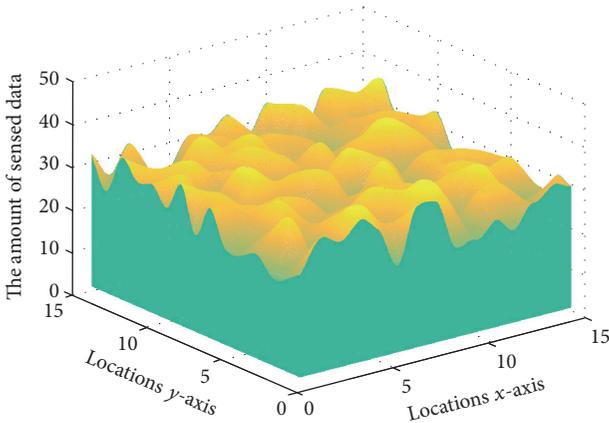
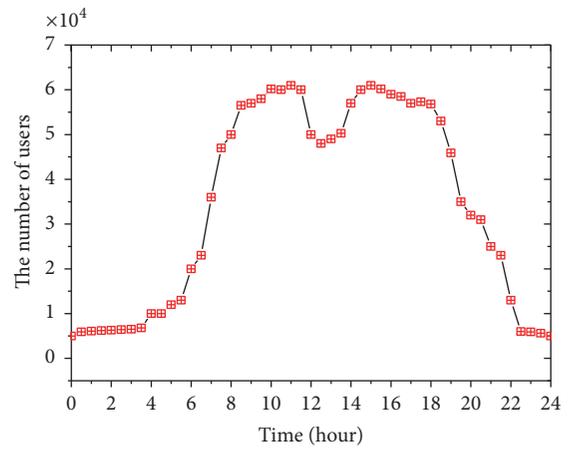


FIGURE 10: The distribution of sensed data of PPA in inactive period of users.



—■— The variation of users

FIGURE 13: The number of users at different time.

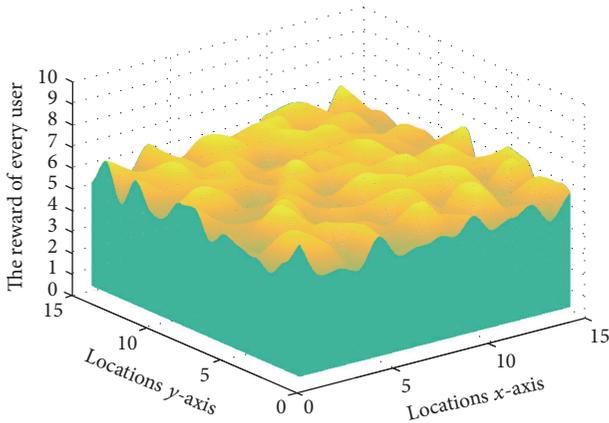
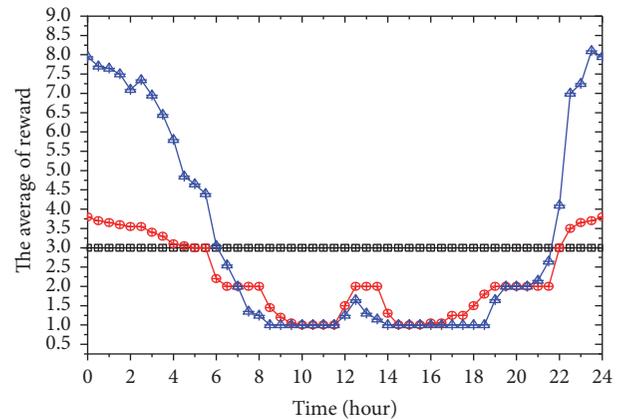


FIGURE 11: The distribution of users' reward of TLCI1 in inactive period of users.



—■— PPA  
—●— GCMA2  
—▲— GCMA1

FIGURE 14: The average of reward for a user at different time.

are shown in Figures 15 and 16, in which total reward and total participants grow up in number when there are few users so that it can collect more sensor data. In addition, we can see the total cost of system is reduced by 19.97% and 42.79% in TCL11 and TCL12, respectively. Figure 17 shows the cumulative total reward over the time, which proves the

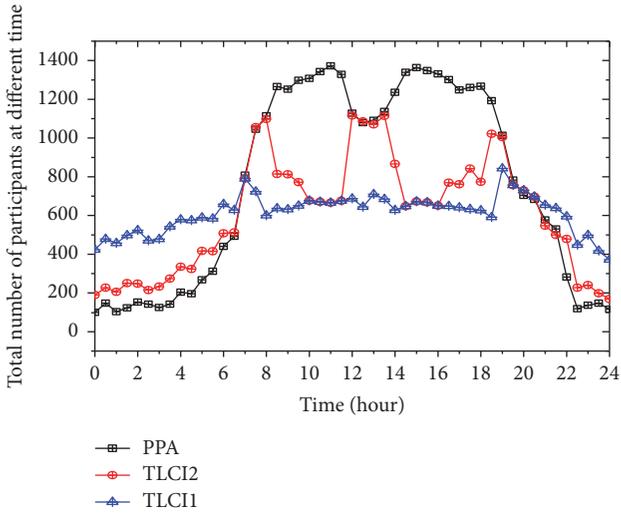


FIGURE 15: The number of participants at different time.

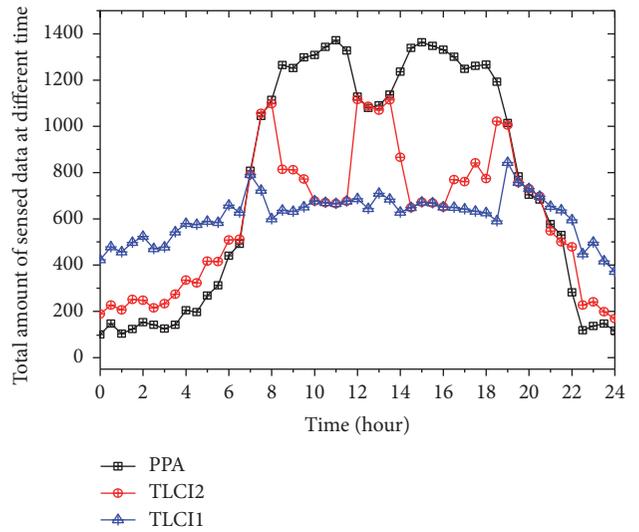


FIGURE 18: The total amount of sensed data.

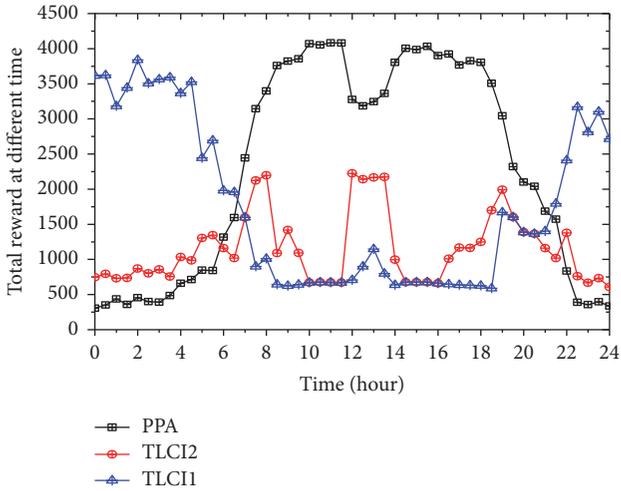


FIGURE 16: The total reward at different time.

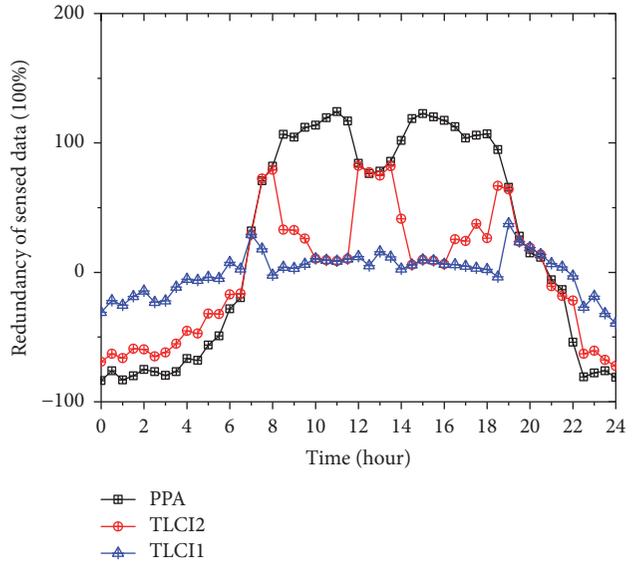


FIGURE 19: The redundancy of sensed data.

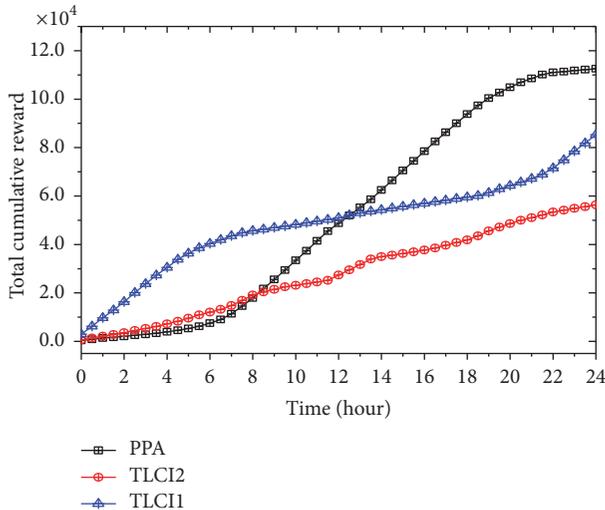


FIGURE 17: The total cumulative reward over time.

effectiveness of our two algorithms and the TLCI2 could reduce the cost. In addition to the total reward, we also study the sensed data over the time in Figure 18. Figure 19 shows that the redundancy of sensed data over time that the percentage of the difference between actual and expected sensed data to expected data. It is similar to the sensed data over time. And in Figure 20, we calculate the cumulative sensed data over time and we can see the sensed data in PPA exceeds the demand. It shows the PPA algorithms have a bad adaptability compared with TLCI algorithms.

**5.3.2. The Concentrated Distribution of Users.** In the situation of the concentrated distribution of users, the 50% of users focus on the 30% of central area. In this subsection, we stimulate the average reward of users and total reward budget for users' contribution at different time, as shown in Figures

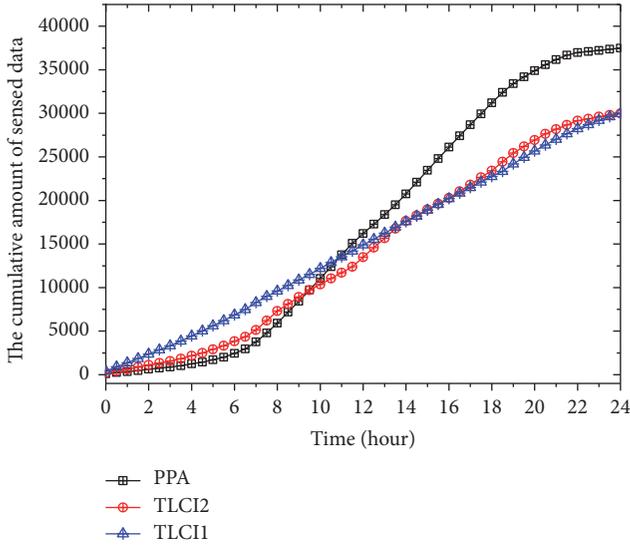


FIGURE 20: The cumulative amount of sensed data.

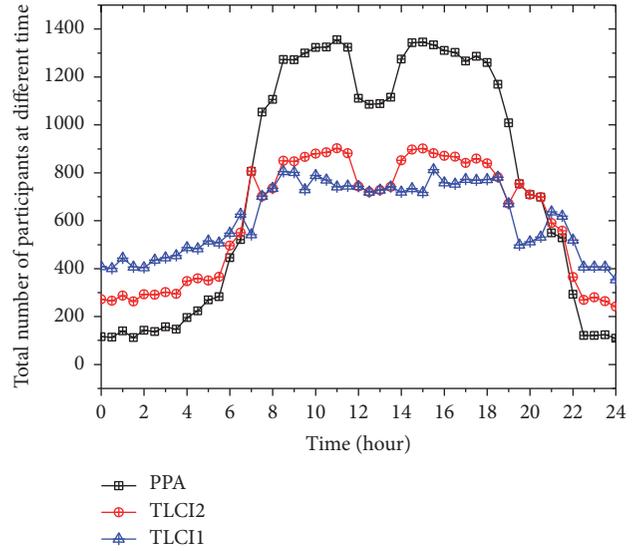


FIGURE 22: The number of participants at different time.

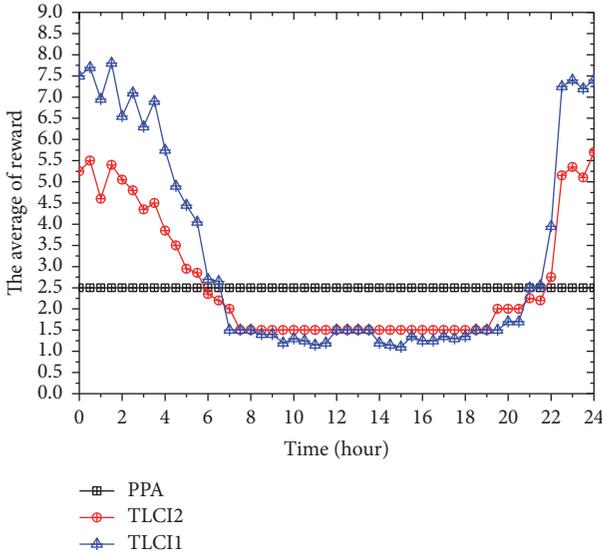


FIGURE 21: The average of reward for a user at different time.

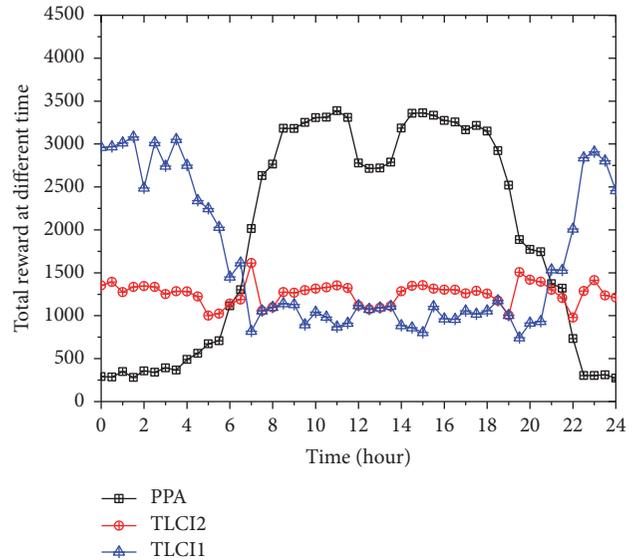


FIGURE 23: The total reward at different times.

21 and 23. It is obvious that compared to fixed reward in PPA, our two TLCI algorithms have a better adaptability, adjusting the user's reward according to the number of users. The overall trend is that the average of reward is declining with the more users as shown in Figure 21, which can be interpreted as the payment in exchange for participation. As a result, the number of participants and the total reward at different time are shown in Figures 22 and 23, in which total reward and total participants grow up in number when there are few users so that it can collect more sensor data. In addition, we can see the total cost of system is reduced by 34.20% and 14.89% in TLCI1 and TLCI2, respectively. Figure 24 shows the cumulative total reward over the time, which proves the effectiveness of our two algorithms and the TLCI2 could reduce the cost. In addition to the total reward,

we also study the sensed data over the time in Figure 25. Figure 26 shows that the redundancy of sensed data over time that the percentage of the difference between actual and expected sensed data to expected data. It is similar to the sensed data over time. And in Figure 27, we calculate the cumulative sensed data over time and we can see the sensed data in PPA exceeds the demand because the fixed reward in PPA has a bad adaptability compared with our two TLCI algorithms.

5.3.3. *The Comparison of Three Algorithms.* In this subsection, we compare the performance of three algorithms in two situations mentioned above. The results are shown as follows.

We use situation A and situation B to represent the situation of the random distribution and the concentrated

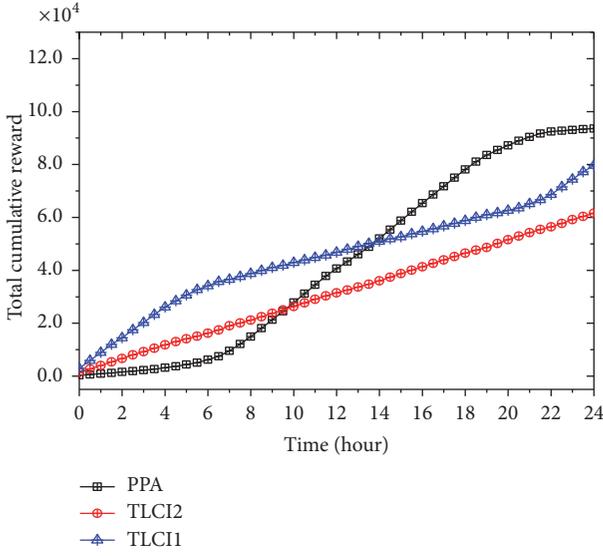


FIGURE 24: The total cumulative reward over time.

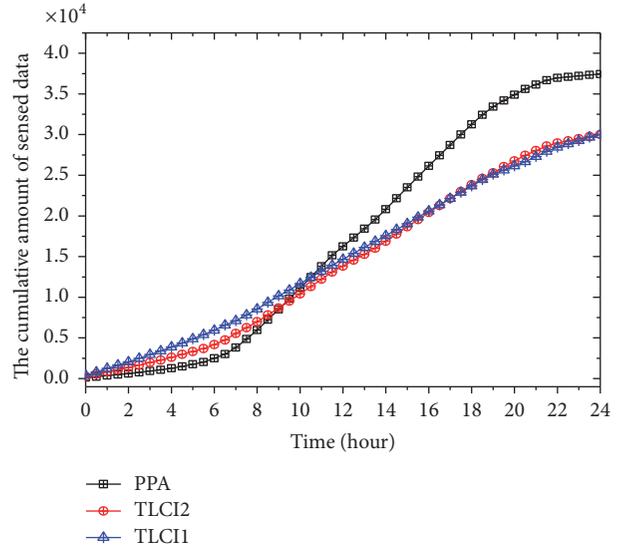


FIGURE 27: The cumulative amount of sensed data.

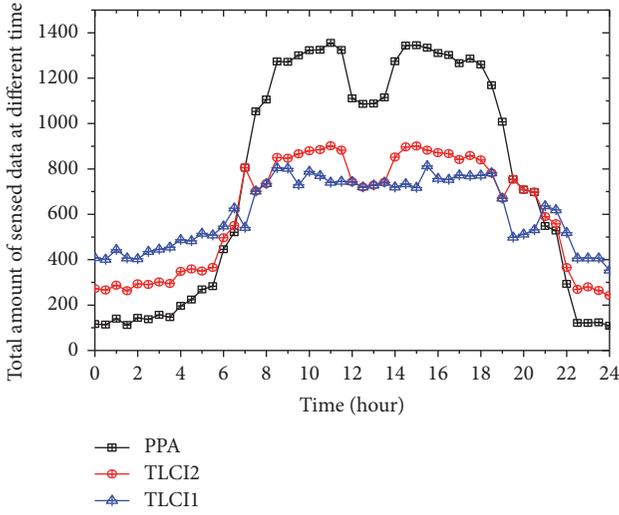


FIGURE 25: The total amount of sensed data.

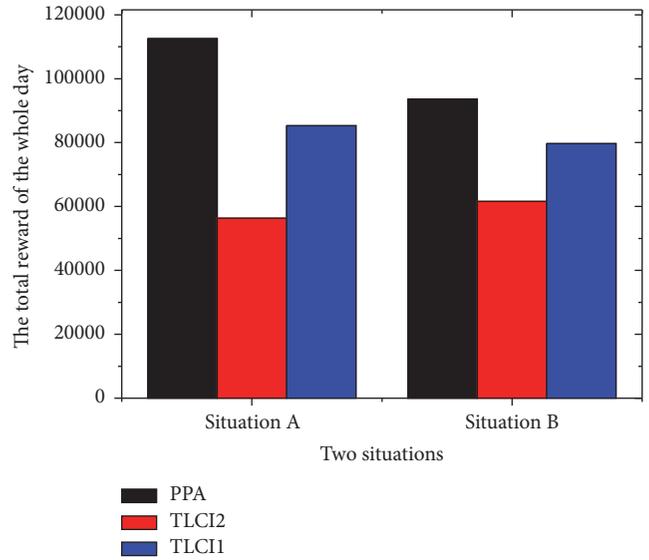


FIGURE 28: The comparison of total reward of the whole day in two situations.

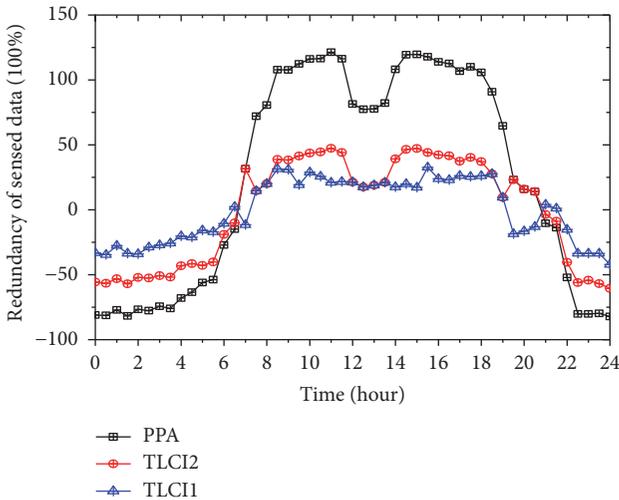


FIGURE 26: The redundancy of sensed data.

distribution, respectively. We first calculate the total reward and total sensed data in the whole day and make a comparison in Figures 28 and 29, respectively. According to Figure 28, we can see that our two TLCI algorithms can reduce the cost for rewarding the participants compared with fixed reward and TLCI2 has a better performance. On average of cost in two TLCI algorithms, it can reduce the cost of system by 37.07% in situation A and 24.54% in situation B, respectively. And Figure 30 shows that the TLCI algorithms can make the cost valuable rather than collecting redundant data.

Then, we make a comparison of total redundancy of sensed data of whole day in two situations and the result is shown in Figure 30. It shows the effectiveness of our two TLCI algorithms and the TLCI1 has a better performance

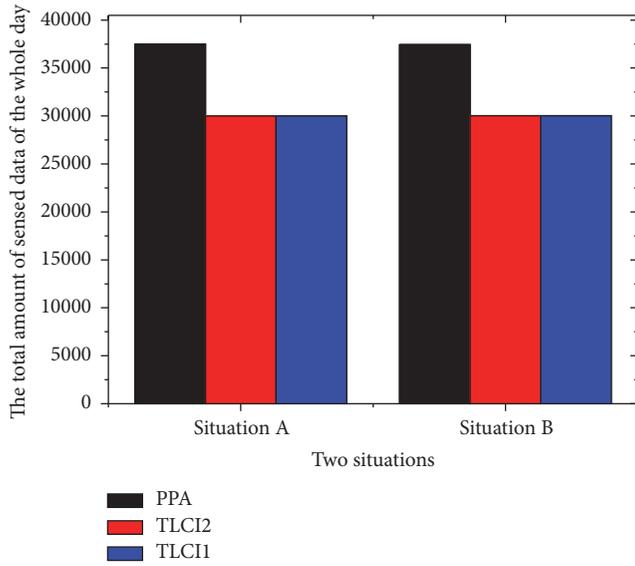


FIGURE 29: The comparison of total sensed data of the whole day in two situations.

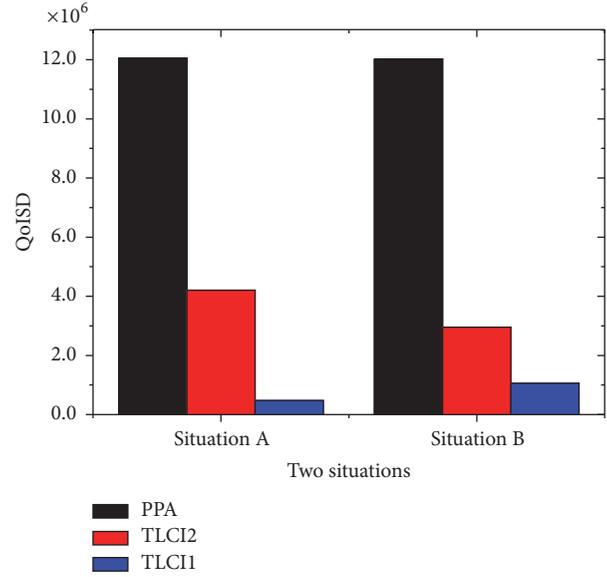


FIGURE 31: The comparison of QoISD of the whole day in two situations.

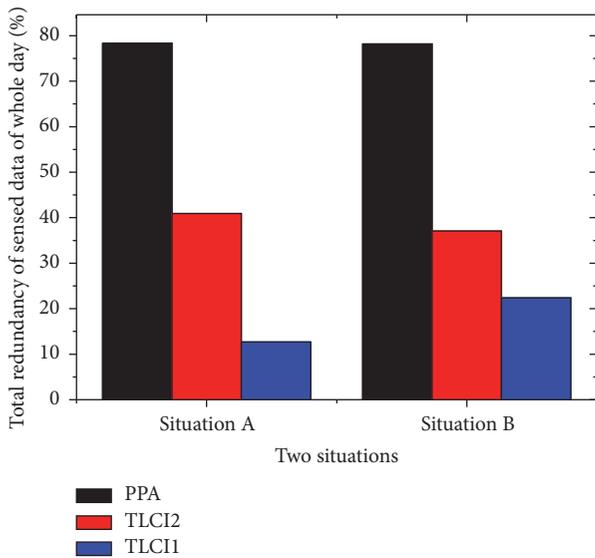


FIGURE 30: The comparison of total redundancy of sensed data of whole day in two situations.

compared with TLCI1. In addition, we make a comparison of performance on optimizing the QoISD in three algorithms, and the result is shown in Figure 31. On the average of QoISD in two situations, the QoISD can be optimized by 81.92%, and the total cost can be reduced by 31.38% with considering two TLCI algorithms.

**5.4. The Quality of Sensed Data.** In this subsection, we use the difference between the amount of actual and expected sensed data to evaluate the performance of our two algorithms (TLCI) and PPA. As above, we evaluate it in two situations, the active and inactive period of users, and the results are shown as follows. Figures 32, 33, and 34 show the difference

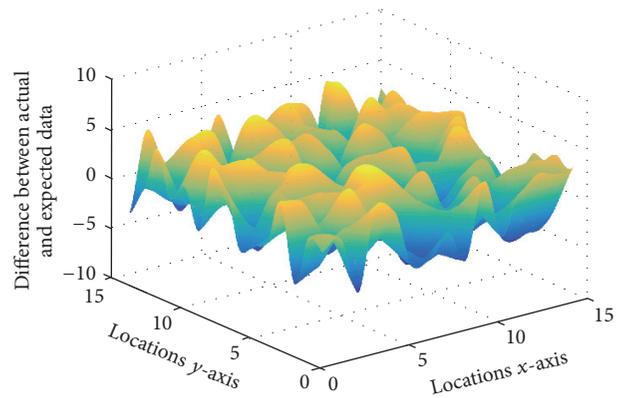


FIGURE 32: The difference between actual and expected sensed data in active period in TLCI1.

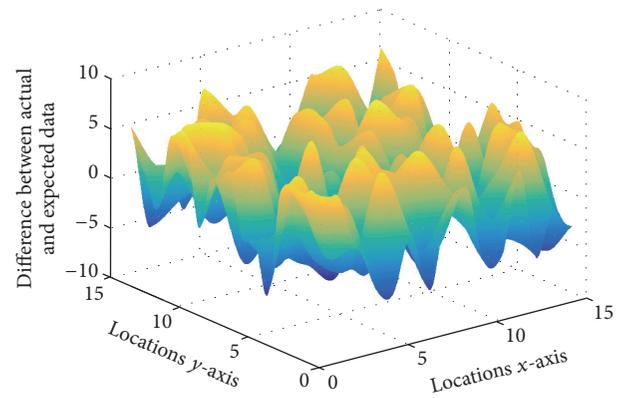


FIGURE 33: The difference between actual and expected sensed data in active period in TLCI2.

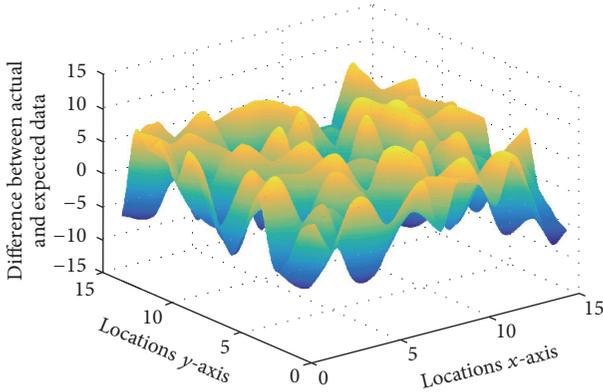


FIGURE 34: The difference between actual and expected sensed data in active period in PPA.

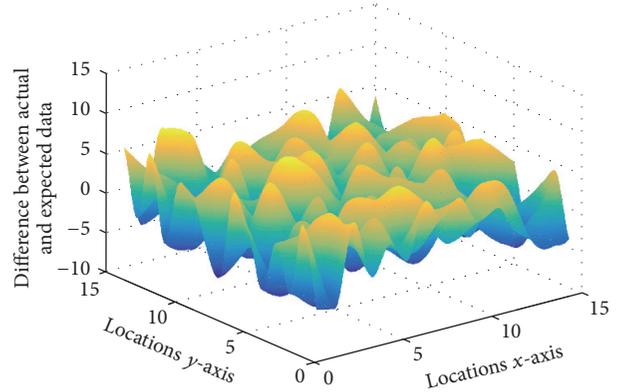


FIGURE 37: The difference between actual and expected sensed data in inactive period in PPA.

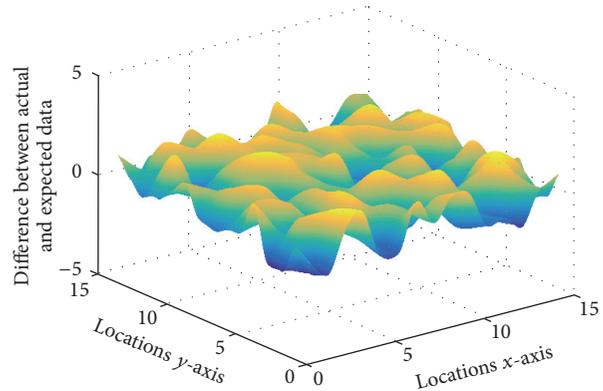


FIGURE 35: The difference between actual and expected sensed data in inactive period in TLCI1.

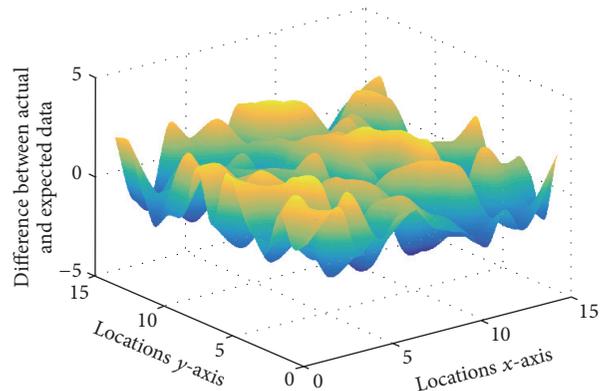
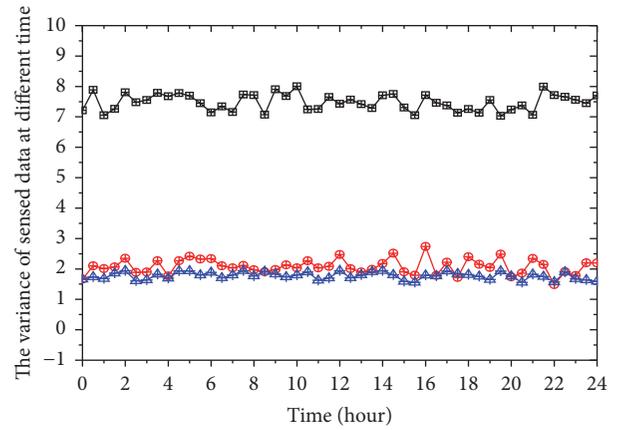


FIGURE 36: The difference between actual and expected sensed data in inactive period in TLCI2.

between actual and expected sensor data in three algorithms in active period. According to the figures, we can see that the difference between actual and expected sensed data in two TLCI algorithms is less than that in PPA algorithm and the performance of two TLCI algorithms is similar, where the difference is varying from  $-2.1$  to  $2.45$  in TLCI1, varying from  $-3.34$  to  $3.12$  in TLCI2, and varying from  $-9.12$  to  $9.27$  in PPA. And Figures 35, 36, and 37 show the same



—■— PPA  
—●— TLCI2  
—▲— TLCI1

FIGURE 38: The variance of sensed data over time.

conclusion as above. At last in this subsection, we calculate the variance of sensed data at different time in the situation of the distribution of users, which is shown in Figure 38. The variance of sensed data in two TLCI algorithms is less than that in PPA algorithms, where two TLCI algorithms have the similar performance on optimizing the quality of data. We can see the advantage of our two TLCI algorithms, making the sensed data evenly distributed at different locations.

**5.5. The Stability of TLCI Algorithms.** In this subsection, we mainly study the stability of TLCI algorithms. We first study the number of iterations by changing the value of  $\lambda$ . The result is shown in Figure 39, and we can see the number of iterations is declining with the increasing  $\lambda$  in TLCI2, which proves the better performance in speed in TLCI2 compared with TLCI1. Then we study the stability of TLCI algorithms. It is known that our algorithms will reach a steady state after limited number of iterations.

In Figures 40 and 41, we compare the total reward and the sensed data over the iteration in our two algorithms and it proves that the TLCI2 has the better performance in terms

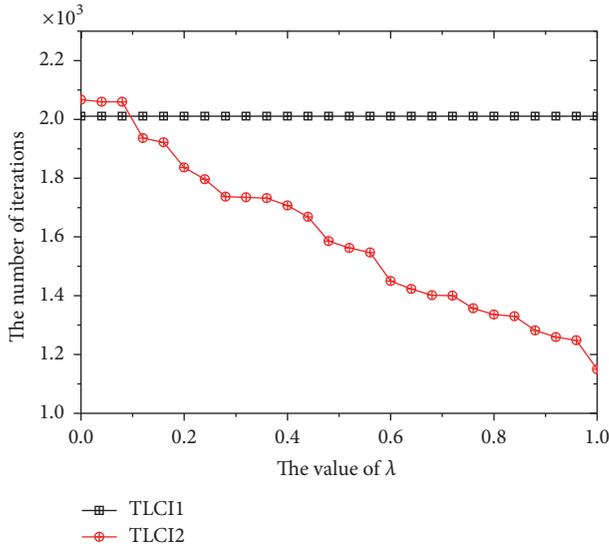


FIGURE 39: The number of iterations by changing the value of  $\lambda$ .

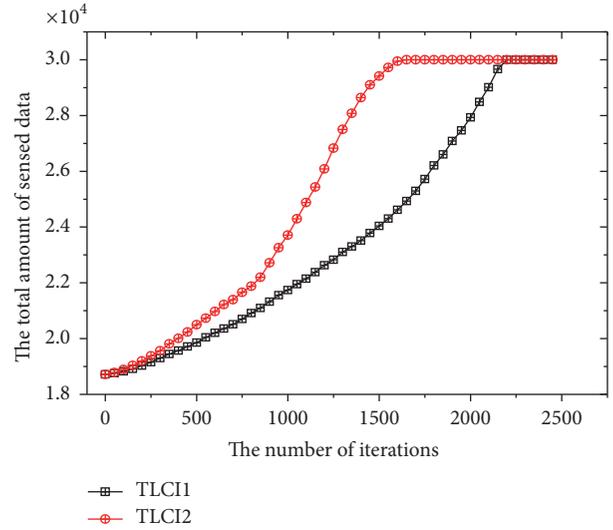


FIGURE 41: The total amount of sensed data over the iteration.

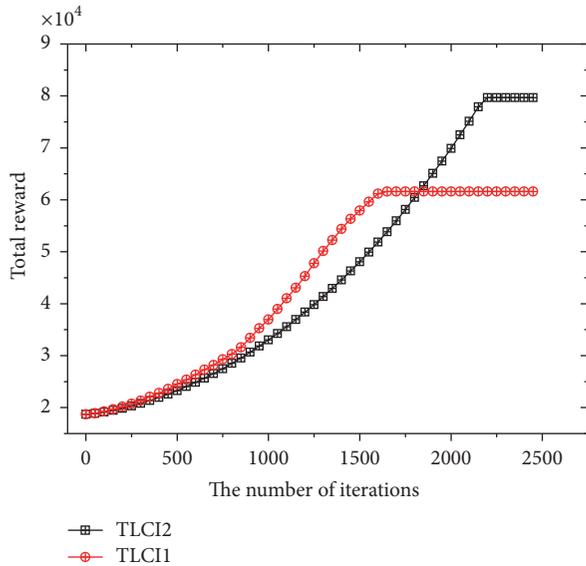


FIGURE 40: The total cumulative reward over the iteration.

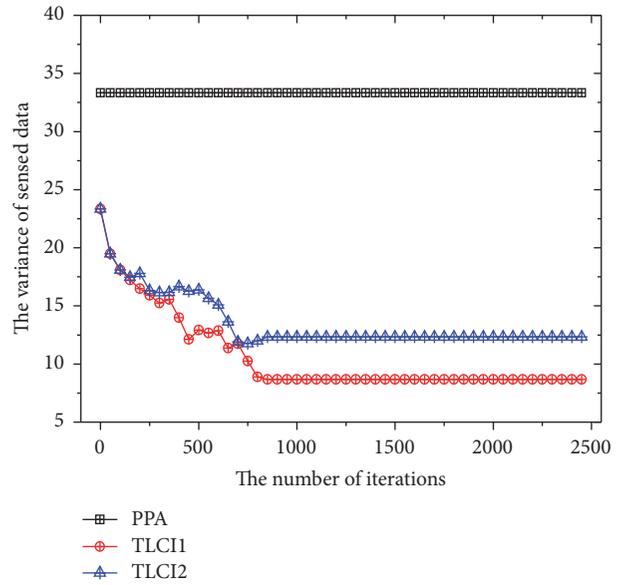


FIGURE 42: The variance of sensed data over the iteration.

of speed of algorithm and cost reduction, respectively. And we can see that the total cost of system after stabilization in TLCI2 is lower than that in TLCI1. In the end of this subsection, we calculate the variance of sensed data over the iteration as shown in Figure 42. The variance of sensed data at different subregions will stabilize after limited iterations and the stable variance of sensed data in TLCI2 is less than that in TLCI1, which means the same conclusion as the previous experiment.

## 6. Conclusion

This paper has focused on the problem of optimizing the quality of sensor data and collecting enough sensor data in a sensor data gathering system. In such a system, the cost

of each user depends on his circumstance, which has an impact on users whether to participate in the sensing task. The platform makes different reward for task participants in different region and time period so that it could make the distribution of sensed data on location and time best. Such a sensing system has been developed to realize such applications, such as NoiseTube [55].

In this paper, we have proposed two effective incentive mechanism algorithms for motivating smartphone users to participate in the smartphone sensing. Both of the algorithms are based on greedy strategies. One algorithm is to solve the problem of optimizing the QoISD. The other algorithm is improved on the basis of the first algorithm, which is to take the less cost on rewarding the participants. Extensive simulations have been performed and the results have confirmed

that the two algorithms can achieve our goal and the latter has a better speed than the former. QoISD can be optimized by 81.92%, and the total cost can be reduced by 31.38%.

## Parameters' Description

$\alpha_i, c_i^j$ :	The parameter in the cost
$u_n$ :	The smartphone users
$u_n$ :	The participants in sensing task
$C_i(\tau)$ :	The cost of user $i$ when sensing time is $\tau$
$r_i$ :	The reward of participant $i$
$\delta(r_i)$ :	The participation probability of a user participating in sensing task
$u_i^j$ :	The number of smartphone users in subperiod $T_j$ and subarea $L_i$
$u_i^j$ :	The number of task participants in subperiod $T_j$ and subarea $L_i$
$r_i^j$ :	The reward for every participant in subperiod $T_j$ and subarea $L_i$
$u_i^j$ :	The number of participants in subperiod $T_j$ and subarea $L_i$ based on probability estimate
$d_i^j$ :	The estimated amount of sensor data in subperiod $T_j$ and subarea $L_i$
$d_n^2$ :	The actual amount of sensor data in subperiod $T_j$ and subarea $L_i$
$\mathfrak{D}_i^j$ :	The actual demand of sensor data in subperiod $T_j$ and subarea $L_i$
$\mathfrak{Q}^j$ :	QoISD in subperiod $T_j$
$C^j$ :	The cost of system for rewarding participants in subperiod $T_j$
$D_{\text{total}}^j$ :	The total demand of sensor data in subperiod $T_j$
$\lambda$ :	The parameter in the target function
$\varepsilon_k$ :	The cost-efficiency of subarea $k$ .

## Conflicts of Interest

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

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (61772554, 61370229, 61572526, and 61572528), the National Basic Research Program of China (973 Program) (2014CB046305), the Science and Technology Projects of Guangdong Province, China (2016B010109008 and 2016B030305004), and the Science and Technology Projects of Guangzhou Municipality, China (201604010003 and 201604010054).

## References

- [1] K. Zhang, X. Liang, R. Lu, and X. Shen, "Sybil attacks and their defenses in the internet of things," *IEEE Internet of Things Journal*, vol. 1, no. 5, pp. 372–383, 2014.
- [2] Z. Sheng, H. Wang, C. Yin, X. Hu, S. Yang, and V. C. M. Leung, "Lightweight management of resource-constrained sensor devices in internet of things," *IEEE Internet of Things Journal*, vol. 2, no. 5, pp. 402–411, 2015.
- [3] T. Qiu, R. Qiao, and D. Wu, "EABS: an event-aware backpressure scheduling scheme for emergency internet of things," *IEEE Transactions on Mobile Computing*, 2017.
- [4] T. Qiu, Y. Zhang, D. Qiao, X. Zhang, M. L. Wymore, and A. K. Sangaiah, "A robust time synchronization scheme for industrial internet of things," *IEEE Transactions on Industrial Informatics*.
- [5] A. Liu, Z. Chen, and N. N. Xiong, "An adaptive virtual relaying set scheme for loss-and-delay sensitive WSNs," *Information Sciences*, vol. 424, pp. 118–136, 2018.
- [6] Z. Ning, F. Xia, N. Ullah, X. Kong, and X. Hu, "Vehicular social networks: enabling smart mobility," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 16–55, 2017.
- [7] X. Hu, V. C. Leung, Y. Kwok et al., "SAFeDJ: a crowd-cloud co-design approach to situation-aware music delivery for drivers," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 12, no. 1, 2015.
- [8] H. Li, D. Liu, Y. Dai, T. H. Luan, and S. Yu, "Personalized search over encrypted data with efficient and secure updates in mobile clouds," *IEEE Transactions on Emerging Topics in Computing*, vol. PP, no. 99, 2016.
- [9] T. Wang, J. Zeng, M. Z. Bhuiyan et al., "Trajectory privacy preservation based on a fog structure for cloud location services," *IEEE Access*, vol. 5, pp. 7692–7701, 2017.
- [10] J. Gao, P. Liu, X. Kang, L. Zhang, and J. Wang, "PRS: parallel relaxation simulation for massive graphs," *The Computer Journal*, vol. 59, no. 6, pp. 848–860, 2016.
- [11] K. Zhang, X. Liang, R. Lu, and X. Shen, "PIF: a personalized fine-grained spam filtering scheme with privacy preservation in mobile social networks," *IEEE Transactions on Computational Social Systems*, vol. 2, no. 3, pp. 41–52, 2015.
- [12] J. Xu, X. Liu, M. Ma, A. Liu, T. Wang, and C. Huang, "Intelligent aggregation based on content routing scheme for cloud computing," *Symmetry*, vol. 9, no. 10, 2017.
- [13] Z. Su, Q. Xu, M. Fei, and M. Dong, "Game theoretic resource allocation in media cloud with mobile social users," *IEEE Transactions on Multimedia*, vol. 18, no. 8, pp. 1650–1660, 2016.
- [14] K. Zhang, K. Yang, X. Liang, Z. Su, X. Shen, and H. H. Luo, "Security and privacy for mobile healthcare networks: from a quality of protection perspective," *IEEE Wireless Communications Magazine*, vol. 22, no. 4, pp. 104–112, 2015.
- [15] D. Zeng, L. Gu, L. Lian, S. Guo, H. Yao, and J. Hu, "On cost-efficient sensor placement for contaminant detection in water distribution systems," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2177–2185, 2016.
- [16] C. Hu, M. Li, D. Zeng, and S. Guo, "A survey on sensor placement for contamination detection in water distribution systems," *Wireless Networks*, 2016.
- [17] J. Gui and K. Zhou, "Flexible adjustments between energy and capacity for topology control in heterogeneous wireless multi-hop networks," *Journal of Network and Systems Management*, vol. 24, no. 4, pp. 789–812, 2016.
- [18] Z. Ning, X. Hu, Z. Chen et al., "A cooperative quality-aware service access system for social internet of vehicles," *IEEE Internet of Things Journal*.
- [19] M. Zhou, M. Zhao, A. Liu, M. Ma, T. Wang, and C. Huang, "Fast and efficient data forwarding scheme for tracking mobile targets in sensor networks," *Symmetry*, vol. 9, no. 11, p. 269, 2017.

- [20] 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.
- [21] M. Marjanović, L. Skorin-Kapov, K. Pripuzić, A. Antonić, and I. P. Žarko, "Energy-aware and quality-driven sensor management for green mobile crowd sensing," *Journal of Network and Computer Applications*, vol. 59, pp. 95–108, 2016.
- [22] A. Antonić, M. Marjanović, K. Pripuzić, and I. P. Žarko, "A mobile crowd sensing ecosystem enabled by CUPUS: cloud-based publish/subscribe middleware for the internet of things," *Future Generation Computer Systems*, vol. 56, pp. 607–622, 2016.
- [23] Y. Gao, Y. Chen, and K. J. R. Liu, "On cost-effective incentive mechanisms in microtask crowdsourcing," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 7, no. 1, pp. 3–15, 2015.
- [24] T. Wang, J. Zeng, Y. Lai et al., "Data collection from WSNs to the cloud based on mobile Fog elements," *Future Generation Computer Systems*, 2017.
- [25] T. Wang, Y. Li, G. Wang, J. Cao, M. Z. Bhuiyan, and W. Jia, "Sustainable and efficient data collection from WSNs to cloud," *IEEE Transactions on Sustainable Computing*, 2017.
- [26] H. Gao, C. H. Liu, W. Wang et al., "A survey of incentive mechanisms for participatory sensing," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 918–943, 2015.
- [27] X. Hu, T. H. S. Chu, V. C. M. Leung, E. C.-H. Ngai, P. Kruchten, and H. C. B. Chan, "A survey on mobile social networks: applications, platforms, system architectures, and future research directions," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1557–1581, 2014.
- [28] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, "A survey of mobile phone sensing," *IEEE Communications Magazine*, vol. 48, no. 9, pp. 140–150, 2010.
- [29] Y. Wang, Z. Cai, G. Yin, Y. Gao, X. Tong, and G. Wu, "An incentive mechanism with privacy protection in mobile crowdsourcing systems," *Computer Networks*, vol. 102, pp. 157–171, June 2016.
- [30] C.-K. Tham and T. Luo, "Quality of contributed service and market equilibrium for participatory sensing," *IEEE Transactions on Mobile Computing*, vol. 14, no. 4, pp. 829–842, 2015.
- [31] S. Reddy, D. Estrin, and M. Srivastava, "Recruitment framework for participatory sensing data collections," in *Pervasive Computing*, vol. 6030 of *Lecture Notes in Computer Science*, pp. 138–155, Springer, Berlin, Germany, 2010.
- [32] Z. Song, C. H. Liu, J. Wu, J. Ma, and W. Wang, "QoI-aware multitask-oriented dynamic participant selection with budget constraints," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4618–4632, 2014.
- [33] X. Liu, "A novel transmission range adjustment strategy for energy hole avoiding in wireless sensor networks," *Journal of Network and Computer Applications*, vol. 67, pp. 43–52, 2016.
- [34] H. Xin and X. Liu, "Energy-balanced transmission with accurate distances for strip-based wireless sensor networks," *IEEE Access*, vol. 5, pp. 16193–16204, 2017.
- [35] J. Gui, L. Hui, and N. Xiong, "A game-based localized multi-objective topology control scheme in heterogeneous wireless networks," *IEEE Access*, vol. 5, pp. 2396–2416, 2017.
- [36] J. Gui and J. Deng, "A topology control approach reducing construction cost for lossy wireless sensor networks," *Wireless Personal Communications*, vol. 3, pp. 2173–2202, 2017.
- [37] D. Zeng, P. Li, S. Guo, T. Miyazaki, J. Hu, and Y. Xiang, "Energy Minimization in Multi-Task Software-Defined Sensor Networks," *IEEE Transactions on Computers*, vol. 64, no. 11, pp. 3128–3139, 2015.
- [38] T. Qiu, A. Zhao, F. Xia, W. Si, and D. O. Wu, "ROSE: robustness strategy for scale-free wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2944–2959, 2017.
- [39] X. Hu, T. H. S. Chu, H. C. B. Chan, and V. C. M. Leung, "Vita: a crowdsensing-oriented mobile cyber-physical system," *IEEE Transactions on Emerging Topics in Computing*, vol. 1, no. 1, pp. 148–165, 2013.
- [40] G. Yang, S. He, and Z. Shi, "Leveraging crowdsourcing for efficient malicious users detection in large-scale social networks," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 330–339, 2017.
- [41] S. He, D. Shin, J. Zhang, J. Chen, and P. Lin, "An exchange market approach to mobile crowdsensing: pricing, task allocation, and walrasian equilibrium," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 4, pp. 921–934, 2017.
- [42] X. Duan, C. Zhao, S. He, P. Cheng, and J. Zhang, "Distributed algorithms to compute walrasian equilibrium in mobile crowdsensing," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4048–4057, 2017.
- [43] A. Kittur, E. H. Chi, and B. Suh, "Crowdsourcing user studies with Mechanical Turk," in *Proceedings of the 26th Annual CHI Conference on Human Factors in Computing Systems, (CHI '08)*, pp. 453–456, Florence, Italy, April 2008.
- [44] K. Lu, S. Wang, L. Xie, Z. Wang, and M. Li, "A dynamic reward-based incentive mechanism: Reducing the cost of P2P systems," *Knowledge-Based Systems*, vol. 112, pp. 105–113, 2017.
- [45] Z. Chen, M. Ma, X. Liu, A. Liu, and M. Zhao, "Reliability improved cooperative communication over wireless sensor networks," *Symmetry*, vol. 9, no. 10, p. 209, 2017.
- [46] J.-S. Lee and B. Hoh, "Sell your experiences: a market mechanism based incentive for participatory sensing," in *Proceedings of the 8th IEEE International Conference on Pervasive Computing and Communications, (PerCom '10)*, pp. 60–68, April 2010.
- [47] L. G. Jaimes, I. Vergara-Laurens, and M. A. Labrador, "A location-based incentive mechanism for participatory sensing systems with budget constraints," in *Proceedings of the 10th IEEE International Conference on Pervasive Computing and Communications, (PerCom '12)*, pp. 103–108, March 2012.
- [48] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing," in *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking (MobiCom '12)*, pp. 173–184, August 2012.
- [49] C. H. Liu, B. Zhang, X. Su, J. Ma, W. Wang, and K. K. Leung, "Energy-aware participant selection for smartphone-enabled mobile crowd sensing," *IEEE Systems Journal*, vol. PP, no. 99, 2015.
- [50] C. H. Liu, J. Fan, P. Hui, J. Wu, and K. K. Leung, "Toward QoI and energy efficiency in participatory crowdsourcing," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 10, pp. 4684–4700, 2014.
- [51] C. H. Liu, J. Fan, P. Hui, J. Wu, and K. K. Leung, "Toward QoI and energy efficiency in participatory crowdsourcing," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 10, pp. 4684–4700, 2015.
- [52] C. H. Liu, B. Zhang, X. Su, J. Ma, W. Wang, and K. K. Leung, "Energy-aware participant selection for smartphone-enabled mobile crowd sensing," *IEEE Systems Journal*, vol. 11, no. 3, pp. 1435–1446, 2017.

- [53] Y. Zhang and M. Van Der Schaar, "Robust reputation protocol design for online communities: a stochastic stability analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 5, pp. 907–920, 2013.
- [54] W. Wang, H. Gao, C. H. Liu, and K. K. Leung, "Credible and energy-aware participant selection with limited task budget for mobile crowd sensing," *Ad Hoc Networks*, vol. 43, pp. 56–70, 2016.
- [55] N. Maisonneuve, M. Stevens, M. E. Niessen, and L. Steels, "NoiseTube: measuring and mapping noise pollution with mobile phones," in *Information Technologies in Environmental Engineering*, Environmental Science and Engineering, pp. 215–228, Springer, Berlin, Germany, 2009.

## Research Article

# Network Traffic Prediction Based on Deep Belief Network and Spatiotemporal Compressive Sensing in Wireless Mesh Backbone Networks

Laisen Nie,<sup>1</sup> Xiaojie Wang ,<sup>2</sup> Liangtian Wan ,<sup>2</sup> Shui Yu,<sup>3</sup> Houbing Song,<sup>4</sup> and Dingde Jiang<sup>5</sup>

<sup>1</sup>School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710072, China

<sup>2</sup>School of Software, Dalian University of Technology, Dalian 116620, China

<sup>3</sup>School of Information Technology, Deakin University, Burwood, VIC 3125, Australia

<sup>4</sup>Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL 32114, USA

<sup>5</sup>School of Astronautics & Aeronautics, University of Electronic Science and Technology of China, Chengdu 611731, China

Correspondence should be addressed to Xiaojie Wang; wangxj1988@mail.dlut.edu.cn

Received 9 September 2017; Accepted 7 December 2017; Published 4 January 2018

Academic Editor: Kuan Zhang

Copyright © 2018 Laisen Nie 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.

Wireless mesh network is prevalent for providing a decentralized access for users and other intelligent devices. Meanwhile, it can be employed as the infrastructure of the last few miles connectivity for various network applications, for example, Internet of Things (IoT) and mobile networks. For a wireless mesh backbone network, it has obtained extensive attention because of its large capacity and low cost. Network traffic prediction is important for network planning and routing configurations that are implemented to improve the quality of service for users. This paper proposes a network traffic prediction method based on a deep learning architecture and the Spatiotemporal Compressive Sensing method. The proposed method first adopts discrete wavelet transform to extract the low-pass component of network traffic that describes the long-range dependence of itself. Then, a prediction model is built by learning a deep architecture based on the deep belief network from the extracted low-pass component. Otherwise, for the remaining high-pass component that expresses the gusty and irregular fluctuations of network traffic, the Spatiotemporal Compressive Sensing method is adopted to predict it. Based on the predictors of two components, we can obtain a predictor of network traffic. From the simulation, the proposed prediction method outperforms three existing methods.

## 1. Introduction

The Wireless Mesh Network (WMN) provides ubiquitous and last few miles connectivity for future wireless service, for example, IoT, 5G mobile network, and cognitive radio. Besides, it is also a promising solution to IoT crowdsensing applications by connecting a large group of individuals with capacities of computing and sensing. Compared with other wireless architectures (e.g., ad hoc networks), the WMN has high capacity, robustness, and low-cost deployment [1]. Hence, it is much more popular as an emerging access paradigm in practice. With the rapid development of mobile communications, mobile cloud systems, and IoT, the applications provided by wireless networks become multitudinous.

Besides, the big data has become a crucial role in both industry and daily life. The quantity and substantial growth of WMN (both in scale and in service) bring a series of new challenges for the capacity; for example, the network congestion may appear in a wireless mesh backbone network induced by huge traffic demands.

Mesh routers constitute the principal infrastructure of the WMN known as the wireless mesh backbone network. The self-organized manner of the WMN architecture reinforces the resilience of the network to failures, but it arises some limitations, typically, resource allocation problem, and so on. Above all, imperative network management operation is useful to provide a cost-effective solution for improving

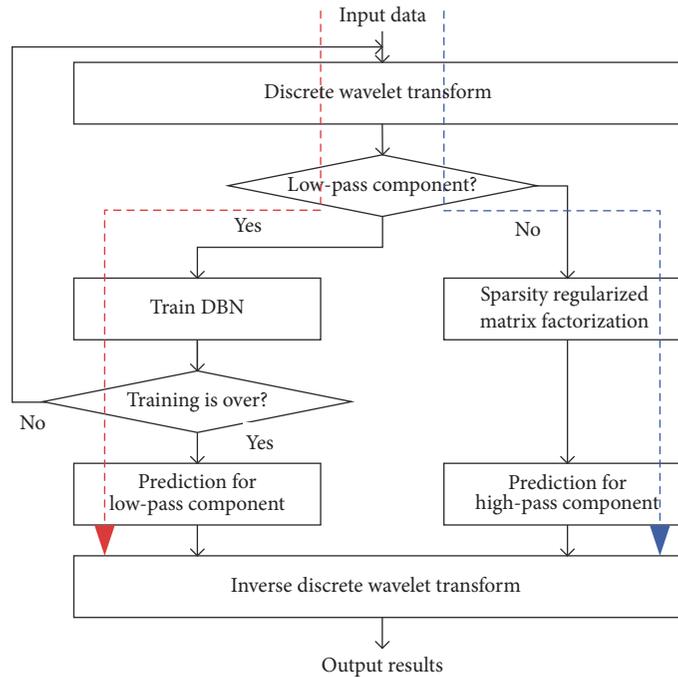


FIGURE 1: Prediction method.

the performance of a WMN. These network management operations are implemented in terms of the relative network traffic information. For instance, in order to improve the quality of service for users, predictive network planning is necessary for ISPs. This planning is carried out according to the future traces of network traffic flows between all possible origin-destination (OD) node pairs [2, 3].

A great number of methods have been proposed to deal with the network traffic prediction problem in traditional IP backbone networks [4–8]. Statistical methods have been widely adopted in this field. Originally, some simple models, such as Autoregressive (AR) and Autoregressive Integrated Moving Average (ARIMA), are used to pursue the short-range dependence (SRD) of network traffic [4]. However, current network traffic exhibits a long-range dependence (LRD) characteristic and multifractal features in terms of the behaviors of terminals [4]. In this case, the Fractional Autoregressive Integrated Moving Average (FARIMA) model and the Multifractal Wavelet model (MWM) are involved in this field to deal with the network traffic prediction problem [5]. With the variety of network services and applications, the characteristics of network traffic are much more complex. For instance, it exhibits some nonlinear features [4, 6]. Hence, some methods refer to the GARCH model to model network traffic for prediction. Besides, a lot of methods based on hybrid models have been proposed to predict network traffic [7]. However, these methods are not suitable for dealing with the problems of network traffic prediction in a wireless mesh backbone network [2]. Generally, the users join in a Wireless Mesh Network randomly. Additionally, they often have the complicated individual association, which is significantly discrepant compared with the users of a traditional IP backbone network. Although the users' applications of two

networks are probably coincident, the dominant applications usually are distinguishing.

Motivated by this issue, we propose a network traffic prediction method based on the deep belief network (DBN) and the Spatiotemporal Compressive Sensing (STCS) method, named Deep Belief Network and Spatiotemporal Compressive Sensing (DBNSTCS) models. To the best of our knowledge, this is the first paper that focuses on the problem of traffic prediction for wireless mesh backbone networks and takes into account the spatiotemporal characteristic for prediction. We take account of the long-range dependence and irregular fluctuation behaviors of network traffic independently; see Figure 1 [9]. By the discrete wavelet transform (DWT), the network traffic is divided into two components tagged by scaling and discrete wavelet transform coefficients. Namely, the DWT just likes a filter that decomposes the network traffic into a low-pass component and a high-pass component. The low-pass component expresses the long-range dependence of network traffic, and the high-pass one declares the gusty and irregular fluctuations. The LRD means that the network traffic at any time depends upon multiple previous traffic data. The LRD of network traffic derives from a series of interacted factors (e.g., the behaviors of users). To describe these multifarious relationships, the former is predicted by a deep architecture based on DBN. The proposed architecture can deeply learn the LRD of network traffic. For the short-range and irregular fluctuations, the STCS method known as an excellent interpolation algorithm is employed to predict them.

The contributions of the paper are proposed as follows:

- (i) We use the DWT to extract the low-pass and high-pass components of network traffic. The DWT of a

time series can be viewed as making this time series pass a low-pass filter and a high-pass filter, respectively. Hence, we can obtain the low-pass and high-pass components of network traffic. They show the low-pass approximation and the details of network traffic, respectively. In our method, we predict two types of coefficients independently.

- (ii) We propose a deep architecture based on DBN to capture the low-pass component of network traffic. Under this architecture, by learning the built deep architecture in terms of a training set via known network traffic, the deep architecture can describe the LRD characteristic of traffic flows and carry out a prediction for network traffic.
- (iii) We adopt the Spatiotemporal Compressive Sensing to fit the gusty and irregular fluctuations of network traffic. We first assume that the high-pass component obeys a spatiotemporal dependence. Then, we achieve a predictor of network traffic by the Sparsity Regularized Matrix Factorization (SRMF) method.

The remaining parts of this paper are organized as follows. Section 2 reviews the related work about network traffic prediction problem. In Section 3, we introduce some definitions about network traffic, the DWT techniques, the DBN theory, and the Spatiotemporal Compressive Sensing method, respectively. We propose our prediction method in Section 4. Then, we verify the performance of our method in Section 5. Section 6 concludes our work of this paper.

## 2. Related Work

Lots of researchers have investigated network traffic prediction that is instructive for congestion control, predictive network planning, and intelligent routing [10–14]. The existing network traffic prediction techniques consist of four categories: linear time series methods, nonlinear time series methods, hybrid model methods, and decomposition model methods.

The linear time series methods (e.g., AR, MA, and ARMA) are frequently used to model end-to-end traffic flows for prediction. According to novel research findings, the traffic flows exhibit observably nonlinear features under complex network users behaviors and various applications. Typically, the GARCH model in [10] is used to model the burst characteristics. Besides, neural network is also a valid method to track the traffic flows with nonlinear characteristic [12].

With the rapidly development of network services, the ISP network has been a heterogeneous and complex network. Traffic flows show manifold statistic characteristics such as LRD, SRD, heavy-tailed distribution, and multifractal feature. Therefore, researchers adopt some hybrid models to model the traffic flows with complicated distributions. The methods based on hybrid model take advantage of two or more models to capture the traces of traffic flows. In [4], the authors combine the ARIMA model with the GARCH model to fit several characteristics of traffic flows (i.e., LRD and SRD

characteristics). Meanwhile, the proposed hybrid model can also model the self-similarity and multifractal features of traffic flows. The autocorrelation function and the partial autocorrelation function are employed by the authors to determine the parameters in the hybrid model. The fourth method as mentioned in the above part is the decomposition model methods in which the traffic flows are divided into several components. Based on this decomposition, the gained components are, respectively, modeled and predicted. These methods can be viewed as an evolution of the hybrid model methods. In [13], the authors jointly use the Stationary Wavelet Transform (SWT), the Quantum Genetic Algorithm (QGA), and the Backpropagation Neural Network (BPNN) to implement traffic prediction of wireless network traffic. They first decompose the traffic flows by the SWT such that the traffic flows are made up of several stationary components. After that, all these components are predicted by a trained BPNN using the QGA. The authors in [14] decompose the traffic of a large scale cellular network into regular and random components by a classic time series decomposition method.

## 3. Background

*3.1. Traffic Matrix.* A traffic matrix is an expression form of network traffic. After collecting network traffic information, the operators implement appropriate network management functions in terms of this network traffic information. During this period, the network traffic information is expressed as the traffic matrix. If we denote an OD flow by  $x_{p,q}(t)$  which describes the mean of the volume of traffic flow from the origin node  $p$  to the destination node  $q$  in the  $t$ th time slot, then the traffic matrix is defined as

$$X = \begin{bmatrix} x_{1,1}(1) & x_{1,1}(2) & \cdots & x_{1,1}(T) \\ x_{1,2}(1) & x_{1,2}(2) & \cdots & x_{1,2}(T) \\ \vdots & \vdots & \cdots & \vdots \\ x_{N,N}(1) & x_{N,N}(2) & \cdots & x_{N,N}(T) \end{bmatrix}, \quad (1)$$

where  $p, q \in \{1, 2, \dots, N\}$ , and  $t \in \{1, 2, \dots, T\}$ . This traffic matrix reports the network traffic data with  $T$  time slots. Generally, the length of time slot is 5 or 15 minutes.

*3.2. Discrete Wavelet Transform.* For a time series  $f(t)$ , it can be expressed by

$$f(t) = \sum_{n=-\infty}^{+\infty} c_{L,n} 2^{-L/2} \phi\left(\frac{t}{2^L} - n\right) + \sum_{l=1}^L \sum_{n=-\infty}^{+\infty} d_{l,n} 2^{-l/2} \psi\left(\frac{t}{2^l} - n\right), \quad (2)$$

where  $c_{L,n}$  and  $d_{l,n}$  are scaling and discrete wavelet transform coefficients.  $2^{-L/2} \phi(t/2^L - n)$  is the scaling function at scale  $L$ .  $2^{-l/2} \psi(t/2^l - n)$  is so-called the wavelet function. The scaling coefficients express a coarse approximation of  $f(t)$ . The

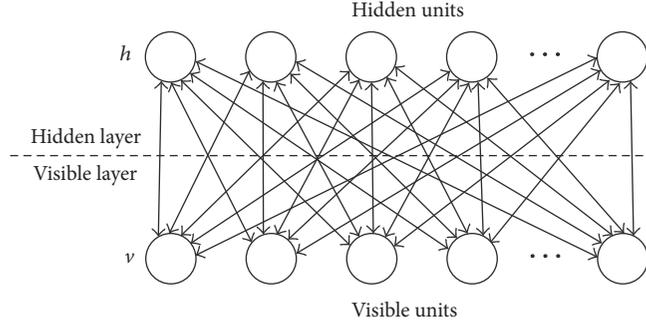


FIGURE 2: Restricted Boltzmann Machine.

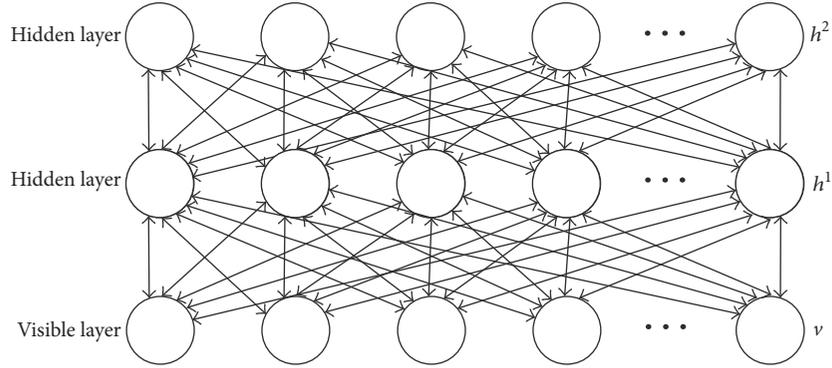


FIGURE 3: Deep Belief Network with two Restricted Boltzmann Machines.

discrete wavelet transform coefficients represent the details of  $f(t)$ . Hence, (2) can be viewed as making the time series pass a filter and then obtaining a representation of  $f(t)$  by the combination of low-pass and high-pass approximations.

**3.3. Deep Belief Network.** The DBN is a common deep learning primitive. It is a combination of a number of Restricted Boltzmann Machines (RBMs) [15–17]. A RBM that is two-layer undirected graphical model consists of the visible and hidden layers denoted by  $v$  and  $h$  (shown by Figure 2) [9, 15]. Each unit in a layer is connected with all units of the other layer by undirected edges. The units in the same layer are disconnected with each other. Figure 3 shows an example of DBN architecture with two RBMs [9]. The DBN is a stack of many RBMs. The values of all units are stochastic variables [16]. Generally, they obey a Bernoulli distribution or a Gaussian distribution. When the visible and hidden units are Gaussian and Bernoulli, we have the following conditional distribution:

$$\begin{aligned} P(v_i | h) &= N\left(b_i + \sum_{j=1}^J w_{i,j} h_j, 1\right), \\ P(h_j = 1 | v) &= \text{sigm}\left(a_j + \sum_{i=1}^I w_{i,j} v_i\right), \end{aligned} \quad (3)$$

where  $N(b_i + \sum_{j=1}^J w_{i,j} h_j, 1)$  denotes the Gaussian distribution whose mean and variance are  $b_i + \sum_{j=1}^J w_{i,j} h_j$  and 1.  $\text{sigm}(z) =$

$\exp(z)/(1 + \exp(z))$  is the sigmoid function.  $I$  and  $J$  are the numbers of visible and hidden units, respectively [18].  $b_i$  and  $a_j$  are the biases of visible and hidden units.  $w_{i,j}$  expresses the symmetric interaction term between the visible unit  $v_i$  and the hidden unit  $h_j$ . For a RBM, the joint probability distribution function over visible and hidden units can be denoted by

$$P(v, h) = \frac{\exp(-E(v, h))}{\sum_{v, h} \exp(-E(v, h))}, \quad (4)$$

where  $E(v, h)$  is termed as the energy function defined as

$$E(v, h) = -\frac{1}{2} \sum_{i=1}^I (b_i - v_i)^2 - \sum_{j=1}^J a_j h_j - \sum_{i=1}^I \sum_{j=1}^J w_{i,j} v_i h_j. \quad (5)$$

To train the DBN, the idea is to employ a layer-wise greedy strategy. Besides, the parameters are updated by minimizing the log probability  $\log P(v)$ .

**3.4. Spatiotemporal Compressive Sensing and Sparsity Regularized Matrix Factorization.** Compressive sensing is a novel sampling technique for signal processing in recent years, which makes good use of the structure or redundancy of real-world signals. It takes advantage of an adaptive sampling scheme to sense these structural signals. In detail, the structure of these signals means that it can be denoted by a vector that just has several nonzero elements (i.e., the vector is sparse). In the adaptive sampling scheme, a random

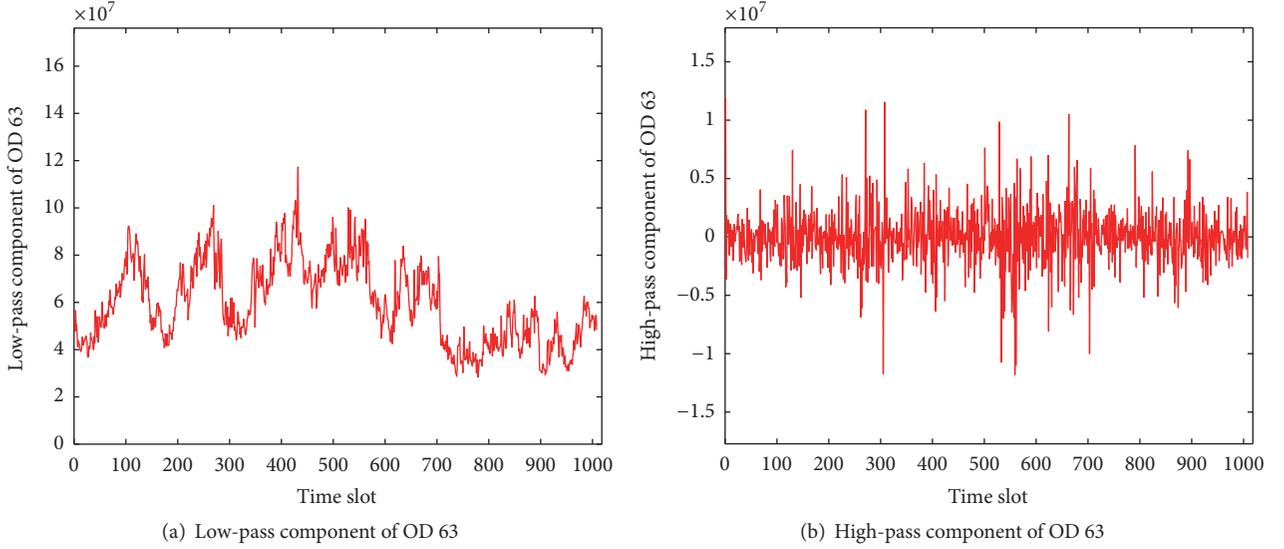


FIGURE 4: Decomposition of OD 63.

matrix called the measurement matrix is used to concurrently implement compressing and coding. During the decoding phase, the compressive sensing reconstruction algorithm is a splendid approach to deal with the inverse problem with ill-posed feature.

As a derivative of compressive sensing, the spatiotemporal compressive sensing technique is always used as an interpolation algorithm to recover the missing elements of a data set. The spatiotemporal feature means that the values of neighboring elements in the data set are properly similar. In terms of this feature, the SRMF method is proposed in [19], where the missing elements can be recovered precisely though the data loss probability is tremendous. Besides, the SRMF method is also an accurate tool for prediction. Under the prediction process, the elements that need to be predicted are viewed as continuous missing elements.

## 4. Our Methodology

**4.1. Decomposition of Network Traffic.** We assume that the known network traffic is denoted by  $X$  whose each OD flow is denoted by a time series  $x_{p,q}(t)$ , where  $t = 1, 2, \dots, T$ . According to (1), it can be denoted by

$$x_{p,q}(t) = \sum_{n=-\infty}^{+\infty} c_{L,n,p,q} 2^{-L/2} \phi\left(\frac{t}{2^L} - n\right) + \sum_{l=1}^L \sum_{n=-\infty}^{+\infty} d_{l,n,p,q} 2^{-l/2} \psi\left(\frac{t}{2^l} - n\right). \quad (6)$$

If we set the scale to be 1, then we have

$$x_{p,q}(t) = \sum_{n=-\infty}^{+\infty} c_{1,n,p,q} 2^{-1/2} \phi\left(\frac{t}{2} - n\right) + \sum_{n=-\infty}^{+\infty} d_{1,n,p,q} 2^{-1/2} \psi\left(\frac{t}{2} - n\right). \quad (7)$$

The above equation divides the network traffic into two components. One is the low-pass approximation (shown by the scaling coefficients) that exhibits the LRD of the network traffic  $x_{p,q}(t)$ , and the other is the high-pass approximation (described by the discrete wavelet transform coefficients) that expresses the gusty and irregular fluctuation behaviors of the network traffic  $x_{p,q}(t)$ . For a traffic matrix that describes the volume of traffic between all OD node pairs, obviously, its low-pass and high-pass approximation components can be denoted by two matrices, respectively.

Figures 4 and 5 give two examples of the traffic flow decomposition by the DWT. We select two OD flows from the real network traffic data set randomly, and plot their low-pass and high-pass components, respectively. Obviously, we see that the low-pass components are periodical, which means that they are much easier to be predicted comparing with the high-pass components shown by Figures 4(b) and 5(b). In this case, two components are predicted independently in this paper.

### 4.2. Deep Architecture for Low-Pass Component Prediction.

For an OD flow  $x_{p,q}(t)$ , we assume that the length of this series is an even number. In this case, the number of scaling coefficients is  $T/2$ . The deep architecture for prediction is plotted in Figure 6. There are  $M$  hidden layers in this architecture. Both the hidden and the input layers have  $T/2$  units. At the top of the deep architecture, a single neuron defined as the logistic regression is employed for prediction. The logistic regression model is made up of a hidden layer with  $T/2$  units and an output layer with one unit. The deep architecture is trained by the backpropagation algorithm in our method. The parameters of the deep architecture are determined lay by layer [18].

In our method, we first collect  $K$  training set denoted by  $(x_{p,q}^1(t), \dots, x_{p,q}^K(t))$ . Then, we have the scaling coefficient training set  $(c_{p,q}^1, \dots, c_{p,q}^K)$ . Meanwhile, the corresponding

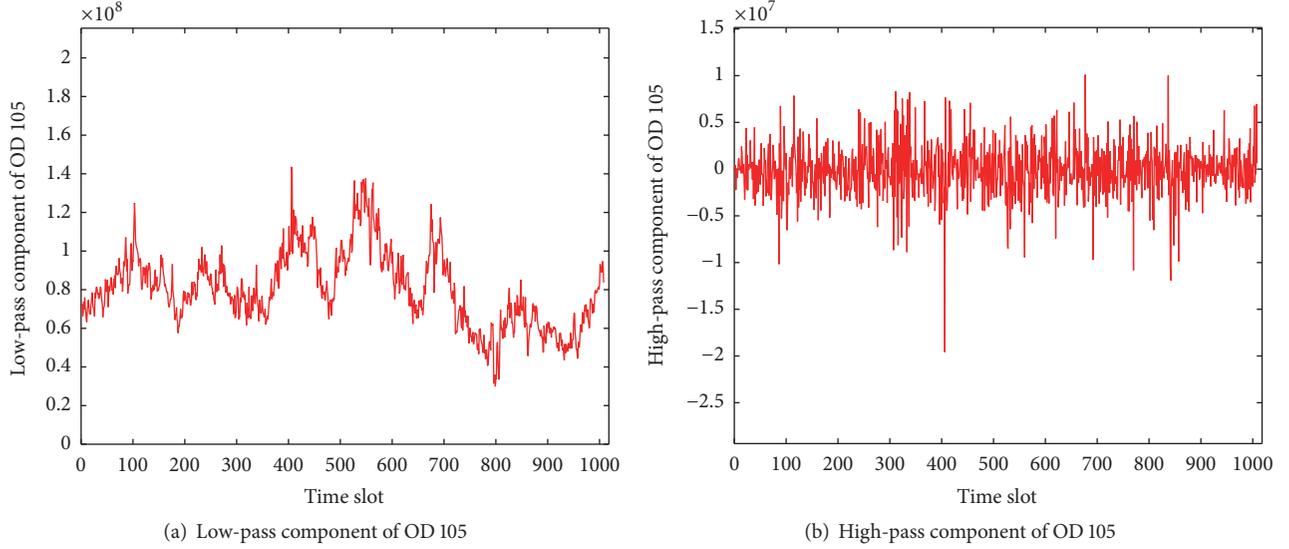


FIGURE 5: Decomposition of OD 105.

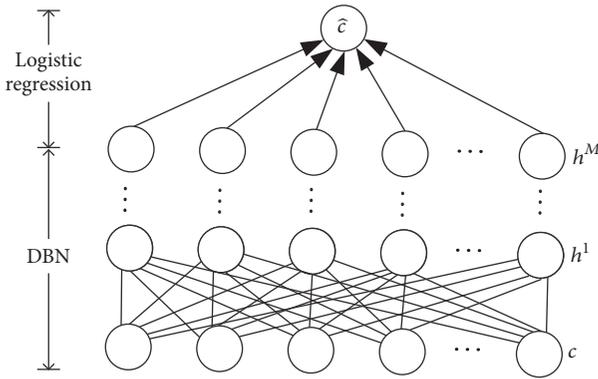


FIGURE 6: Deep architecture for prediction.

predictors are  $(\hat{c}_{p,q}^1, \dots, \hat{c}_{p,q}^K)$ . By training the proposed deep architecture using the scaling coefficient training set, we can obtain a relationship between input and output scaling coefficients. Furthermore, we use the scaling coefficients of  $x_{p,q}(t)$  as an input, and then a predictor of the scaling coefficient will be achieved.

**4.3. Spatiotemporal Compressive Sensing for High-Pass Component Prediction.** For the discrete wavelet transform coefficients, SRMF is a matrix-oriented interpolation algorithm. Therefore, different from the prediction of the scaling coefficients where each OD flow is predicted independently, we predict the discrete wavelet transform coefficients of all OD flows at the same time. According to (7), the discrete wavelet transform coefficients of all OD flows constitute a matrix denoted by  $D$  in this paper. The matrix  $D$  consists of two portions. One is from the training data obtained by measured network traffic data, and the other needs to be predicted. We denote the final predicting result by  $\hat{D}$ , and then it

can be predicted by the following regularized optimization model:

$$\|\hat{D} - D\|_F^2 + \lambda (\|L\|_F^2 + \|R\|_F^2) + \|(LR^T)H\|_F^2, \quad (8)$$

where the notations  $\|\cdot\|_F$  and  $(\cdot)^T$  denote the Frobenius norm and transposition, respectively. The matrix  $H$ , so-called the temporal constraint matrix (shown by (9)), describes the temporal neighbors.

$$H = \begin{bmatrix} 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}. \quad (9)$$

The matrices  $L$  and  $R$  are from the singular value decomposition of the matrix  $\hat{D}$ ; that is,

$$\hat{D} = U\Sigma V^T. \quad (10)$$

$U$  and  $T$  are two unitary matrices, and  $\Sigma$  is a diagonal matrix whose diagonal elements are the singular values of  $\hat{D}$ . The matrices  $L$  and  $R$  are equal to  $U\Sigma^{1/2}$  and  $V\Sigma^{1/2}$ .

Finally, according to the predictors of the scaling and discrete wavelet transform coefficients, we predict the network traffic by inverse discrete wavelet transform. Algorithm 1 proposes the details of our method.

## 5. Simulation Results and Analysis

This section will verify the performance of our prediction method. In our simulations, a real network traffic data set with 2016 time slots is sampled on a time scale of 5 minutes. For exact prediction, the first 2000 time slots are used as the prior information to train the deep architecture and the

```

(1) Input: training set  $(x_{p,q}^1(t), x_{p,q}^2(t), \dots, x_{p,q}^K(t))$ ; known network traffic  $X$ 
(2) Output: network traffic predictor  $X(T+1)$ 
(3) for  $i = 1 : K$  do
(4)    $c_{p,q}^i \leftarrow \text{DWT}(x_{p,q}^i(t))$ 
(5) end for
(6) Training deep architecture using  $(c_{p,q}^1, c_{p,q}^2, \dots, c_{p,q}^K)$  as a training set
(7)  $[C, D] \leftarrow \text{DWT}(X)$ 
(8)  $\widehat{C} \leftarrow \text{DBN}(C)$ 
(9)  $\widehat{D} \leftarrow \text{SRMF}(D)$ 
(10)  $X(T+1) \leftarrow \text{IDWT}(\widehat{C}, \widehat{D})$ 
(11) return  $X(T+1)$ 

```

ALGORITHM 1: DBNSTCS algorithm.

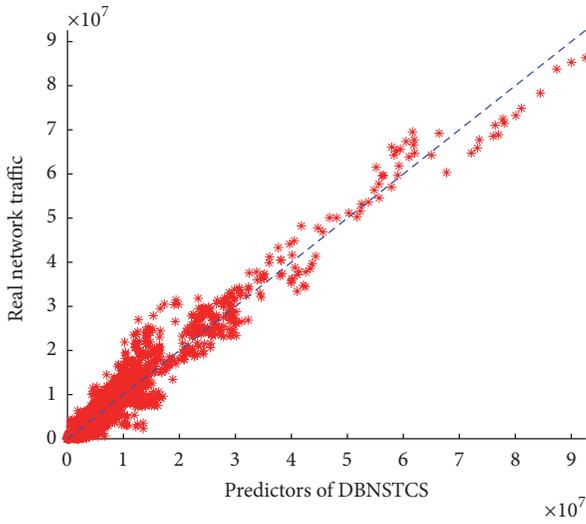


FIGURE 7: Real traffic data versus their predictors via DBNSTCS.

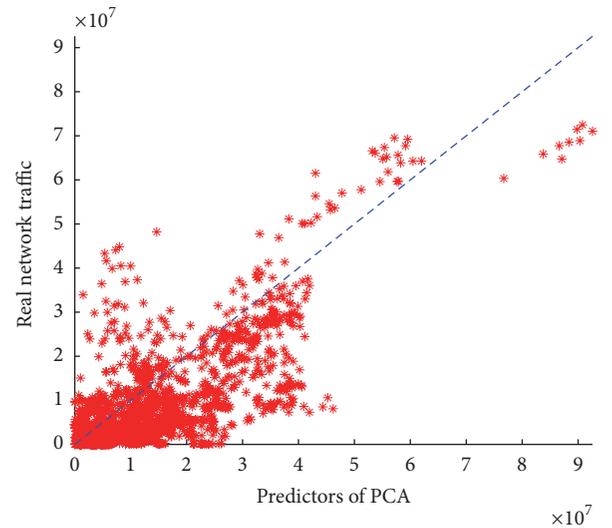


FIGURE 8: Real traffic data versus their predictors via PCA.

matrix  $D$ . We will compare our method with three state-of-the-art methods in network traffic prediction field, that is, the principal component analysis (PCA) method [20], the Tomogravity method [21], and the SRMF method [19]. The proposed method is implemented by MATLAB in a single machine with Core i5 central processing unit, 4 GB memory, and 1664 MB graphics processing unit (GPU) memory. Meanwhile, we set  $M = 8$  and  $T = 800$ .

We first plot the real network traffic versus their predictors from four methods, respectively. Figure 7 displays the prediction results of our method. The  $x$ -axis and  $y$ -axis denote predictors and real network traffic, respectively. From Figure 7, we see that our method has low prediction biases for small network traffic flows. By contrast, our method shows positive predictions for large network traffic. The same conclusion can be obtained from Figure 8. For large network traffic, it also has positive predictions. For small network traffic, PCA has much larger prediction bias. Tomogravity has consistently positive predictions for large network traffic shown by Figure 9. For small network traffic, Tomogravity shows a desired prediction error. Besides, for large network

traffic, SRMF in Figure 10 has positive or negative predictions more or less.

Now, we refer to the spatial and temporal relative errors as a metric to compare four methods. The spatial and temporal relative errors are defined as

$$\begin{aligned}
 \text{SRE}(n) &= \frac{\sqrt{\sum_{t=1}^T (\widehat{x}_n(t) - x_n(t))^2}}{\sqrt{\sum_{t=1}^T x_n^2(t)}}, \\
 \text{TRE}(t) &= \frac{\sqrt{\sum_{n=1}^{N^2} (\widehat{x}_n(t) - x_n(t))^2}}{\sqrt{\sum_{n=1}^{N^2} x_n^2(t)}},
 \end{aligned} \tag{11}$$

where  $x_n(t)$  and  $\widehat{x}_n(t)$  are the  $n$ th end-to-end network traffic flow and its predictor. As mentioned above,  $p, q \in \{1, 2, \dots, N\}$ ; thus the number of OD flows is  $N^2$ . Figure 11(a) exhibits the spatial relative errors (SREs) of four methods. The  $x$ -axis is the identities of end-to-end network traffic flows. The end-to-end network traffic flows are sorted in descending order with respect to their means. Meanwhile, the  $y$ -axis is the SRE. From this simulation, we find that

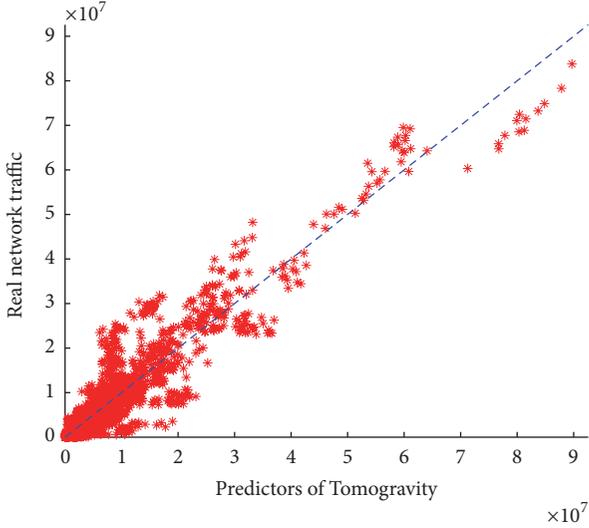


FIGURE 9: Real traffic data versus their predictors via Tomogravity.

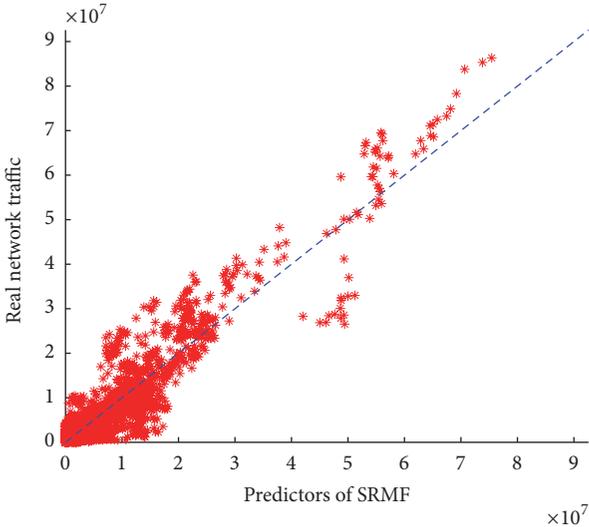


FIGURE 10: Real traffic data versus their predictors via SRMF.

our method has a consistently low SRE comparing with the other three methods. Figure 11(b) shows the TREs of four methods. From Figure 11(b), it shows that the TREs of PCA are much higher than that of the other methods. The TRE of DBNSTCS is the lowest one in four methods. The cumulative distributions of SRE and TRE are shown by Figure 12. It can show the prediction error more directly. Besides, we find that DBNSTCS has much more prominent improvement in TRE, comparing with SRE. That is because our method predicts the low-pass component of each traffic flow independently. In addition, contrasting to the high-pass component, this component has a significant effect on prediction. Hence, the weak improvement of SRE is caused by predicting the high-pass component using the Spatiotemporal Compressive Sensing method.

The diminutive error or bias of a prediction method does not mean it is available. Though it has low error, it fails to

provide precise predictors when it has high variance. Hence, the standard deviation is involved in our simulation as a metric for variance, which is defined as

$$SD(n) = \sqrt{\frac{1}{T-1} \sum_{t=1}^T (\hat{x}_n(t) - x_n(t) - b(n))^2}, \quad (12)$$

where  $b(n) = (1/T) \sum_{t=1}^T (\hat{x}_n(t) - x_n(t))$ . In (12),  $T$  is the length of predicted traffic data set. Figure 13 shows the bias versus standard deviation of four methods. We find that the four methods perform very differently with respect to variance. Tomogravity has larger variance compared with the other methods. In contrast, the PCA method exhibits relatively high variance. The DBNSTCS and SRMF methods show relatively low variance.

Finally, the performance improvement ratio is shown in Figure 14 as an overall evaluation. The performance improvement ratio is defined as

$$PIR = \frac{\sum_{n=1}^{N^2} \sum_{t=1}^T |x_{n,a}(t)| - \sum_{n=1}^{N^2} \sum_{t=1}^T |x_{n,b}(t)|}{\sum_{n=1}^{N^2} \sum_{t=1}^T |x_{n,a}(t)|}, \quad (13)$$

where  $x_{n,a}(t)$  and  $x_{n,b}(t)$  denote the predictors via the algorithms  $a$  and  $b$ , respectively. The performance improvement ratios of DBNSTCS are 68.74%, 5.24%, and 14.70% to PCA, Tomogravity, and SRMF.

## 6. Conclusions and Future Work

This paper focuses on the problem of network traffic prediction in wireless mesh backbone networks and proposes a hierarchical prediction method. The proposed hierarchical prediction method divides the network traffic into two components and then predicts each component by different models. In detail, the proposed method takes advantage of DBN and Spatiotemporal Compressive Sensing for network traffic prediction. In our method, the DWT is applied to dividing the network traffic into two components, that is, the long-range dependence component and the fluctuation component represented by the low-pass and high-pass components, respectively. A deep architecture consisting of a DBN layer and a logistic regression architecture is proposed to predict the low-pass component. Meanwhile, the other is predicted by the SRMF method which can capture the spatiotemporal characteristic of the high-pass component. We assess the performance of the proposed prediction method and compare it with three methods that are widely used for network traffic prediction. According to the simulation, our method is well in prediction error, especially in TRE. The main bottleneck of wireless mesh backbone network traffic prediction is the predicted accuracy for the irregular fluctuations of network traffic. Thereby, the prediction algorithm aiming at low-pass components is necessary in the future.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

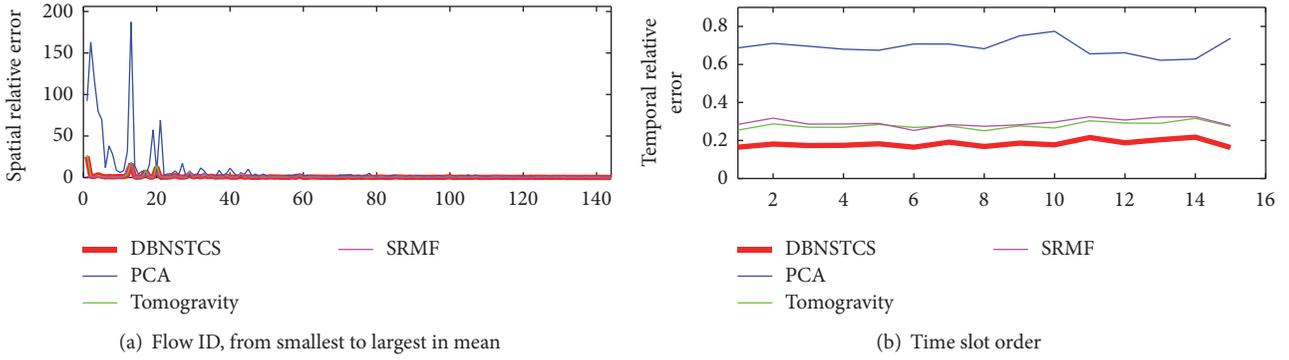


FIGURE 11: SRE and TRE.

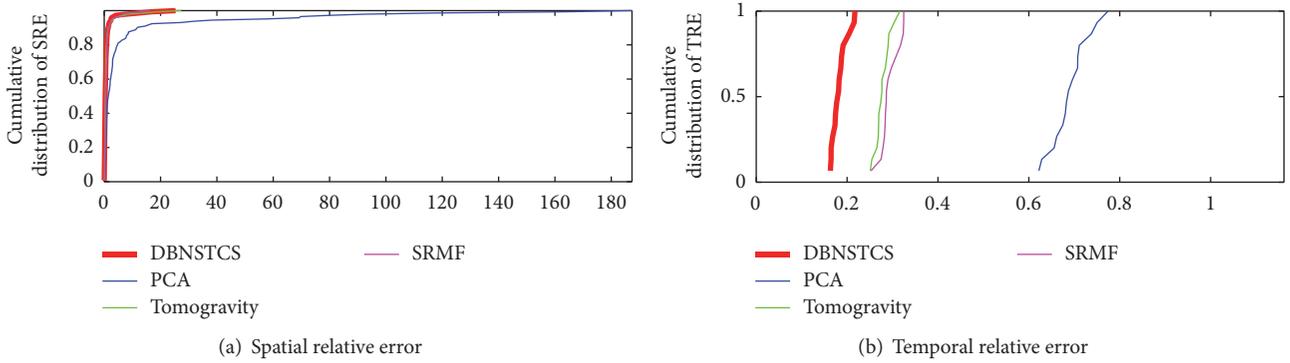


FIGURE 12: Cumulative distributions of SRE and TRE.

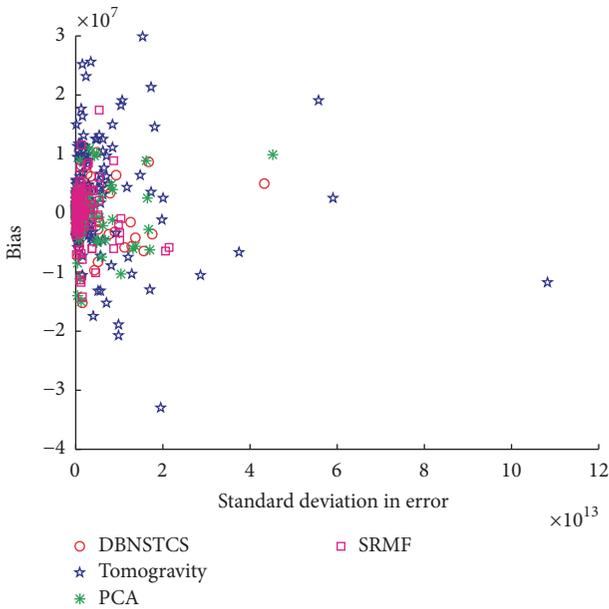


FIGURE 13: Bias versus standard deviation.

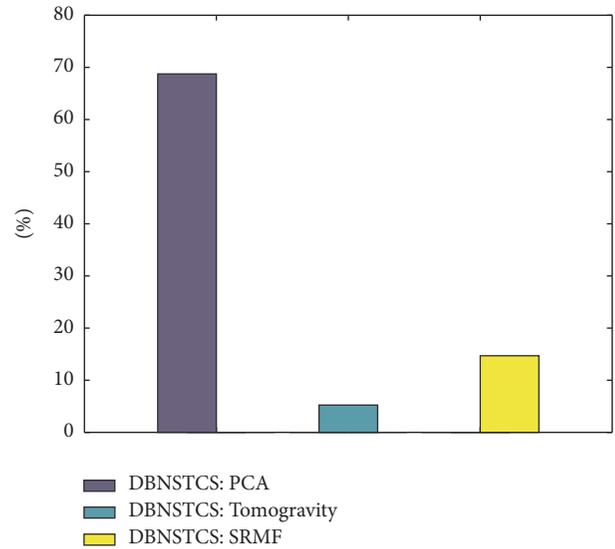


FIGURE 14: Performance improvement ratio.

**Acknowledgments**

This work was supported by the National Natural Science Foundation of China (61701406, 61701413) and the Fundamental Research Funds for the Central Universities (3102017OQD022).

**References**

- [1] T. Qiu, R. Qiao, and D. Wu, "EABS: an event-aware backpressure scheduling scheme for emergency internet of things," *IEEE Transactions on Mobile Computing*, 2017.
- [2] T. B. Reddy, B. S. Manoj, and R. Rao, "On the accuracy of sampling schemes for wireless network characterization," in

- Proceedings of the IEEE Wireless Communications and Networking Conference, WCNC 2008*, pp. 3314–3319, USA, April 2008.
- [3] T. Qiu, A. Zhao, F. Xia, W. Si, and D. O. Wu, “ROSE: Robustness Strategy for Scale-Free Wireless Sensor Networks,” *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2944–2959, 2017.
  - [4] B. Zhou, D. He, and Z. Sun, “Traffic predictability based on ARIMA/GARCH model,” in *Proceedings of the 2006 2nd Conference on Next Generation Internet Design and Engineering, NGI 2006*, pp. 200–207, Spain, April 2006.
  - [5] P. Kuanhoong, I. Tan, and C. YikKeong, “BitTorrent network traffic forecasting with ARIMA,” vol. 4, pp. 143–156, 2012.
  - [6] T. Qiu, Y. Zhang, D. Qiao, X. Zhang, M. L. Wymore, and A. K. Sangaiah, “A Robust Time Synchronization Scheme for Industrial Internet of Things,” *IEEE Transactions on Industrial Informatics*, pp. 1-1.
  - [7] P. Calyam and A. Devulapalli, “Modeling of multi-resolution active network measurement time-series,” in *Proceedings of the 33rd IEEE Conference on Local Computer Networks, LCN 2008*, pp. 900–907, Canada, October 2008.
  - [8] J. Huang, C. Xing, and C. Wang, “Simultaneous wireless information and power transfer: technologies, applications, and research challenges,” *IEEE Communications Magazine*, vol. 55, no. 11, pp. 26–32, 2017.
  - [9] L. Nie, D. Jiang, S. Yu, and H. Song, “Network traffic prediction based on deep belief network in wireless mesh backbone networks,” in *Proceedings of the 2017 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–5, San Francisco, CA, USA, March 2017.
  - [10] N. C. Anand, C. Scoglio, and B. Natarajan, “GARCH - Non-linear time series model for traffic modeling and prediction,” in *Proceedings of the NOMS 2008 - IEEE/IFIP Network Operations and Management Symposium: Pervasive Management for Ubiquitous Networks and Services*, pp. 694–697, Brazil, April 2008.
  - [11] J. Huang, Q. Duan, C.-C. Xing, and H. Wang, “Topology control for building a large-scale and energy-efficient internet of things,” *IEEE Wireless Communications Magazine*, vol. 24, no. 1, pp. 67–73, 2017.
  - [12] M. Joshi and T. Hadi, *A review of network traffic analysis and prediction techniques*, Computer Science, 2015.
  - [13] Y. Liu, B. Li, X. Sun, and Z. Zhou, “A fusion model of SWT, QGA and BP neural network for wireless network traffic prediction,” in *Proceedings of the 15th IEEE International Conference on Communication Technology (ICCT '13)*, pp. 769–774, IEEE, Guilin, China, November 2013.
  - [14] F. Xu, Y. Lin, J. Huang et al., “Big data driven mobile traffic understanding and forecasting: a time series approach,” *IEEE Transactions on Services Computing*, vol. 9, no. 5, pp. 796–805, 2016.
  - [15] W. Huang, G. Song, H. Hong, and K. Xie, “Deep architecture for traffic flow prediction: deep belief networks with multitask learning,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2191–2201, 2014.
  - [16] G. E. Hinton, S. Osindero, and Y.-W. Teh, “A fast learning algorithm for deep belief nets,” *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.
  - [17] J. Huang, Q. Duan, Y. Zhao, Z. Zheng, and W. Wang, “Multicast Routing for Multimedia Communications in the Internet of Things,” *IEEE Internet of Things Journal*, vol. 4, no. 1, pp. 215–224, 2017.
  - [18] R. B. Palm, “Prediction as a candidate for learning deep hierarchical models of data,” Tech. Rep., Technical University of Denmark, 2012.
  - [19] M. Roughan, Y. Zhang, W. Willinger, and L. Qiu, “Spatio-temporal compressive sensing and internet traffic matrices (Extended Version),” *IEEE/ACM Transactions on Networking*, vol. 20, no. 3, pp. 662–676, 2012.
  - [20] A. Soule, A. Lakhina, N. Taft et al., “Traffic matrices: Balancing measurements, inference and modeling,” in *Proceedings of the SIGMETRICS 2005: International Conference on Measurement and Modeling of Computer Systems*, pp. 362–373, Canada, June 2005.
  - [21] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, “Computation of IP traffic from link,” in *Proceedings of the ACM SIGMETRICS 2003 - International Conference on Measurement and Modeling of Computer Systems*, pp. 206–217, USA, June 2003.

## Research Article

# Channel Access and Power Control for Mobile Crowdsourcing in Device-to-Device Underlaid Cellular Networks

Yue Ma <sup>1</sup>, Li Zhou <sup>2</sup>, Zhenghua Gu,<sup>1</sup> Yang Song,<sup>1</sup> and Bin Wang<sup>1</sup>

<sup>1</sup>China Aerodynamics Research and Development Center, Mianyang 621000, China

<sup>2</sup>School of Electronic Science, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Li Zhou; [zhouli2035@nudt.edu.cn](mailto:zhouli2035@nudt.edu.cn)

Received 2 September 2017; Accepted 4 December 2017; Published 4 January 2018

Academic Editor: Kun Bai

Copyright © 2018 Yue Ma 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 access of a myriad of smart handheld devices in cellular networks, mobile crowdsourcing becomes increasingly popular, which can leverage omnipresent mobile devices to promote the complicated crowdsourcing tasks. Device-to-device (D2D) communication is highly desired in mobile crowdsourcing when cellular communications are costly. The D2D cellular network is more preferable for mobile crowdsourcing than conventional cellular network. Therefore, this paper addresses the channel access and power control problem in the D2D underlaid cellular networks. We propose a novel semidistributed network-assisted power and a channel access control scheme for D2D user equipment (DUE) pieces. It can control the interference from DUE pieces to the cellular user accurately and has low information feedback overhead. For the proposed scheme, the stochastic geometry tool is employed and analytic expressions are derived for the coverage probabilities of both the cellular link and D2D links. We analyze the impact of key system parameters on the proposed scheme. The Pareto optimal access threshold maximizing the total area spectral efficiency is obtained. Unlike the existing works, the performances of the cellular link and D2D links are both considered. Simulation results show that the proposed method can improve the total area spectral efficiency significantly compared to existing schemes.

## 1. Introduction

With the development of intelligent handheld devices, mobile crowdsourcing is emerging as an efficient strategy to improve the user experiences of applications in wireless networks [1]. With the help of ubiquitous mobile users who can share the information between each other, the large-scale crowdsourcing tasks can be facilitated [2]. The D2D communications can realize the high-speed data transmission among mobile users with lower power and shorter delay, which largely eases collaborations among participants [3–6]. Therefore, the D2D cellular network is more preferable for mobile crowdsourcing than conventional cellular network [2].

Despite making full use of radio resources, the underlaid scenario leads to the coexistence of cellular and D2D communications in the same frequency band, which brings cross-tier and cotier interferences. Therefore, interference management becomes essential in the underlaid scenario.

Transmit power control and channel allocation are widely used to mitigate interference in wireless networks. Different from ad hoc networks and traditional D2D technologies (e.g., WiFi direct), the base station (BS) can assist the DUE pieces in controlling transmit power and allocate the subchannels in the underlaid D2D cellular networks [7]. Compared to DUE pieces, the BS has a more powerful signal processing ability and more easily obtains useful information for resource allocation. Generally, the resource management methods in D2D underlaid cellular networks can be classified into two main categories: the network-controlled device-assisted method and the network-assisted device-decided method [8–14]. In the network-controlled device-assisted method, each DUE needs to report its own information (e.g., channel state information, CSI) to the BS, and then the BS allocates the radio resource to each DUE [8–10]. In [8], a joint power control and mode selection strategy was proposed to improve the overall system performance. To protect existing cellular

links, the authors in [9] proposed a dynamic power control method to limit the D2D transmission power. In [10], the centralized algorithm was used to control the power of both the cellular user equipment (CUE) and DUE pieces. Note that the network-controlled device-assisted method requires global information at the BS. However, when the density of DUE pieces is high, the overhead to transmit global information to the BS will reduce the system efficiency significantly and it may be unacceptable for real-time processing.

For the network-assisted device-decided method, each DUE selects suitable channels and decides the transmission power according to a minimum transmit power criterion and the power control instruction from the BS [11]. In this scheme, the BS does not need to know global information about each DUE and the overhead can be reduced significantly. In [11], the authors proposed a network-assisted device-decided scheme to jointly select the channels and adjust the transmission power for DUE pieces. However, the number of active DUE pieces on one certain channel is not limited, which may lead to severe interference among the DUE pieces. In [10], based on the statistical features of CSI, distributed on-off power control algorithm was proposed and the fixed threshold was applied to control the number of active DUE pieces. The fixed threshold in [10] is the same for all DUE pieces. In [12], to minimize the D2D transmit power, a statistical features-based power control with opportunistic access control was proposed. Unlike [10], each DUE has its own optimal access threshold in [12].

However, most existing channel access schemes only consider the performance of DUE pieces, while the performance of the cellular user is ignored [10–12]. Moreover, although the works in [10, 12] control the power of DUE pieces according to a minimum transmit power criterion, the interference from DUE pieces to the cellular user is not controlled accurately. In this paper, we focus on the power and channel access control problem in a network-assisted device-decided scheme. The main contributions of this paper are as follows:

- (1) A novel network-assisted power and channel access control scheme is presented. The proposed scheme can control the interference from DUE pieces to the cellular user accurately. Moreover, the proposed scheme is semidistributed, which can guarantee the performances of both the cellular link and D2D links without high information feedback overhead.
- (2) For the proposed method, we apply the stochastic geometry tool and derive analytic expressions including the coverage probabilities of both the cellular link and D2D links.
- (3) The Pareto optimal access threshold maximizing the total area spectral efficiency is obtained. Different from the existing works, the performances of the cellular link and D2D links are both considered in obtaining the optimal access threshold.

In the D2D cellular network, both the cellular users and D2D users may participate in the crowdsourcing tasks. The proposed scheme considers the performances of both the cellular link and D2D links, which can allow cellular users

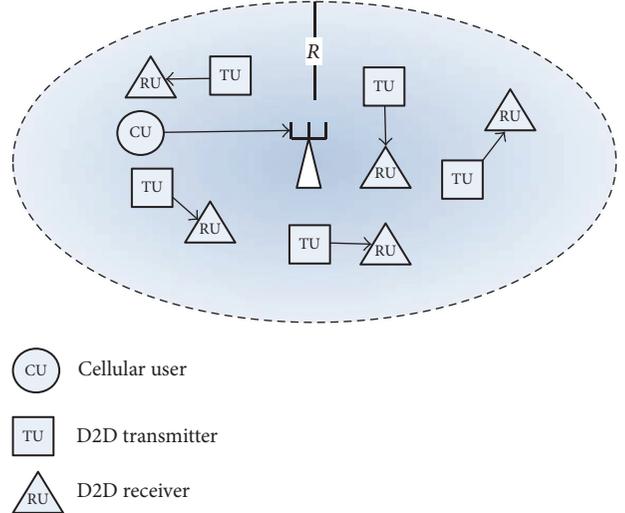


FIGURE 1: Illustration of a D2D underlaid cellular network.

and D2D users to better carry out crowdsourcing tasks. On the other hand, large-scale crowdsourcing tasks need the participation of ubiquitous mobile users, which is based on the network with high performance. The proposed scheme can improve the system efficiency significantly without high information feedback overhead, which is preferable for large-scale crowdsourcing tasks.

## 2. System Model

The architecture of a D2D underlaid network is considered, as shown in Figure 1. In this model, we denote the circular disk  $C$  with radius  $R$  as the coverage region of a base station (BS) centered at the origin. There are one BS (i.e.,  $k_0$ ) and  $N_d^{\text{tol}}$  D2D pairs. We assume that the cell has one cellular user equipment (CUE) piece on one subchannel. Denote the CUE as  $s_0$ . Each D2D pair comprises one transmit DUE (TU) piece and one DUE reception (RU) piece. Denote TU and RU pieces as  $(u_{t_1}, u_{t_2}, \dots, u_{t_{N_d^{\text{tol}}}})$  and  $(u_{r_1}, u_{r_2}, \dots, u_{r_{N_d^{\text{tol}}}})$ . The DUE pieces can share the uplink subchannel of the CUE  $s_0$ . All nodes are equipped with one antenna. The CUE is uniformly located in the region  $C$ . The locations of the TU pieces follow the independent homogeneous Poisson point processes (PPPs)  $\Phi_{\text{DU}}$  with density  $\lambda_{\text{DU}}$ . The intended RU pieces are uniformly and independently located within the distance  $R_d$  of their associated TU pieces. Under the given assumptions,  $\mathbb{E}[N_d^{\text{tol}}] = \lambda_{\text{DU}}\pi R^2$ . The CUE operates at a fixed transmission power, defined as  $P_{s_0}$ .

## 3. Network-Assisted Channel Access and Power Control for DUE Pieces

**3.1. Semidistributed Channel Access.** In this subsection, we present a semidistributed channel access method, which is an effective interference mitigation method that requires assisted information from the BS. In Figure 2, the subchannel represents a particular time slot. For example, at slot 1, the

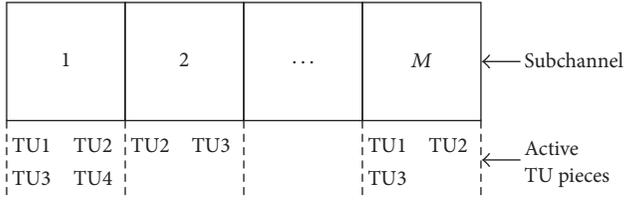


FIGURE 2: The illustration of D2D user's channel access.

active TU1, TU2, TU3, and TU4 use the same frequency band. As in [10–12], the link-gain is applied as the access metric. We use a nonnegative threshold to limit the number of DUE pieces on the subchannel and reduce the interference to cellular users. Moreover, each D2D pair has its own threshold, which is calculated by itself based on the statistical parameters from the BS. The access threshold is calculated in TU. The RU can obtain the link-gain by channel estimation and feed back it to the TU.  $G_{u_n,i}$  denotes the threshold for the D2D pair  $u_n$  (i.e.,  $(u_t, u_r)$ ) on the subchannel  $i$ . Then, in order to avoid the overheads for reporting global information to the BS, each D2D pair decides whether to access the current subchannel solely based on its own link-gain and the threshold independently. Specifically, when the link-gain of the D2D pair  $u_n$  on subchannel  $i$  is larger than  $G_{u_n,i}$ , the D2D pair  $u_n$  is active on the subchannel  $i$ ; otherwise, the D2D pair  $u_n$  is idle on the subchannel  $i$ .

The transmit probability of the D2D pair  $u_n$  in the subchannel  $i$  is given by

$$\begin{aligned} PS_{u_n,i} &= \mathbb{P} \left[ |h_{u_t, u_r, i}|^2 d_{u_t, u_r, i}^{-\alpha} > G_{u_n,i} \right] \\ &= \exp \left( -G_{u_n,i} d_{u_t, u_r, i}^{\alpha} \right), \end{aligned} \quad (1)$$

where  $h_{u_t, u_r, i}$  and  $d_{u_t, u_r, i}$  represent the small-scale fast fading and distance from TU  $u_t$  to RU  $u_r$  on subchannel  $i$ , respectively.  $\alpha$  is the path-loss exponent. The Rayleigh distribution is assumed (i.e.,  $h_{u_t, u_r, i} \sim \mathcal{CN}(0, 1)$ ). The proposed algorithm is semidistributed as each D2D pair decides whether to access the subchannel by its own channel gain  $|h_{u_t, u_r, i}|^2$  and network-assisted threshold  $G_{u_n,i}$ . Note that the threshold  $G_{u_n,i}$  not only has an influence on the performance of cellular users, but also determines the performance of the D2D links. On the one hand, larger  $G_{u_n,i}$  reduce both the interference from DUE pieces to cellular users and the inter-D2D interference. On the other hand, larger  $G_{u_n,i}$  leads to smaller number of active DUE pieces. Therefore, in order to balance these two competing factors,  $G_{u_n,i}$  is optimized in Section 5.

**3.2. Semidistributed Power Control.** Based on the channel access method in Section 3.1, the semidistributed power control strategy is presented as shown in Figure 3.

The core idea of the proposed power control strategy mainly includes the following.

*Step 1.* The BS broadcasts the statistical parameters for each D2D pair to calculate the threshold  $G_{u_n,i}$ .

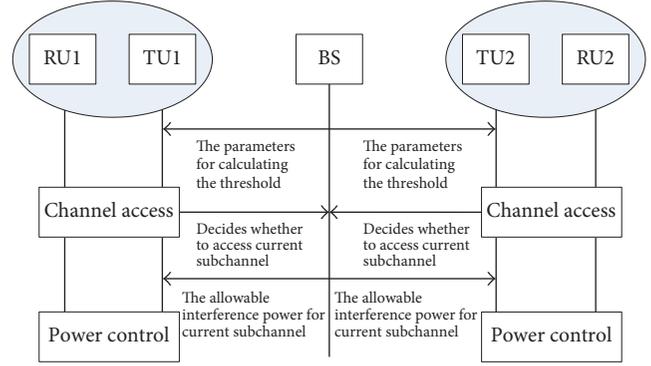


FIGURE 3: Network-assisted, device-decided channel access and power control scheme.

*Step 2.* Each D2D pair decides whether to access the current subchannel by its own channel gain  $|h_{u_t, u_r, i}|^2$  and threshold  $G_{u_n,i}$  and then transmits the result to the BS.

*Step 3.* The BS calculates the allowable power for each TU according to both the CUE performance requirement and the number of active D2D pairs on the current subchannel and then transmits the allowable power to each TU.

*Step 4.* According to the assistant power instruction broadcasted from the BS and the maximum transmission power for the TU, each TU autonomously adjusts the transmit power independently.

Steps 1 and 2 refer to Section 3.1. In the following, without loss of generality, we take the subchannel  $i$  as an example and the index of subchannel  $i$  will be ignored. For example,  $h_{u_t, u_r, i}$  will be replaced by  $h_{u_t, u_r}$ . The detailed processes of Steps 3 and 4 will be presented. The conclusions can be easily extended to other subchannels.

In order to ensure the link reliability of the CUE, the proposed power control aims at adjusting the transmission power of TU pieces. Denote the target  $I_{k_0}$  as the allowable interference power from TU pieces to the CUE on the subchannel  $i$ . According to Steps 1 and 2 of the proposed power control algorithm, the BS can count the number of active TU pieces that prefer the subchannel  $i$  to perform the D2D communications. Hence, the allowable transmission power unit for TU  $u_t$  is given by

$$PL_{u_t} = \frac{I_{k_0}}{N_d \left( |h_{u_t, k_0}|^2 d_{u_t, k_0}^{-\alpha} \right)}, \quad (2)$$

where  $N_d$  is the number of active D2D pairs on the subchannel  $i$ , and  $\mathbb{E}[N_d] = \lambda_{DU} PS_{u_n} \pi R^2$ .  $PS_{u_n}$  is the access probability for D2D links on the subchannel  $i$ .  $h_{u_t, k_0}$  and  $d_{u_t, k_0}$  represent the small-scale fast fading and distance from TU  $u_t$  to BS  $k_0$  on  $i$ th subchannel, respectively. The Rayleigh distribution is assumed (i.e.,  $h_{u_t, k_0} \sim \mathcal{CN}(0, 1)$ ). From (2), it can be observed that less interference link-gain leads to larger

allowable transmission power unit, which means that the TU far from the BS is more likely to transmit with larger power.

Thus, the transmission power of TU  $u_{t_n}$  can be expressed as

$$P_{u_{t_n}} = \min \{ PL_{u_{t_n}}, P_{u_{t_n}, \max} \}, \quad (3)$$

where  $P_{u_{t_n}, \max}$  is the maximum transmission power for TU  $u_{t_n}$  on  $i$ th subchannel. Moreover, we can get

$$P_{u_{t_n}} = \begin{cases} PL_{u_{t_n}}, & \text{with } P_{b, u_{t_n}} \\ P_{u_{t_n}, \max}, & \text{with } 1 - P_{b, u_{t_n}}, \end{cases} \quad (4)$$

where  $P_{b, u_{t_n}}$  denotes the probability given by

$$\begin{aligned} P_{b, u_{t_n}} &= \mathbb{P} \left[ P_{u_{t_n}, \max} > PL_{u_{t_n}} \right] \\ &= \mathbb{P} \left[ \left| h_{u_{t_n}, k_0} \right|^2 d_{u_{t_n}, k_0}^{-\alpha} > \frac{I_{k_0}}{N_d P_{u_{t_n}, \max}} \right] \\ &= \exp \left( -d_{u_{t_n}, k_0}^{-\alpha} \frac{I_{k_0}}{N_d P_{u_{t_n}, \max}} \right). \end{aligned} \quad (5)$$

Note that the proposed power control is semidistributed as each TU decides the transmission power according to the allowable transmission power unit broadcasted by the BS and its own maximum power.

## 4. Performance Analysis

In this section, we analyze the proposed scheme in terms of the coverage probability. The performances of the cellular link and D2D links are both analyzed.

*4.1. The Cellular Coverage Probability.* The uplink SINR from CUE to the BS on  $i$ th subchannel can be expressed as [11]

$$\tilde{\gamma}_{s_0, k_0} = \frac{P_{s_0} |h_{s_0, k_0}|^2 d_{s_0, k_0}^{-\alpha}}{I_{k_0} + \sigma_{k_0}^2}, \quad (6)$$

where  $P_{s_0}$  denotes the transmission power of CUE on  $i$ th subchannel.  $h_{s_0, k_0}$  and  $d_{s_0, k_0}$  represent the small-scale fast fading and distance from CUE  $s_0$  to BS  $k_0$  on  $i$ th subchannel, respectively.  $h_{s_0, k_0} \sim \mathcal{CN}(0, 1)$ .  $I_{k_0} = \sum_{u_{t_n} \in \Phi_{DU}} P_{u_{t_n}} |h_{u_{t_n}, k_0}|^2 d_{u_{t_n}, k_0}^{-\alpha}$  represents the total uplink interference from TU pieces to CUE  $s_0$  in the BS  $k_0$  on the  $i$ th subchannel.  $\sigma_{k_0}^2$  is the power of the additive Gaussian white noise at  $k_0$ .

**Lemma 1.** *The coverage probability of the cellular link can be expressed as*

$$P_{cc}(\eta) = \mathbb{E}_{d_{s_0, k_0}} \left\{ \exp(-A_1 X_1 - A_2 X_1^{2/\alpha}) \right\}, \quad (7)$$

where  $A_1 = \sigma_{k_0}^2 \eta$ ,  $A_2 = (\pi \lambda_{DU} P_{S_{u_{t_n}}} \eta^{2/\alpha} / \sin c(2/\alpha)) \mathbb{E}[P_{u_{t_n}}^{2/\alpha}]$ ,  $X_1 = P_{s_0}^{-1} d_{s_0, k_0}^{-\alpha}$ , and  $\eta$  represents the minimum SINR value for reliable uplink connection.

*Proof.* The coverage probability of the cellular link can be expressed as [10, 12]

$$\begin{aligned} P_{cc}(\eta) &= \mathbb{P}(\tilde{\gamma}_{s_0, k_0} \geq \eta) = \mathbb{P} \left( \frac{P_{s_0} |h_{s_0, k_0}|^2 d_{s_0, k_0}^{-\alpha}}{I_{k_0} + \sigma_{k_0}^2} \geq \eta \right) \\ &= \mathbb{P} \left( |h_{s_0, k_0}|^2 \geq \eta P_{s_0}^{-1} d_{s_0, k_0}^{-\alpha} (I_{k_0} + \sigma_{k_0}^2) \right) \\ &\stackrel{(a)}{=} \mathbb{E} \left[ \exp(-\eta P_{s_0}^{-1} d_{s_0, k_0}^{-\alpha} (I_{k_0} + \sigma_{k_0}^2)) \right] \\ &= \mathbb{E} \left\{ e^{-\eta P_{s_0}^{-1} d_{s_0, k_0}^{-\alpha} \sigma_{k_0}^2} e^{-\eta P_{s_0}^{-1} d_{s_0, k_0}^{-\alpha} \sum_{u_{t_n} \in \Phi_{DU}} P_{u_{t_n}} |h_{u_{t_n}, k_0}|^2 d_{u_{t_n}, k_0}^{-\alpha}} \right\}, \end{aligned} \quad (8)$$

where in the equality (a) we use the fact that  $|h_{s_0, k_0}|^2 \sim \exp(1)$  [10]. Denote  $X_1 = P_{s_0}^{-1} d_{s_0, k_0}^{-\alpha}$ . Conditioning on  $d_{s_0, k_0}^{-\alpha}$ , we can obtain

$$\begin{aligned} &\mathbb{P}(\tilde{\gamma}_{s_0, k_0} \geq \eta) \Big|_{d_{s_0, k_0}^{-\alpha}} \\ &= e^{-\eta P_{s_0}^{-1} d_{s_0, k_0}^{-\alpha} \sigma_{k_0}^2} e^{-\eta P_{s_0}^{-1} d_{s_0, k_0}^{-\alpha} \sum_{u_{t_n} \in \Phi_{DU}} P_{u_{t_n}} |h_{u_{t_n}, k_0}|^2 d_{u_{t_n}, k_0}^{-\alpha}}. \end{aligned} \quad (9)$$

The Laplace transform is applied as follows [15]:

$$\begin{aligned} L_{\Phi_D}(s) &= e^{-s(\sum_{u_{t_n} \in \Phi_{DU}} P_{u_{t_n}} |h_{u_{t_n}, k_0}|^2 d_{u_{t_n}, k_0}^{-\alpha})} \\ &= e^{-(\pi \lambda_{DU} P_{S_{u_{t_n}}} / \text{sinc}(2/\alpha)) \mathbb{E}[P_{u_{t_n}}^{2/\alpha}] s^{2/\alpha}}. \end{aligned} \quad (10)$$

Deconditioning with respect to  $d_{s_0, k_0}^{-\alpha}$ , we combine (9) and (10), and then (7) can be obtained.

According to (7), we can observe that the D2D-related parameter  $\mathbb{E}[P_{u_{t_n}}^{2/\alpha}]$  affects the cellular link coverage probability. In the following, by giving the expression of  $\mathbb{E}[P_{u_{t_n}}^{2/\alpha}]$ , we will present how the proposed power control method affects the performance of the cellular link.

Denote  $z = |h_{u_{t_n}, k_0}|^2 d_{u_{t_n}, k_0}^{-\alpha}$ ; then  $\mathbb{E}[P_{u_{t_n}}^{2/\alpha}]$  can be expressed as

$$\begin{aligned} \mathbb{E}[P_{u_{t_n}}^{2/\alpha}] &= \int_0^{+\infty} \min \{ PL_{u_{t_n}}^{2/\alpha}, P_{u_{t_n}, \max}^{2/\alpha} \} f(z) dz \\ &= \int_0^{+\infty} \min \left\{ \left( \frac{I_{k_0}}{N_d z} \right)^{2/\alpha}, P_{u_{t_n}, \max}^{2/\alpha} \right\} f(z) dz. \end{aligned} \quad (11)$$

Conditioning on the distance  $d_{u_{t_n}, k_0}$  and  $I_{k_0}$ ,  $\mathbb{E}[P_{u_{t_n}}^{2/\alpha}]$  is given by

$$\begin{aligned} &\mathbb{E}[P_{u_{t_n}}^{2/\alpha}] \Big|_{d_{u_{t_n}, k_0}} \\ &= \int_0^{+\infty} \min \left\{ \left( \frac{I_{k_0}}{N_d d_{u_{t_n}, k_0}^{-\alpha} x} \right)^{2/\alpha}, P_{u_{t_n}, \max}^{2/\alpha} \right\} \\ &\quad \cdot \exp(-x) dx \\ &= \min \left\{ \int_0^{+\infty} \left( \frac{I_{k_0}}{N_d d_{u_{t_n}, k_0}^{-\alpha} x} \right)^{2/\alpha} \exp(-x) dx, \right. \end{aligned}$$

$$\begin{aligned}
& \int_0^{+\infty} P_{u_{t_n}, \max}^{2/\alpha} \exp(-x) dx \Big\} \\
& = \min \left\{ \int_0^{+\infty} \left( \frac{I_{k_0}}{N_d d_{u_{t_n}, k_0}^{-\alpha}} \right)^{2/\alpha} \exp(-x) dx, \right. \\
& P_{u_{t_n}, \max}^{2/\alpha} \Big\} = \min \left\{ \left( \frac{I_{k_0}}{N_d d_{u_{t_n}, k_0}^{-\alpha}} \right)^{2/\alpha} \Gamma\left(\frac{-2+\alpha}{\alpha}\right), \right. \\
& P_{u_{t_n}, \max}^{2/\alpha} \Big\}, \tag{12}
\end{aligned}$$

where  $\Gamma(x)$  denotes the gamma function. Then, deconditioning with respect to  $d_{u_{t_n}, k_0}$ , we can obtain

$$\begin{aligned}
\mathbb{E} \left[ P_{u_{t_n}}^{2/\alpha} \right] & = \min \left\{ \mathbb{E} \left[ \left( \frac{I_{k_0}}{N_d d_{u_{t_n}, k_0}^{-\alpha}} \right)^{2/\alpha} \Gamma\left(\frac{-2+\alpha}{\alpha}\right) \right], \right. \\
& P_{u_{t_n}, \max}^{2/\alpha} \Big\} \\
& = \min \left\{ \mathbb{E} \left[ \left( \frac{I_{k_0}}{N_d} \right)^{2/\alpha} \Gamma\left(\frac{-2+\alpha}{\alpha}\right) d_{u_{t_n}, k_0}^2 \right], \right. \\
& P_{u_{t_n}, \max}^{2/\alpha} \Big\} = \min \left\{ \left( \frac{I_{k_0}}{N_d} \right)^{2/\alpha} \Gamma\left(\frac{-2+\alpha}{\alpha}\right) \mathbb{E} \left[ d_{u_{t_n}, k_0}^2 \right], \right. \\
& P_{u_{t_n}, \max}^{2/\alpha} \Big\} \stackrel{(a)}{=} \min \left\{ \left( \frac{I_{k_0}}{N_d} \right)^{2/\alpha} \Gamma\left(\frac{-2+\alpha}{\alpha}\right) R^2, \right. \\
& P_{u_{t_n}, \max}^{2/\alpha} \Big\}, \tag{13}
\end{aligned}$$

where  $\mathbb{E}[d_{u_{t_n}, k_0}^2] = R^2$  given in [13]. From (13), we can see that the proposed network-assisted power control method can control the interferences from TU pieces to the CUE accurately by adjusting the parameter  $I_{k_0}$ . The coverage probability of the cellular link decreases with  $I_{k_0}$ . Larger  $I_{k_0}$  leads to more serious interference.

Generally, we set  $I_{k_0}$  to keep the uplink SINR of the CUE to be above the target  $\eta$ . In D2D underlaid cellular networks, the communication link is usually more severely affected by the interference compared with the noise. Hence, we ignore the effect of noise by setting  $\sigma_{k_0}^2 = 0$  to facilitate the theoretical analysis. In the interference limited regime ( $\sigma_{k_0}^2 = 0$ ), we have

$$I_{k_0} = \frac{P_{s_0} |h_{s_0, k_0}|^2 d_{s_0, k_0}^{-\alpha}}{\eta}. \tag{14}$$

Then, plugging (14) into (13) and deconditioning with respect to  $|h_{s_0, k_0}|^2$ , it can be obtained that

$$\begin{aligned}
\mathbb{E} \left[ P_{u_{t_n}}^{2/\alpha} \right] & = \min \left\{ \mathbb{E} \left[ \left( \frac{I_{k_0}}{N_d} \right)^{2/\alpha} \Gamma\left(\frac{-2+\alpha}{\alpha}\right) R^2 \right], \right. \\
& P_{u_{t_n}, \max}^{2/\alpha} \Big\} = \min \left\{ \mathbb{E} \left[ \left( \frac{P_{s_0} |h_{s_0, k_0}|^2 d_{s_0, k_0}^{-\alpha}}{\eta N_d} \right)^{2/\alpha} \right. \right. \\
& \cdot \Gamma\left(\frac{-2+\alpha}{\alpha}\right) R^2 \Big], P_{u_{t_n}, \max}^{2/\alpha} \Big\} \\
& \stackrel{(a)}{=} \min \left\{ \left( \frac{P_{s_0} d_{s_0, k_0}^{-\alpha}}{N_d \eta} \right)^{2/\alpha} \Gamma\left(\frac{-2+\alpha}{\alpha}\right) \Gamma\left(\frac{2+\alpha}{\alpha}\right) R^2, \right. \\
& P_{u_{t_n}, \max}^{2/\alpha} \Big\}, \tag{15}
\end{aligned}$$

where the equality (a) follows from  $|h_{s_0, k_0}|^2 \sim \exp(1)$ .  $\square$

**4.2. The Performance of D2D Links.** The SINR at RU  $u_{r_n}$  on  $i$ th subchannel can be expressed as [11]

$$\tilde{\gamma}_{u_{t_n}, u_{r_n}} = \frac{P_{u_{t_n}} |h_{u_{t_n}, u_{r_n}}|^2 d_{u_{t_n}, u_{r_n}}^{-\alpha}}{I_{D, u_{r_n}} + P_{s_0} |h_{s_0, u_{r_n}}|^2 d_{s_0, u_{r_n}}^{-\alpha} + \sigma_{u_{r_n}}^2}, \tag{16}$$

where  $P_{u_{t_n}}$  denotes the transmission power of TU  $u_{t_n}$ .  $h_{u_{t_n}, u_{r_n}}$  and  $d_{u_{t_n}, u_{r_n}}$  represent the small-scale fast fading and distance between TU  $u_{t_n}$  and its corresponding RU  $u_{r_n}$ , respectively.  $h_{u_{t_i}, u_{r_n}} \sim \mathcal{CN}(0, 1)$ .  $I_{D, u_{r_n}} = \sum_{u_{t_i} \in \Phi_{DU} \setminus \{u_{t_n}\}} P_{u_{t_i}} |h_{u_{t_i}, u_{r_n}}|^2 d_{u_{t_i}, u_{r_n}}^{-\alpha}$  is the total interference from other D2D pairs to RU  $u_{r_n}$ .  $P_{s_0} |h_{s_0, u_{r_n}}|^2 d_{s_0, u_{r_n}}^{-\alpha}$  is the interference from the CUE  $s_0$  to RU  $u_{r_n}$ .  $\sigma_{u_{r_n}}^2$  is the power of the additive Gaussian white noise at  $u_{r_n}$ .

**Lemma 2.** *Conditioning on the distance  $d_{u_{t_n}, u_{r_n}}$ , the coverage probability of RU  $u_{r_n}$  in the interference limited regime ( $\sigma_{u_{r_n}}^2 = 0$ ) is given by*

$$\begin{aligned}
& P_{dc}(\beta) \Big|_{d_{u_{t_n}, u_{r_n}}} \\
& \approx \mathbb{E}_{P_{u_{t_n}}} \left[ \exp\left(-B_1 P_{u_{t_n}}^{-2/\alpha}\right) \frac{1}{1 + B_2 P_{u_{t_n}}^{-2/\alpha}} \right], \tag{17}
\end{aligned}$$

where  $B_1 = (\pi\lambda_{DU}PS_{u_n}\beta^{2/\alpha}/\sin c(2/\alpha))\mathbb{E}[P_{u_n}^{2/\alpha}]d_{u_n,u_r}^2$ ,  $B_2 = (\beta P_{s_0})^{2/\alpha}((45\pi)^2 d_{u_n,u_r}^2/(128R)^2)$ , and  $\beta$  represents the minimum SINR value for reliable D2D connection.

*Proof.* The coverage probability of RU  $u_r$  can be expressed as [10, 12]

$$\begin{aligned} P_{dc}(\beta)|_{d_{u_n,u_r}} &= \mathbb{P}(\bar{\gamma}_{u_t,u_r} \geq \beta)|_{d_{u_n,u_r}} \\ &= \mathbb{P}\left(\frac{P_{u_n}|h_{u_n,u_r}|^2 d_{u_n,u_r}^{-\alpha}}{I_{D,u_r} + P_{s_0}|h_{s_0,u_r}|^2 d_{s_0,u_r}^{-\alpha}} \geq \beta\right) \\ &= \mathbb{P}\left(|h_{u_t,u_r}|^2\right) \end{aligned}$$

$$\begin{aligned} &\geq P_{u_n}^{-1} d_{u_n,u_r}^{\alpha} \beta \left( I_{D,u_r} + P_{s_0} |h_{s_0,u_r}|^2 d_{s_0,u_r}^{-\alpha} \right) \\ &= \mathbb{E}\left[ e^{-P_{u_n}^{-1} d_{u_n,u_r}^{\alpha} \beta (I_{D,u_r} + P_{s_0} |h_{s_0,u_r}|^2 d_{s_0,u_r}^{-\alpha})} \right]. \end{aligned} \quad (18)$$

We use Slivnyak's theorem [16] as follows:

$$\begin{aligned} L_{\Phi_{DU} \setminus \{u_n\}}(s) &= \mathbb{E}\left[ e^{-s \sum_{u_t \in \Phi_{DU} \setminus \{u_n\}} P_{u_t} |h_{u_t,u_r}|^2 d_{u_t,u_r}^{-\alpha}} \right] \\ &= L_{\Phi_{DU}}(s) = e^{-(\pi\lambda_{DU}/\sin c(2/\alpha))\mathbb{E}[P_{u_n}^{2/\alpha}]s^{2/\alpha}}. \end{aligned} \quad (19)$$

Conditioned on  $P_{u_n}$ , (18) can be written as

$$P_{dc}(\beta)|_{d_{u_n,u_r}} = e^{-(\pi\lambda_{DU}\beta^{2/\alpha}/\sin c(2/\alpha))\mathbb{E}[P_{u_n}^{2/\alpha}]d_{u_n,u_r}^2 P_{u_n}^{-2/\alpha}} \mathbb{E}\left[ e^{-P_{u_n}^{-1} d_{u_n,u_r}^{\alpha} \beta P_{s_0} |h_{s_0,u_r}|^2 d_{s_0,u_r}^{-\alpha}} \right], \quad (20)$$

where  $\mathbb{E}[e^{-P_{u_n}^{-1} d_{u_n,u_r}^{\alpha} \beta P_{s_0} |h_{s_0,u_r}|^2 d_{s_0,u_r}^{-\alpha}}] = \mathbb{E}_{d_{s_0,u_r}}[1/(1 + \beta P_{s_0} P_{u_n}^{-1} d_{u_n,u_r}^{\alpha} d_{s_0,u_r}^{-\alpha})]$ .

Furthermore, we use the approximation  $\mathbb{E}_{d_{s_0,u_r}}[1/(1 + k/d_{s_0,u_r}^{\alpha})] \approx [1/(1 + k^{2/\alpha}/\mathbb{E}[d_{s_0,u_r}^2])] in [10] and  $\mathbb{E}[d_{s_0,u_r}] = 128R/45\pi$  in [13]. Equation (20) can be expressed as in (17).$

In (17), we can see that the coverage probability of the D2D link is determined by two factors: (a) the total interference power created by other active D2D links (i.e.,  $B_1 P_{u_n}^{-2/\alpha}$ ) and (b) the approximated effect of the uplink interference from the CUE (i.e.,  $1/(1 + B_2 P_{u_n}^{-2/\alpha})$ ).

Unlike intuition, it is interesting to find that the coverage probability of the D2D link does not necessarily increase with the parameter  $I_{k_0}$  for any given  $d_{u_n,u_r}$ ,  $d_{u_n,k_0}$ , and  $|h_{u_n,k_0}|^2$ .

$$\begin{aligned} P_{dc}(\beta)|_{d_{u_n,u_r}, d_{u_n,k_0}, |h_{u_n,k_0}|^2} \\ \approx \exp\left(-\frac{\pi\lambda_{DU}PS_{u_n}\beta^{2/\alpha}\mathbb{E}[P_{u_n}^{2/\alpha}]d_{u_n,u_r}^2}{\sin c(2/\alpha)P_{u_n}^{2/\alpha}}\right) \\ \cdot \frac{1}{1 + (\beta(P_{s_0}/P_{u_n}))^{2/\alpha}((45\pi)^2 d_{u_n,u_r}^2/(128R)^2)}. \end{aligned} \quad (21)$$

According to (3), (13), and (21), there are four cases as follows:

- (1)  $(I_{k_0}/N_d)^{2/\alpha}\Gamma((-2 + \alpha)/\alpha)R^2 \leq P_{u_n,\max}^{2/\alpha}$  and  $P_{u_n,\max}^{2/\alpha} \leq (I_{k_0}/N_d)^{2/\alpha}(|h_{u_n,k_0}|^{-2})^{2/\alpha}d_{u_n,k_0}^2$ ,  $P_{dc}(\beta)$  decrease with  $I_{k_0}$ .
- (2)  $(I_{k_0}/N_d)^{2/\alpha}\Gamma((-2 + \alpha)/\alpha)R^2 \leq P_{u_n,\max}^{2/\alpha}$  and  $(I_{k_0}/N_d)^{2/\alpha}(|h_{u_n,k_0}|^{-2})^{2/\alpha}d_{u_n,k_0}^2 \leq P_{u_n,\max}^{2/\alpha}$ ,  $P_{dc}(\beta)$  increase with  $I_{k_0}$ .

(3)  $P_{u_n,\max}^{2/\alpha} \leq (I_{k_0}/N_d)^{2/\alpha}\Gamma((-2 + \alpha)/\alpha)R^2$  and  $P_{u_n,\max}^{2/\alpha} \leq (I_{k_0}/N_d)^{2/\alpha}(|h_{u_n,k_0}|^{-2})^{2/\alpha}d_{u_n,k_0}^2$ ,  $P_{dc}(\beta)$  donot change with  $I_{k_0}$ .

(4)  $P_{u_n,\max}^{2/\alpha} \leq (I_{k_0}/N_d)^{2/\alpha}\Gamma((-2 + \alpha)/\alpha)R^2$  and  $(I_{k_0}/N_d)^{2/\alpha}(|h_{u_n,k_0}|^{-2})^{2/\alpha}d_{u_n,k_0}^2 \leq P_{u_n,\max}^{2/\alpha}$ ,  $P_{dc}(\beta)$  increase with  $I_{k_0}$ .

For case (1), the channel gain from TU  $u_t$  to BS  $k_0$  is small; TU  $u_t$  can transmit with the maximum transmission power  $P_{u_n,\max}$ . However, for other TU pieces, the transmission powers are limited by  $I_{k_0}$ . When  $I_{k_0}$  increases, the transmission power of TU  $u_t$  does not change and the average transmission power of other TU pieces (i.e.,  $\mathbb{E}[P_{u_n}^{2/\alpha}]$ ) increases, which reduces the coverage probability of the D2D link ( $u_t, u_r$ ).  $\square$

## 5. Optimizing D2D On-Off Threshold

In this section, we assume that each TU has the same maximum transmission power (i.e.,  $P_{u_n,\max} = P_{u,\max}$ ). In the following, we first introduce the area spectral efficiency of D2D links and the cellular link. Second, the upper bound of the area spectral efficiency of D2D links and the lower bound of the area spectral efficiency of the cellular link are obtained, respectively. Third, we optimize the D2D on-off threshold by maximizing the approximated total area spectral efficiency.

Since the analysis performed for a typical link indicates the spatially averaged performance of the network by Slivnyak's theorem [16, 17], for any given  $d_{u_n,u_r}$ ,  $d_{u_n,k_0}$ , and  $|h_{u_n,k_0}|^2$ , the area spectral efficiency of D2D communications is given as [12, 18]

$$\begin{aligned}
Y^D(\beta) &= \lambda_{DU} PS_{u_n} \log_2(1 + \beta) \mathbb{P}(\bar{Y}_{u_n, u_{rn}} \geq \beta) \\
&\approx \lambda_{DU} PS_{u_n} \\
&\cdot \exp\left(-\frac{\pi \lambda_{DU} PS_{u_n} \beta^{2/\alpha} \mathbb{E}[P_{u_n}^{2/\alpha}]}{\sin c(2/\alpha) P_{u_n}^{2/\alpha}} d_{u_n, u_{rn}}^2\right) \\
&\cdot \frac{\log_2(1 + \beta)}{1 + (\beta(P_{s_0}/P_{u_n}))^{2/\alpha} ((45\pi)^2 d_{u_n, u_{rn}}^2 / (128R^2))}.
\end{aligned} \quad (22)$$

The area spectral efficiency of the cellular link is as follows [10]:

$$Y^C(\eta) = \frac{\log 2(1 + \eta) P_{cc}(\eta)}{(\pi R^2)}. \quad (23)$$

According to (22) and (23), the optimal access probability for D2D links can be obtained by solving the following optimization problem:

$$\begin{aligned}
&Y_{upper}^D(\beta) \\
&= \begin{cases} \lambda_{DU} PS_{u_n} D_2 \exp(-D_1), & \text{for } \left(\frac{P_{s_0} d_{s_0, k_0}^{-\alpha}}{N_d \eta}\right)^{2/\alpha} \Gamma\left(\frac{-2 + \alpha}{\alpha}\right) \Gamma\left(\frac{2 + \alpha}{\alpha}\right) R^2 \geq P_{u, max}^{2/\alpha} \\ \lambda_{DU} PS_{u_n} D_2 \exp\left(-\frac{D_1 (P_{s_0} d_{s_0, k_0}^{-\alpha} / N_d \eta)^{2/\alpha} \Gamma((-2 + \alpha)/\alpha) \Gamma((2 + \alpha)/\alpha) R^2}{P_{u, max}^{2/\alpha}}\right), & \text{for } \left(\frac{P_{s_0} d_{s_0, k_0}^{-\alpha}}{N_d \eta}\right)^{2/\alpha} \Gamma\left(\frac{-2 + \alpha}{\alpha}\right) \Gamma\left(\frac{2 + \alpha}{\alpha}\right) R^2 < P_{u, max}^{2/\alpha}, \end{cases} \quad (26)
\end{aligned}$$

where  $D_1 = (\pi \lambda_{DU} PS_{u_n} \beta^{2/\alpha} / \sin c(2/\alpha)) d_{u_n, u_{rn}}^2$ ,  $D_2 = \log_2(1 + \beta) / (1 + (\beta(P_{s_0}/P_{u, max}))^{2/\alpha} ((45\pi)^2 d_{u_n, u_{rn}}^2 / (128R^2)))$ .

*Proof.* See Appendix A.  $\square$

Conditioning on the distance  $d_{s_0, k_0}$ , we also compute a lower bound of the coverage probability of the cellular link by setting  $\mathbb{E}[P_{u_n}^{2/\alpha}] = P_{u, max}^{2/\alpha}$  in the interference limited regime ( $\sigma_{k_0}^2 = 0$ ), which can be expressed as

$$P_{cc, low}(\eta) = \exp\left(-\frac{\pi \lambda_{DU} PS_{u_n} \eta^{2/\alpha} P_{u, max}^{2/\alpha}}{\sin c(2/\alpha) P_{s_0}^{2/\alpha}} d_{s_0, k_0}^2\right). \quad (27)$$

The lower bound of the area spectral efficiency of the cellular link is given by

$$Y_{low}^C(\eta) = \frac{\log 2(1 + \eta) P_{cc, low}(\eta)}{(\pi R^2)}. \quad (28)$$

According to (25) and (28), original problem (24) can be approximated as follows:

$$\begin{aligned}
&\max_{PS_{u_n}}: \omega_1 Y^C(\eta) \\
&\max_{PS_{u_n}}: \omega_2 Y^D(\beta) \\
&\text{s.t. } 0 < PS_{u_n} < 1,
\end{aligned} \quad (24)$$

where  $\omega_1$  and  $\omega_2$  are the introduced scalar weights, which represent the priorities of CUE and DUE pieces, respectively. Note that  $Y^{\text{total}}(\eta, \beta) = \omega_1 Y^C(\eta) + \omega_2 Y^D(\beta)$  can represent the total area spectral efficiency including both the cellular link and D2D links. Note that it is difficult to obtain the closed-form solution of the original optimization problem. Therefore, in order to facilitate the analysis and reduce the computational complexity, we introduce the upper bound of the area spectral efficiency of D2D links and the lower bound of the area spectral efficiency of the cellular link.

**Lemma 3.** *The upper bound of the area spectral efficiency of D2D communications is obtained as*

$$Y^D(\beta) \leq Y_{upper}^D(\beta) \quad (25)$$

with

$$\begin{aligned}
&\max_{PS_{u_n}}: \omega_1 Y_{low}^C(\eta) \\
&\max_{PS_{u_n}}: \omega_2 Y_{upper}^D(\beta) \\
&\text{s.t. } 0 < PS_{u_n} < 1
\end{aligned} \quad (29)$$

which is a multicriteria optimization problem. According to (25) and (28),  $Y_{low}^C(\eta)$  decreases with  $PS_{u_n}$ , while  $Y_{upper}^D(\beta)$  is nonmonotonic with  $PS_{u_n}$ . Generally, it is difficult to maximize the two objectives simultaneously. In order to balance these two objectives, the weighted sum method in multicriteria optimization is used to obtain the Pareto optimal solution [19]. Then, the multicriteria optimization with high complexity can be converted into a simpler single objective optimization problem given by

$$\begin{aligned}
&\max_{PS_{u_n}}: \omega_1 Y_{low}^C(\eta) + \omega_2 Y_{upper}^D(\beta) \\
&\text{s.t. } 0 < PS_{u_n} < 1.
\end{aligned} \quad (30)$$

The solution of (30) is a Pareto optimal solution to the optimization problem of (29) [19].

**Theorem 4.** For the optimization problem

$$\begin{aligned} \max: \quad & f(x) = \varepsilon_1 e^{-ax} + \varepsilon_2 x e^{-bx} \\ \text{s.t.} \quad & 0 < x < 1 \end{aligned} \quad (31)$$

with  $\varepsilon_1 \geq 0$ ,  $\varepsilon_2 \geq 0$ ,  $a > 0$ , and  $b > 0$ , the optimal solution  $x^*$  is obtained as

$$x_{opt} = \max \{ \min \{ \hat{x}, 1 \}, 0 \} \quad (32)$$

with

$$\hat{x} = \frac{1}{b} + \frac{1}{q} \omega \left( -\frac{pqe^{-q/b}}{b} \right), \quad (33)$$

where  $p = a\varepsilon_1/\varepsilon_2$ ,  $q = a - b$ , and  $\omega(x)$  denotes the Lambert  $\omega$  function defined as the inverse function of  $g(x) = xe^x$ .

*Proof.* See Appendix B.  $\square$

According to Theorem 4, the optimal access threshold can be obtained as

$$PS_{u_n}^* = \max \left\{ \min \left\{ \hat{PS}_{u_n}, 1 \right\}, 0 \right\} \quad (34)$$

with

$$\hat{PS}_{u_n} = \frac{1}{\rho_2} + \frac{1}{\phi_2} \omega \left( -\frac{\phi_1 \phi_2 e^{-\phi_2/\rho_2}}{\rho_2} \right), \quad (35)$$

where

$$\begin{aligned} \phi_1 &= \frac{\rho_1 \varepsilon_1}{\varepsilon_2}, \\ \phi_2 &= \rho_1 - \rho_2, \\ \varepsilon_1 &= \omega_1 \frac{\log_2(1 + \eta)}{\pi R^2}, \\ \rho_1 &= \frac{\pi \lambda_{DU} \eta^{2/\alpha}}{\sin c(2/\alpha)} \left( \frac{P_{u,\max}}{P_{s_0}} \right)^{2/\alpha} d_{s_0, k_0}^2, \\ \varepsilon_2 &= \frac{\omega_2 \lambda_{DU} \log_2(1 + \beta)}{1 + (\beta (P_{s_0}/P_{u,\max}))^{2/\alpha} ((45\pi)^2 d_{u_n, u_n}^2 / (128R)^2)}, \\ \rho_2 &= \begin{cases} \frac{\pi \lambda_{DU} \beta^{2/\alpha}}{\sin c(2/\alpha)} d_{u_n, u_n}^2, & \text{for } \left( \frac{P_{s_0} d_{s_0, k_0}^{-\alpha}}{N_d \eta} \right)^{2/\alpha} \Gamma \left( \frac{-2 + \alpha}{\alpha} \right) \Gamma \left( \frac{2 + \alpha}{\alpha} \right) R^2 \geq P_{u,\max}^{2/\alpha} \\ \frac{\pi \lambda_{DU} \beta^{2/\alpha} (P_{s_0} d_{s_0, k_0}^{-\alpha} / N_d \eta)^{2/\alpha} \Gamma((-2 + \alpha)/\alpha) \Gamma((2 + \alpha)/\alpha) R^2}{P_{u,\max}^{2/\alpha}} d_{u_n, u_n}^2, & \text{for } \left( \frac{P_{s_0} d_{s_0, k_0}^{-\alpha}}{N_d \eta} \right)^{2/\alpha} \Gamma \left( \frac{-2 + \alpha}{\alpha} \right) \Gamma \left( \frac{2 + \alpha}{\alpha} \right) R^2 < P_{u,\max}^{2/\alpha}. \end{cases} \end{aligned} \quad (36)$$

Then, the optimal access threshold is designed according to the optimal access probability. According to (1) and (34), the optimal access threshold for the D2D pair  $u_n$  is given as

$$G_{\min, u_n}^* = \frac{-\ln(PS_{u_n}^*)}{d_{u_n, u_n}^\alpha}. \quad (37)$$

One D2D pair decides whether to access the current subchannel based solely on the knowledge of its own communication distance, channel gain, and the optimal access threshold. In order to calculate the access threshold, each TU needs some system parameters (e.g.,  $\lambda_{DU}$ ,  $P_{s_0}$ , and  $d_{s_0, k_0}$ ). Compared with DUE pieces, the BS is more convenient for obtaining these parameters and broadcasting them to each TU. Then, each D2D pair can calculate its own optimal access threshold according to (34) and (37) based on the system parameters and  $d_{u_n, u_n}$  independently.  $d_{u_n, u_n}$  can be easily

known by the associated TU  $u_n$ . Since (34) and (37) have the closed-forms, the computational burden is quite low for each TU.

Note that the parameter  $\rho_2$  in (35) has two possible expressions due to the relationship between  $P_{u,\max}^{2/\alpha}$  and  $(P_{s_0} d_{s_0, k_0}^{-\alpha} / N_d \eta)^{2/\alpha} \Gamma((-2 + \alpha)/\alpha) \Gamma((2 + \alpha)/\alpha) R^2$ . However, the number of active TU pieces (i.e.,  $N_d$ ) cannot be known before all the D2D pairs decide whether to access the current subchannel. In this paper, we average over the past subchannels as in Algorithm 1 and apply  $\widehat{N}_{d,i}$  as the estimation of  $N_{d,i}$  that is used to calculate  $\rho_2$  in (35).

Though  $\widehat{N}_{d,i}$  is not the real number of active TU pieces on the subchannel  $i$ , the optimal access threshold based on  $\widehat{N}_{d,i}$  can achieve a good performance, which will be verified by the extensive simulations in the following. This is because  $\widehat{N}_{d,i}$  represents the statistical average of  $N_{d,i}$  and  $\widehat{N}_{d,i}$  is close to  $N_{d,i}$  with high D2D link density  $\lambda_{DU}$ .

- (1) Initial set:  $\widehat{N}_{d,1} = N_d^{\text{tol}}$
- (2) For  $i = 2 : M$
- (3)  $\widehat{N}_{d,i} = (\widehat{N}_{d,1} + \dots + \widehat{N}_{d,i-1}) / (i - 1)$  is used to calculate the optimal access as in Eq. (25) and Eq. (34).
- (4) Count the real number of active  $N_{d,i}$  on  $i$ th sub-channel.
- (5)  $\widehat{N}_{d,i} = N_{d,i}$
- (6) end

ALGORITHM 1: Estimating the number of active TUs for optimal access threshold.

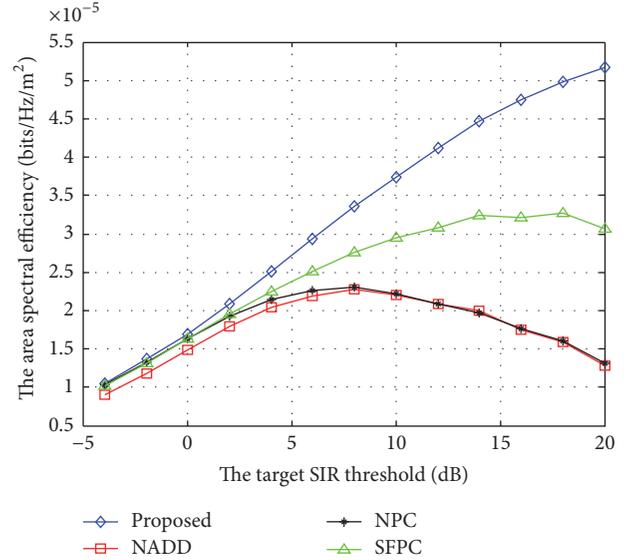
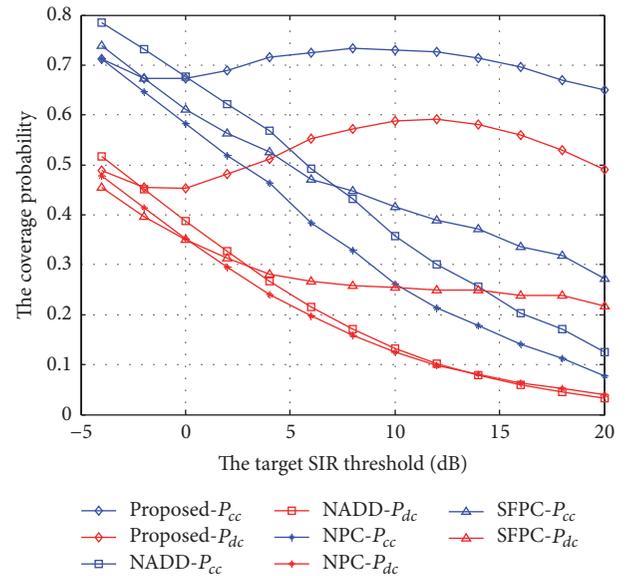
## 6. Simulation Results

To evaluate the proposed scheme, we carry out extensive simulations by the Matlab tool and discuss the results in this section. The path-loss exponent  $\alpha = 4$ . We assume the average radius of a cell is 500 m; that is,  $R = 500$  m. The BS is located at the center and the CUE is uniformly located in the cell of radius  $R$ . The transmit power of CUE  $P_{s_0} = 100$  mW. Given the D2D communication range  $R_d = 50$  m, each intended RU is uniformly and independently located within  $R_d$  of its associated TU in the isotropic direction. The maximum transmit power of TU pieces  $P_{u_n, \max} = P_{u, \max} = 0.1$  mW. The scalar weight factor for the DUE is  $\omega_2 = 1$ . The performance of the proposed method is compared with three other strategies:

- (1) NADD: network-assisted, device-decided scheme [11]
- (2) SFPC: statistical features-based power control scheme [12]
- (3) NPC: all D2D pairs access the current subchannel. There is no power control and all TU pieces transmit with maximum transmit power  $P_{u, \max}$
- (4) Proposed: the proposed method.

In Figure 4, the area spectral efficiency performances versus target SIR threshold according to different power control methods are shown. We can see that the proposed method improves the area spectral efficiency compared to other algorithms, especially in the high target SIR regime. This implies that the proposed method is efficient at mitigating both intra-D2D and cross-tier interference when D2D links communicate with a high data rate. Moreover, it is interesting to see that the NADD method is even worse than NPC in terms of the area spectral efficiency. This is because though the NADD method guarantees the performance of the CUE, the NADD method leads to a sharp performance fall for D2D links, which results in the area spectral efficiency decrease in the end.

Figure 5 shows the coverage probabilities of CUEs and DUE pieces versus target SIR threshold according to different power control methods. In Figure 5, it can be observed that the proposed method achieves the best performance for both the CUE and DUE pieces in terms of the coverage probability in the high target SIR regime. Although the proposed method does not have the highest coverage probabilities for the CUE and DUE pieces in the low target SIR regime (e.g., the target


 FIGURE 4: The area spectral efficiency  $\Upsilon^{\text{total}}(\beta)$  versus target SIR threshold according to different strategies with  $\eta = \beta$ ,  $\lambda_{\text{DU}} = 3 \times 10^{-5}$ , and  $\omega_1 = 10$ .

 FIGURE 5: Coverage probability performance of both the CUE and the DUE according to different strategies with  $\eta = \beta$ ,  $\lambda_{\text{DU}} = 3 \times 10^{-5}$ , and  $\omega_1 = 10$ .

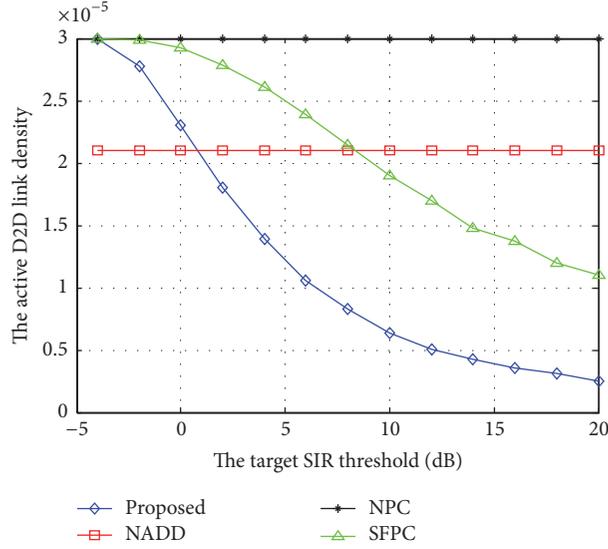


FIGURE 6: The active D2D link density versus target SIR threshold according to different strategies with  $\eta = \beta$ ,  $\lambda_{DU} = 3 \times 10^{-5}$ , and  $\omega_1 = 10$ .

SIR = -4 dB), it can achieve the best area spectral efficiency as shown in Figure 4. This is because there are more active DUE pieces for the proposed scheme as shown in Figure 6.

In Figure 6, the active D2D link density versus target SIR threshold is shown. We can observe that the active D2D link densities of the proposed method and the SFPC decrease with the target SIR threshold, and the active D2D link densities of the NADD and NPC do not change with the target SIR threshold. On the one hand, compared with the NADD and NPC, the proposed method and the SFPC can limit the active D2D link density to guarantee the area spectral efficiency performance. On the other hand, though the active D2D link density of the proposed method is lower than the other methods in the high target SIR regime, the proposed method can achieve the best area spectral efficiency performance. This is because the proposed method can achieve the best performance for both the CUE and DUE pieces in terms of the coverage probability in the high target SIR regime.

Figure 7 plots the area spectral efficiency versus D2D link density  $\lambda_{DU}$ . The proposed method can improve the area spectral efficiency significantly compared to other schemes. With the increase of  $\lambda_{DU}$ , for NADD and NPC, the area spectral efficiency decreases. This is because the number of active DUE pieces increases with  $\lambda_{DU}$  and the intra-D2D and cross-tier interferences become more serious. For the proposed scheme, the area spectral efficiency does not decrease with  $\lambda_{DU}$  according to the optimal threshold that balances the performances of the CUE and DUE pieces.

In Figures 8 and 9, the numbers in the x-axis represent different schemes as follows: (1) the proposed method; (2) NADD; (3) NPC; (4) SFPC. Figures 8 and 9 show  $\omega_1 \Upsilon^C(\beta)$  and  $\omega_2 \Upsilon^D(\beta)$  according to  $\omega_1 = 5$  and  $\omega_1 = 15$ , respectively. Comparing Figure 8 with Figure 9, we can see that the percentage of  $\omega_1 \Upsilon^C(\beta)$  for the proposed scheme increases

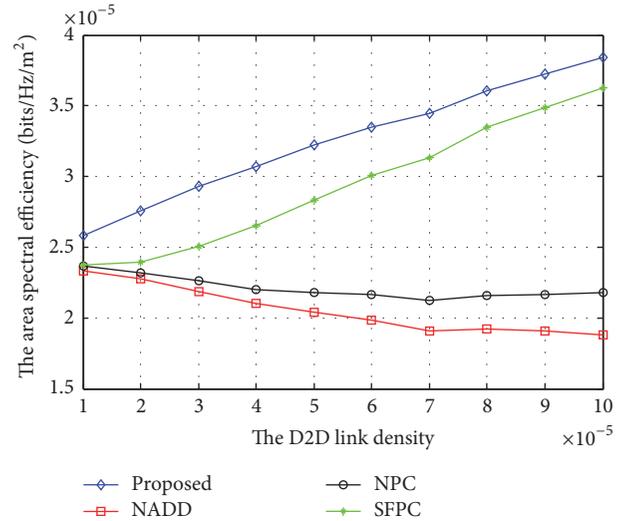


FIGURE 7: The area spectral efficiency  $\Upsilon^{\text{total}}(\beta)$  versus D2D link density  $\lambda_{DU}$  according to different strategies with  $\eta = \beta = 6$  dB and  $\omega_1 = 10$ .

with  $\omega_1$ . This is because the priority of the CUE increases with  $\omega_1$  and the larger percentage of  $\omega_1 \Upsilon^C(\beta)$  leads to a larger total area spectral efficiency. While the percentage of  $\omega_1 \Upsilon^C(\beta)$  for the other schemes also increases, the improved range of the proposed scheme is the largest, leading to greatest performance improvement. Moreover, the performance of NADD (case (2)) is worse than NPC (case (3)) for  $\omega_1 = 5$ , while it is opposite for  $\omega_1 = 15$ . This is because NADD can guarantee the performance of the CUE. When the priority of the CUE increases with  $\omega_1$ , the performance of NADD becomes better than NPC in terms of total area spectral efficiency.

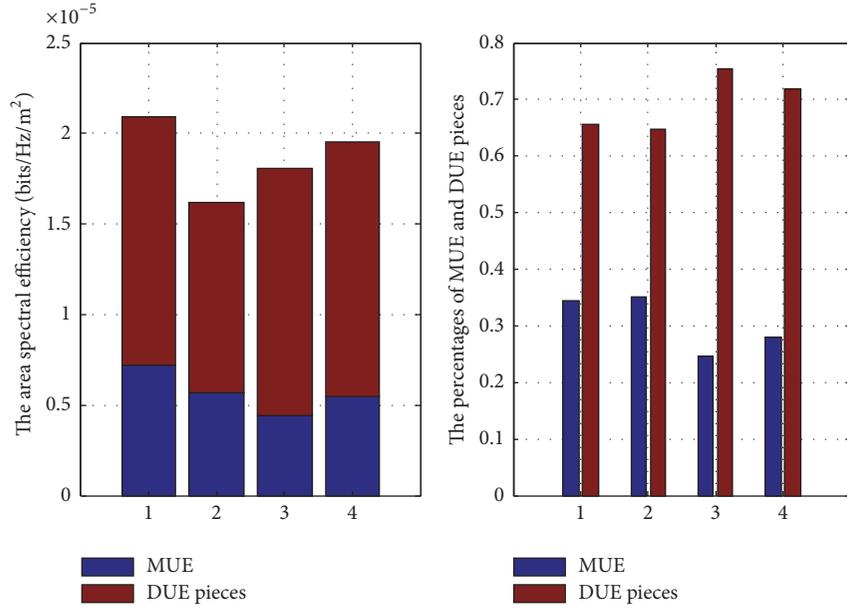


FIGURE 8: The figure shows  $\omega_1 \Upsilon^C(\beta)$  and  $\omega_2 \Upsilon^D(\beta)$  with  $\omega_1 = 5$ ,  $\eta = \beta = 6$  dB, and  $\lambda_{DU} = 3 \times 10^{-5}$ , respectively.

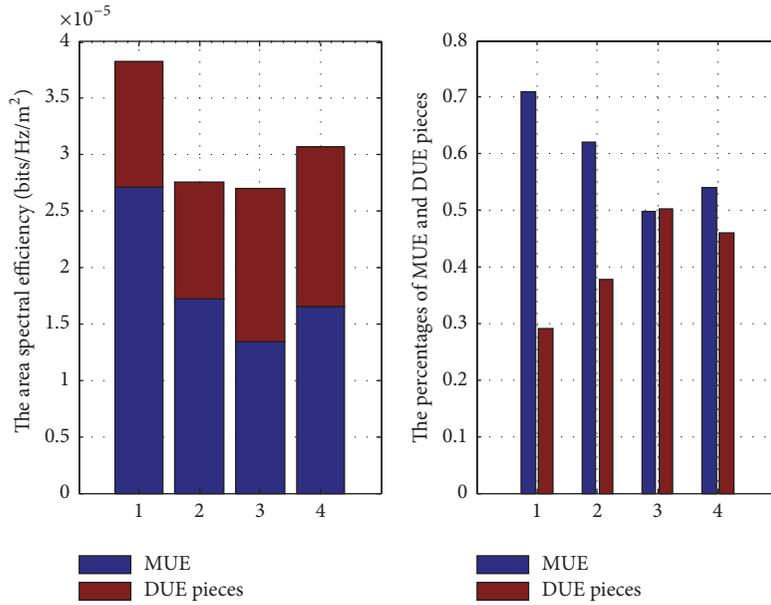


FIGURE 9: The figure shows  $\omega_1 \Upsilon^C(\beta)$  and  $\omega_2 \Upsilon^D(\beta)$  with  $\omega_1 = 15$ ,  $\eta = \beta = 6$  dB, and  $\lambda_{DU} = 3 \times 10^{-5}$ , respectively.

### 7. Conclusions

The D2D cellular network is an efficient way to carry out mobile crowdsourcing. For D2D underlaid cellular networks, the channel access and power control issue is investigated. In order to limit the cross-tier and intratier interferences, we propose a novel semidistributed network-assisted power and channel access control scheme for DUE pieces. In the proposed semidistributed scheme, with the help of assistant information from the BS, each D2D pair decides whether

to access the subchannel and adjusts the transmit power independently. The BS plays a role in offering assistant information, which can maximize the total area spectral efficiency on the condition of guaranteeing the performance of the cellular link. Moreover, assistant information from the BS is statistical, and the optimal access threshold has the closed-form expression. The proposed scheme has the low computational burden and low overhead to broadcast assistant information, which is quite suitable to the practical systems.

## Appendix

### A. The proof of Lemma 3

By setting  $P_{u_n} = P_{u_{\max}}$ , we have

$$\begin{aligned} \Upsilon^D(\beta) &\leq \lambda_{\text{DU}} P S_{u_n} \\ &\cdot \exp\left(-\frac{\pi \lambda_{\text{DU}} P S_{u_n} \beta^{2/\alpha} \mathbb{E}\left[\frac{P_{u_n}^{2/\alpha}}{P_{u_{\max}}^{2/\alpha}}\right] d_{u_n, u_{r_n}}^2}{\sin c(2/\alpha)}\right) \\ &\cdot \frac{\log_2(1+\beta)}{1 + \left(\beta(P_{s_0}/P_{u_{\max}})\right)^{2/\alpha} \left((45\pi)^2 d_{u_n, u_{r_n}}^2 / (128R)^2\right)}. \end{aligned} \quad (\text{A.1})$$

Plugging (15) into (A.1), we have (25).

### B. The proof of Theorem 4

The derivative of  $f(x)$  with respect to  $x$  is written as

$$\frac{\partial f(x)}{\partial x} = -a\omega_1 e^{-ax} + \omega_2 e^{-bx} (1 - bx). \quad (\text{B.1})$$

From the first-order optimality condition, that is,  $\partial f(x)/\partial x = 0$ , we have

$$1 - bx = \frac{a\varepsilon_1}{\varepsilon_2} e^{(b-a)x}. \quad (\text{B.2})$$

Denoting that  $p = a\varepsilon_1/\varepsilon_2$  and  $q = a - b$ , (B.2) is rewritten as

$$1 - bx = p e^{-qx}. \quad (\text{B.3})$$

Then, a closed-form solution of  $x$  which satisfies the first-order optimality condition is derived as

$$\hat{x} = \frac{1}{b} + \frac{1}{q} \omega\left(-\frac{pqe^{-q/b}}{b}\right). \quad (\text{B.4})$$

Considering the constraint, that is,  $0 < x < 1$ , we have Theorem 4.

### Conflicts of Interest

The authors declare that they have no conflicts of interest.

### Acknowledgments

The research was supported in part by the National Natural Science Foundation of China (Grant nos. 61601482 and 61502518).

### References

- [1] Z. Ning, X. Hu, Z. Chen et al., "A cooperative quality-aware service access system for social internet of vehicles," *IEEE Internet of Things Journal*, pp. 1-1.
- [2] Y. Han, T. Luo, D. Li, and H. Wu, "Competition-based participant recruitment for delay-sensitive crowdsourcing applications in D2D networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 12, pp. 2987-2999, 2016.
- [3] M. Tehrani, M. Uysal, and H. Yanikomeroglu, "Device-to-device communication in 5G cellular networks: challenges, solutions, and future directions," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 86-92, 2014.
- [4] X. Hu, J. Deng, J. Zhao et al. et al., "SAFeDJ: a crowd-cloud co-design approach to situation-aware music delivery for drivers," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 12, no. 1s, article 21, 2015.
- [5] X. Hu, T. H. S. Chu, H. C. B. Chan, and V. C. M. Leung, "Vita: a crowdsensing-oriented mobile cyber-physical system," *IEEE Transactions on Emerging Topics in Computing*, vol. 1, no. 1, pp. 148-165, 2013.
- [6] X. Hu, T. H. S. Chu, V. C. M. Leung, E. C.-H. Ngai, P. Kruchten, and H. C. B. Chan, "A survey on mobile social networks: applications, platforms, system architectures, and future research directions," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1557-1581, 2015.
- [7] L. Lei, Y. Kuang, X. Shen, C. Lin, and Z. Zhong, "Resource control in network assisted device-to-device communications: solutions and challenges," *IEEE Communications Magazine*, vol. 52, no. 6, pp. 108-117, 2014.
- [8] C. Yu, K. Doppler, C. B. Ribeiro, and O. Tirkkonen, "Resource sharing optimization for device-to-device communication underlying cellular networks," *IEEE Transactions on Wireless Communications*, vol. 10, no. 8, pp. 2752-2763, 2011.
- [9] J. Gu, S. J. Bae, B.-G. Choi, and M. Y. Chung, "Dynamic power control mechanism for interference coordination of device-to-device communication in cellular networks," in *Proceedings of the 3rd International Conference on Ubiquitous and Future Networks (ICUFN '11)*, pp. 71-75, June 2011.
- [10] N. Y. Lee, X. Q. Lin, J. G. Andrews, and R. W. Heath, "Power control for D2D underlaid cellular networks: modeling, algorithms, and analysis," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 1, pp. 1-13, 2015.
- [11] S.-H. Yang, L.-C. Wang, J.-H. Huang, and A.-H. Tsai, "Network-assisted device-decided channel selection and power control for multi-pair device-to-device (D2D) communications in heterogeneous networks," in *Proceedings of the 2014 IEEE Wireless Communications and Networking Conference, WCNC 2014*, pp. 1356-1361, Istanbul, Turkey, April 2014.
- [12] P. Sun, K. G. Shin, H. Zhang, and L. He, "Transmit power control for d2d-underlaid cellular networks based on statistical features," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 4110-4119, 2017.
- [13] D. Moltchanov, "Distance distributions in random networks," *Ad Hoc Networks*, vol. 10, no. 6, pp. 1146-1166, 2012.
- [14] L. Zhou, X. Hu, E. C.-H. Ngai et al., "A dynamic graph-based scheduling and interference coordination approach in heterogeneous cellular networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 5, pp. 3735-3748, 2016.
- [15] X. Lin, J. G. Andrews, and A. Ghosh, "Spectrum sharing for device-to-device communication in cellular networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 12, pp. 6727-6740, 2014.
- [16] D. Stoyan, W. Kendall, and J. Mecke, *Stochastic Geometry and Its Applications*, Wiley, New York, NY, USA, 1995.
- [17] A. Behnad, A. M. Rabiei, and N. C. Beaulieu, "Performance analysis of opportunistic relaying in a poisson field of amplify-and-forward relays," *IEEE Transactions on Communications*, vol. 61, no. 1, pp. 97-107, 2013.
- [18] Z. Chen and M. Kountouris, "Distributed SIR-aware opportunistic access control for D2D underlaid cellular networks,"

in *Proceedings of the 2014 IEEE Global Communications Conference, GLOBECOM 2014*, pp. 1540–1545, Austin, TX, USA, December 2014.

- [19] M. Ehrgott, *Multicriteria Optimization*, Springer, New York, NY, USA, 2005.

## Research Article

# CPSFS: A Credible Personalized Spam Filtering Scheme by Crowdsourcing

Xin Liu,<sup>1</sup> Pingjun Zou,<sup>1</sup> Weishan Zhang,<sup>1</sup> Jiehan Zhou,<sup>2</sup> Changying Dai,<sup>1</sup>  
Feng Wang,<sup>1</sup> and Xiaomiao Zhang<sup>1</sup>

<sup>1</sup>College of Computer & Communication Engineering China University of Petroleum (East China), Qingdao, China

<sup>2</sup>University of Oulu, Oulu, Finland

Correspondence should be addressed to Xin Liu; lx@upc.edu.cn

Received 12 September 2017; Accepted 5 December 2017; Published 27 December 2017

Academic Editor: Kuan Zhang

Copyright © 2017 Xin 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.

Email spam consumes a lot of network resources and threatens many systems because of its unwanted or malicious content. Most existing spam filters only target complete-spam but ignore semispam. This paper proposes a novel and comprehensive CPSFS scheme: Credible Personalized Spam Filtering Scheme, which classifies spam into two categories: complete-spam and semispam, and targets filtering both kinds of spam. Complete-spam is always spam for all users; semispam is an email identified as spam by some users and as regular email by other users. Most existing spam filters target complete-spam but ignore semispam. In CPSFS, Bayesian filtering is deployed at email servers to identify complete-spam, while semispam is identified at client side by crowdsourcing. An email user client can distinguish junk from legitimate emails according to spam reports from credible contacts with the similar interests. Social trust and interest similarity between users and their contacts are calculated so that spam reports are more accurately targeted to similar users. The experimental results show that the proposed CPSFS can improve the accuracy rate of distinguishing spam from legitimate emails compared with that of Bayesian filter alone.

## 1. Introduction

Email is an essential communication method in the Internet age. However, the abuse of bulk emails allows spam to spread like a plague. Spam consumes network bandwidth and brings also other threats to recipients: unwanted advertisements and pornographic content, as well as malicious viruses [1]. A spammer does not need to get permissions from recipients when sending spam, which causes serious annoyance to people and even leads to information security risks [2]. If a recipient clicks a malicious link in the spam message, their personal information may be automatically sent to the spammer via a malicious program, which is an obvious challenge for privacy protection [3, 4]. Statistics showed that spam accounted for 81.8% of total emails in 2016, compared with 72.9% in 2015 [5], which is obviously an increasing threat to email users.

To tackle this issue, in this paper, we classify spam into two categories according to the scope of affected users. One kind of spam is “complete-spam,” which is defined as email

identified by all users as spam. The other kind of spam is semispam, which is identified as spam by some users but as legitimate by other users. Most spam filters were developed to identify complete-spam [6, 7]. The accuracy of spam detection of some of these filters can be fairly high [8]. However, very often we still find spam in our email inbox. This is because the existing spam filters can only identify complete-spam, but not semispam.

To resolve this issue, we need a comprehensive and personalized filtering mechanism that can utilize user contacts to collaboratively identify semispam, which is a novel scheme called CPSFS—Credible Personalized Spam Filtering Scheme. In CPSFS, a user can make use of social networks to obtain spam reports from his or her contacts, which can then be used to filter both complete-spam and semispam. This is an approach using crowdsourcing from involved email users, where spam reports from credible similar contacts help to boost the performance of collaborative spam filtering.

But how to choose credible contacts for a user? Users on the Internet are interconnected to form social networks

```

Input:
  L: local spam list
  M: an email set from inbox
  T: local trust-similarity list
Output:
  L: updated spam list
(1) Let  $M_i$  is the  $i$ th email of  $M$ ,  $L_j$  is the  $j$ th item of  $L$ ,  $T_k$  is  $k$ th contact's trust,  $S_k$ 
    is  $k$ th contact's similarity,  $T_{M_i}$  is the trust of the sender of  $M_i$ ,  $n$  is the number of
    emails,  $m$  is the number of items in  $L$ ,  $c$  is the number of contacts,  $S_r$  is the
    generated spam report,  $T_h$ : trust threshold,  $S_h$ : interest similarity threshold, flag is
    the subject of email
(2) for  $i \leftarrow 1$  to  $n$  do
(4)   if flag = "spam report"
      if  $T_{M_i} \geq T_h$ 
         $L_{m+1} \leftarrow$  content of the email
         $m \leftarrow m + 1$ 
        for  $k \leftarrow 1$  to  $c$  do
(8)           if  $S_k \geq S_h$ 
              forwarding  $M_i$  to contact  $k$ 
            end if
(10)        end for
      end if
    else
       $MD_5 \leftarrow$  hash( $S_i$ )// Calculating MD5 hash for  $S_i$ 
      for  $j \leftarrow 1$  to  $m$  do
        if  $MD_5 = L_j$ 
          put  $S_i$  into junk box
        end if
      end for
       $L_j \leftarrow MD_5$ 
       $m \leftarrow m + 1$ 
(6)     generating the spam report  $S_r$  of  $S_i$ 
(7)     for  $k \leftarrow 1$  to  $c$  do
(8)       if  $S_k \geq S_h$ 
          sending  $S_r$  to contact  $k$ 
        end if
(10)    end for
(11) end for
(13) return  $L$ 

```

ALGORITHM 1: Local spam filtering algorithm.

according to their relationships [9–11]. Social trust is a key factor that affects the sharing of knowledge and the development of social relationships [12, 13]: users are more likely to accept suggestions from others with high trust value and interests similarity [14]. Social trust can be calculated by analyzing Social Computing [15, 16].

Our contributions in this paper are as follows.

(1) We classify spam into two categories, complete-spam and semispam. We design different methods for filtering these two kinds of spam at both email servers and clients.

(2) We propose CPSFS which uses a crowdsourcing mechanism to filter semispam, where users with similar opinions collaborate together against semispam by sharing spam reports with each other, and social trust is used for users to choose credible contacts in order to avoid malicious users exploiting our scheme to propagate spam.

This paper is organized as follows. Section 2 discusses related work. Section 3 explains how the CPSFS is deployed.

Section 4 describes how to calculate trust value and interest similarity. The underlying local filter algorithm is presented in Algorithm 1. Next, Section 6 evaluates our approach via the experimental validation and compares with other filters. Finally, Section 7 concludes our work.

## 2. Related Work

Most previous works tried to filter out complete-spams for all users. We divided the existing work into four types based on the used techniques: the Black/White List, Bayesian, Machine Learning, and Social Computing.

*2.1. Black/White List.* Jaeyeon and Emil [17] presented a black/white list approach that relies on the number of IP addresses to determine whether an email is spam. The black list includes an email server and an IP address of sender. If the source of email appears in the black list, the email is

identified as spam. The problem with a black/white list is that it is difficult to update and maintain the list.

*2.2. Bayesian Approach.* O'Brien and Vogel [18] applied the Bayesian algorithm for spam filtering. The Bayesian filter parsed emails into keywords and then computed probabilities of keywords that appeared in spam and legitimate emails, respectively. The results showed that the Bayesian filter detected 91.7% of spam. This is a relatively high recognition rate for all emails using uniform criteria without considering semispam emails while calculating recognition rate.

*2.3. Machine Learning Approach.* Haider et al. [19] presented a machine learning-based approach by detecting batches of emails to filter spam effectively. The filter needs to be trained to distinguish keywords in spam. Scholkopf and Platt [20] presented a method that minimizes a loss function with respect to user's personal distribution based on the available biased samples. However, it is difficult to make the sample data have the same Dirichlet distribution.

*2.4. Social Computing Approach.* Zisiadis et al. [21] presented a collaborative method for email filtering called Mailbook which was based on a social network. Each node could mark the received spam and stored it in its own database. If one node marked an email as spam, the votes of spam increased by 1. Once the votes of spam reached a certain number, the system would mark the email as spam automatically. Similarly, Boykin and Roychowdhury [22] proposed a spam filtering approach based on social networks, which allows users to share the spam information with their friends to identify spam.

Sirivianos et al. [23] applied social network and trust mechanism for spam filtering. A node in a social network may report a spammer's IP address to a centralized server for the spam it received. The centralized server calculated the trust value according to the degree of confidence and credibility of the spam reporter; then it decided whether the IP address is a spammer. This method required a central server, which increased additional network overhead, and its accuracy was reduced by dynamic IP addressing. Shen and Li [24] presented a social network-aided spam filter which is used to improve the accuracy of spam filtering by integrating four new components into a Bayesian filter; these components identify spam by the closeness of nodes. Each node needs to collect information and check spam by its social network-aided spam filter, which will increase the overhead of the system.

Apparently, if users can share information on spam with their friends with similar interests, they can help each other to identify spam emails more accurately.

### 3. Designing Spam Filtering Scheme

Users and their contacts with the same interests are called "similar contacts." We assume that similar contacts always have the same opinion on the same email; then we can design

TABLE 1: Local spam list.

ID	MD5	Contact
(1)	469352d907cb67bc2b228e8b0a839eee	Zhang_ch@163.com
(2)	daeb67d732741a4982d6929ee191e210	jamesell@163.com
(3)	6d49148666475138cec9f42cc29a7cd7	qingzhi@163.com
(4)	bac0b74229c3f73757fe72508e25471a	hannan@gmail.com
(5)	2330ead823cd690611b9b990e29cc283	yangxf@upc.edu.cn

a scheme in which similar contacts share their information on spam with each other to filter semispams.

Our scheme consists of two modules: (1) Bayesian-based spam filtering deployed at an email server for all users; (2) credible similarity-based spam filtering by crowdsourcing deployed at each user local host. The structure of the proposed CPSFS is showed in Figure 1.

We deploy the Bayesian spam filtering at email servers to filter complete-spam for all users because the more the emails used for training are, the more accurately the Bayesian filter identifies spam.

The spam filtering deployed at local system uses three lists for storing information on contacts and spam reports from other users, namely, local trust list, local spam list, and local interest list. A user gets his contacts' interests and disinterests by exchanging interest lists to calculate their similarity between them. Users share information on spam with their credible contacts by pushing spam reports to their contacts. Spam reports generated at the local host and that from contacts are stored in the local spam list. At the local system, before a user browses his inbox, those emails in the inbox are checked according to spam information from the local spam list. A credible contact means the similarity and trust between the user and the contact are higher than the corresponding threshold. We construct an interest list and a trust-similarity list at each local host in order to calculate the similarities and social trust values.

*3.1. Local Spam List.* A local spam list contains an MD5 hash of the spam from spam reports and the email address which the spam report is from, as shown in Table 1.

A spam report consists of a MD5 hash of the corresponding spam, which is obtained from the content of spam email to avoid spam with an altered subject heading or a forged address. Some spam reports were generated automatically at local system. Others were from users' credible contacts.

A user may receive different spam reports on the same email from different contacts. Of these, only the report from contact with the highest interest similarity will be recorded into the local spam list; other reports will be dropped.

*3.2. Local Interest List.* The interests of a user in a social network represent the user's personality [25]. There are similar interests among users. The common disinterests between two users also indicate whether they are similar in some way. Therefore we encourage each email user to fill their own interests and disinterests in their local interest lists and exchange their lists with each other via emails.

TABLE 2: Local Interest List.

ID	Interest	Disinterest
(1)	Shopping, movie, music, food, car	Pet, beauty, drawing, cartoon, IT
(2)	Pet, shopping, food, car, reading	Game, music, movie, basketball, IT
(3)	Car, shopping, music, food, game	Singing, beauty, cartoon, drawing, reading
(4)	Music, food, movie, beauty, drawing	Shopping, music, pet, cartoon
(5)	Car, pet, cartoon, music, food	Shopping, movie, basketball, IT, beauty
(6)	IT, movie, music, basketball, reading	Food, car, singing, game, dancing
(7)	Car, food, basketball, game, IT	Movie, shopping, singing, dancing, game

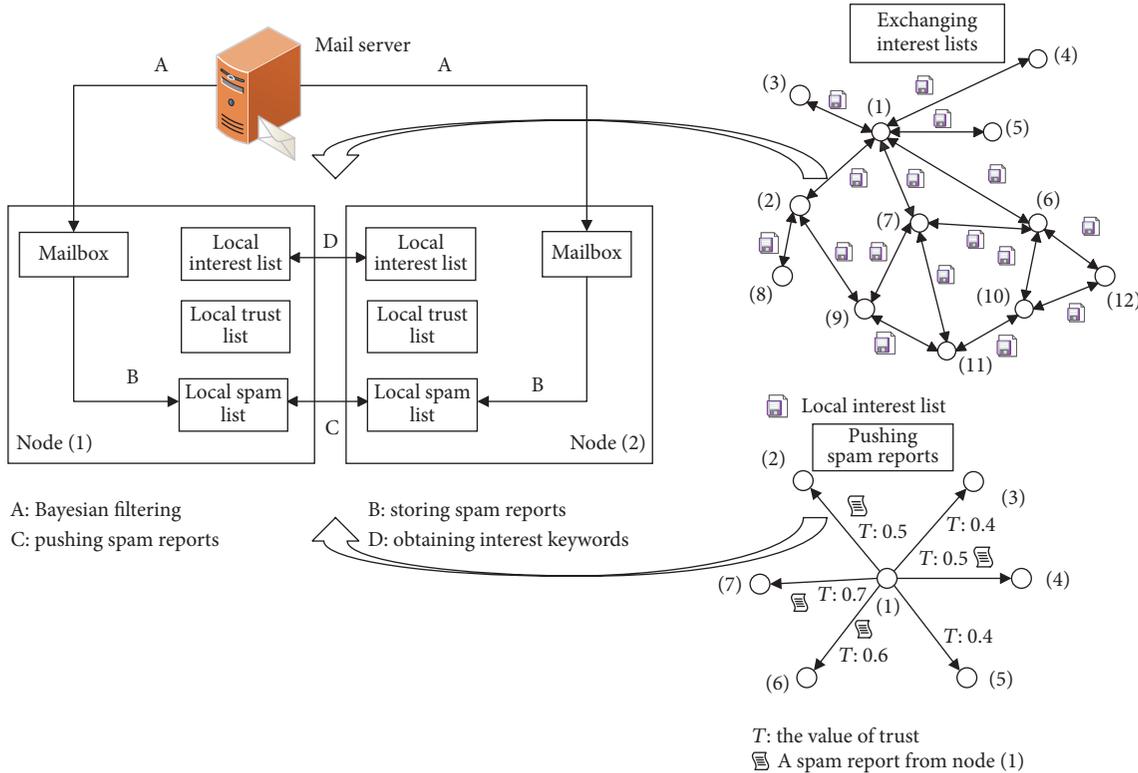


FIGURE 1: The structure of the proposed CPSFS.

These interests and disinterests can be described by some keywords. The interests and disinterests of a user and the user's contacts are stored in the local interest list as shown in Table 2. Each contact of the user has an ID in the local system. We can get the corresponding trust value and the interest similarity according to the contact's ID. Once a user gets a spam report, the local system will check the local trust-similarity list to push the spam report on this email to those credible contacts.

**3.3. Local Trust-Similarity List.** Social trust is a measure of credibility in social networks. It reflects a certain degree of similarity between users, such as likes, dislikes, social relationships, and the closeness of their interests. Direct trust is calculated initially from the historical record of direct contacts between users which indicates the direct friendships (not including friends of a friend) between users and their contacts; then it can be adjusted according to the similarities between them.

TABLE 3: Local trust-similarity list.

ID	Email address	Trust value	Interest similarity
(1)	Zhang_ch@163.com	1.00	1.00
(2)	jamesell@163.com	0.52	0.25
(3)	qingzhi@163.com	0.79	0.42
(4)	hannan@gmail.com	0.56	0.33
(5)	yangxf@upc.edu.cn	0.41	0.33
(6)	dswang@upc.edu.cn	0.60	0.17
(7)	jiaozzy@gmail.com	0.73	0.11

The interest similarity is calculated according to interests and disinterests between a sender and its recipient, which indicates the closeness between them.

Trust values and similarities of contacts on a local host are stored in a local trust-similarity list, as shown in Table 3.

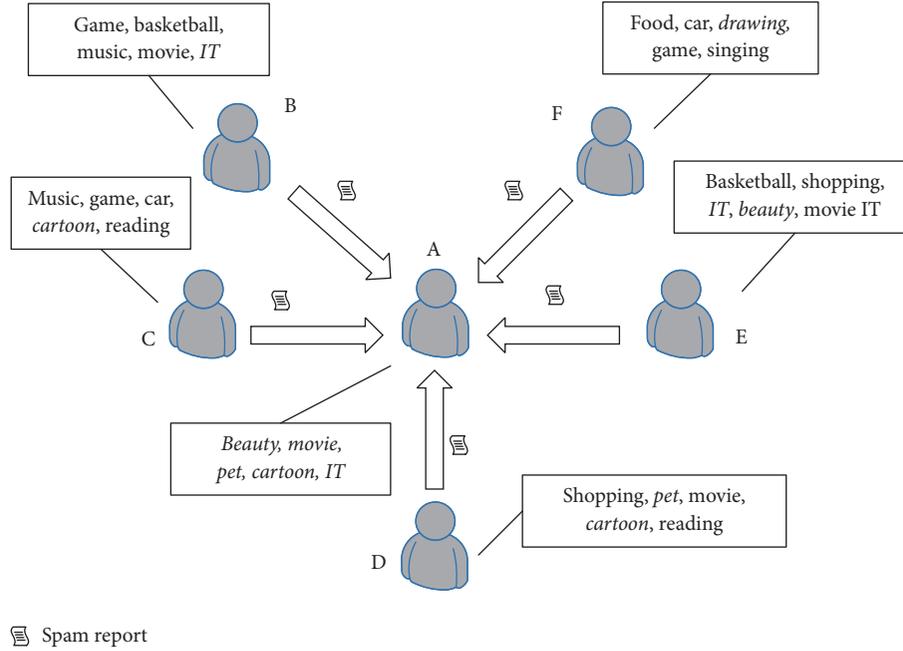


FIGURE 2: A simple example for obtaining personalized spam reports.

The email users and their relationships between them formed an email network. A simple example is shown in Figure 2, where nodes represent users and links represent the relationships between the user and their contacts.

**3.4. Obtaining Personalized Spam Reports at a Client.** A CPSFS client can filter semispam automatically by obtaining personalized reports from a user's contacts. When a user node and his/her contacts have more similar interests and disinterests, the user is more likely to obtain personalized spam reports from his or her contacts. A simple example is shown in Figure 2. We list the disinterests for each user in this figure. User A gets spam reports about *IT* from B, spam reports about *cartoon* from C, spam reports about *pet* and *cartoon* from D, spam reports about *IT* and *beauty* from E, and spam reports about *drawing* from F.

## 4. Calculating Trust Value and Interest Similarity

**4.1. Calculation of Interest Similarity.** The more the mutual interests and disinterests between a user and his or her contacts are, the more similar they are [26]. We calculate the similarity between node A and node B via

$$S(A, B) = \frac{(M_{00} + M_{11})}{(M_A + M_B - M_{11} - M_{00})}, \quad (1)$$

where  $S(A, B)$  is the similarity between A and B.  $M_A$  represents the number of A's interests and disinterests.  $M_B$  represents the number of B's interests and disinterests.  $M_{00}$  represents the number of the mutual disinterests.  $M_{11}$  represents the number of the mutual interests.

**4.2. Calculation of Trust Value.** We use the additive increase/multiplicative-decrease algorithm [27] to adjust the trust value between nodes. In this algorithm, initially, the trust value of users' contacts should be assigned. Alternatively, the trust value of a contact can be calculated according to the number of emails from the contact. We count each contact's emails in the inbox for the past month. The number of emails from the  $i$ th contact is represented by  $num_i$ . Let  $max$  represent the largest number of emails. The initial trust value of a contact is  $T_i = num_i / max$ .

If the similarity between nodes is changed, the trust value will be changed correspondingly. We use  $b$  to indicate the degree of trust value changes. So the trust value will be changed via

$$t_{ij} = t_{ij} \pm b \quad (0 < b < 1). \quad (2)$$

If the interest similarity is higher than the threshold of similarity and the trust value between them is less than the given trust threshold, the trust will increase  $b$  in formula (2). Otherwise the trust value will decrease  $b$ . It is important to set an appropriate value for  $b$ . We will discuss how to adjust the trust value threshold rationally in the evaluation section.

## 5. Spam Filtering Process at Local System

Users can mark an email manually or automatically as spam at local system.

**5.1. Mark an Email Automatically.** All emails of a user are examined by a Bayesian filter at an email server before they reach clients [28]. When the user logs in, the local system should check all emails in the inbox. If there is a spam report in the inbox, the content should be extracted and written

into the local spam list and then the system should forward the email to similar contacts whose similarity is beyond the threshold. Otherwise, the system will calculate the MD5 hash for this email, if an email is identified as spam using the local spam list. That is to say, when the MD5 hash matches an item in the local spam list, the spam is put into a junk box and the spam report is pushed to the similar contacts. The corresponding filtering process is handled by *Algorithm 1 LocSpamFilter*.

**5.2. Mark an Email Manually.** While a user browses the inbox, he or she may find some spam. Once the user puts a spam email into a junk box, the local system will generate a spam report, add it to the local spam list, and push it to similar contacts.

Pushing a spam report is accomplished via sending emails [29]. The subject of the email is *spam report*, which is used to distinguish spam report from other emails. The content of this kind of email is the MD5 hash of this spam. When a user identifies an email as spam, the spam report is automatically generated and pushed to similar contacts.

The trust value is used to limit the recipients of spam reports from the credible users. If the trust value of a contact is over a given threshold, this user will receive spam reports from this contact. If the similarity to a contact is above a given threshold, the local system will push spam reports to the contact. The similarity threshold is used to limit the scope of spam report propagation to similar contacts only. This will both improve the accuracy of identifying semispam, and reduce network overhead. We will discuss the threshold of trust and interest similarity in the evaluation section.

## 6. Experiments and Evaluations

**6.1. Simulation Settings.** The social network we used in our experiments is from Datatang and contains 1133 nodes and 10903 edges [30]. The average number of contacts of each node is 9.63. In addition, we use a sample set of 3000 emails including 1000 spam emails and 2000 legitimate emails from SpamAssassin public mail corpus [31]. We choose 1000 emails from the sample randomly as definite sample, and 500 emails from email boxes of our researchers as indefinite samples. We set the total number of interest keywords to 15, and the average number of interest keywords for each user to 10; then we sent an email of the 1500 emails randomly to 10% of the total nodes.

In our experiments, for formula (2), the initial trust value  $t_{ij}$  is randomly set from 0.5 to 1.0, the threshold of trust value is 0.5 and the initial value of  $b$  is 0.1. If a node received a spam email report from his friend and the trust value between them is less than the given trust threshold, the trust value between him and his friend will increase by  $b$ , which can make a user become trusted by his friend.

**6.2. Comparisons with Other Methods and Accuracy of the CPSFS.** We calculate the accuracy of the CPSFS by comparing with the Bayesian filter. The accuracy rate of filtration is  $R_a$ . We calculate the accuracy rate via

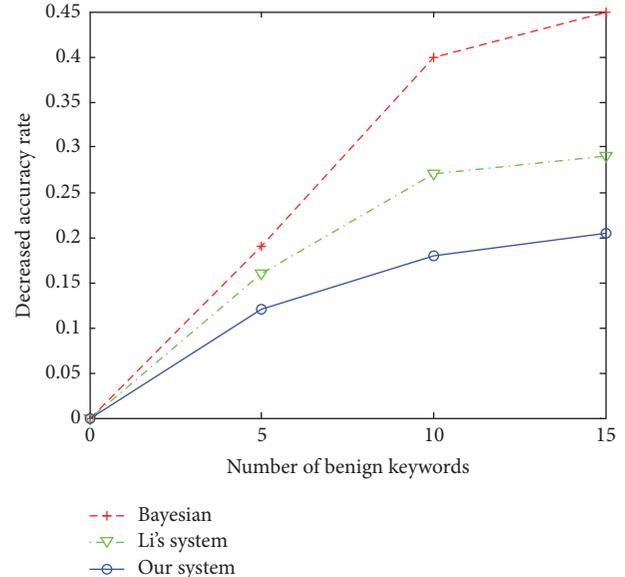


FIGURE 3: The accuracy rate under poison attacks.

$$R_a = \frac{n_1 + (N - n_1) \times \left(\frac{n_f}{n_i}\right) \times \left(\frac{n_c}{n_s}\right)}{N}, \quad (3)$$

where  $N$  is the number of emails used in our experiments. The number of email correctly classified by Bayesian filter is denoted by  $n_1$ .  $n_i$  is the number of emails classified incorrectly by Bayesian filter.  $n_f$  is the number of emails classified as legitimate but are actually spam. When an email is classified correctly by the CPSFS, two cases are supposed to be considered: (1) a user considers it as a legitimate email, and the CPSFS did not mark it as spam; (2) a user considers it as spam, and the CPSFS marked it as spam. If an email is classified by the system correctly, we will record the number of users, which is denoted as  $n_c$ .  $n_s$  is the total number of the users in the social network. We evaluate the system using the same settings as in Section 6.1. The results from formula (3) show that our CPSFS has a higher accuracy rate than that of the Bayesian filter and Li's work (95.1% versus 91.4% versus 93.9%).

Poison attack is that an attacker adds benign keywords into emails intentionally in order to avoid spam to be identified by a spam filter. As word segmentation is the basis of a Bayesian filter, keywords of mail contents have a crucial impact on classification and performance of a Bayesian filter. The decreased accuracy is the decreased value of accuracy when the poison attacks happen. Figure 3 shows the decreased accuracy of the Bayesian filter, Li and Shen's work [27], and the CPSFS when they are subjected to poison attacks. When the number of benign keywords is set to 0, 5, 10, and 15, the accuracy of our CPSFS decrease smoothly. But the accuracy of the Bayesian filter decreases more quickly than that of our CPSFS because the Bayesian filter is entirely dependent on the detection of spam content. Our CPSFS reduces the effects of poison attacks on accuracy by sharing information on spam using crowdsourcing. The CPSFS considers the association of an email and its recipients,

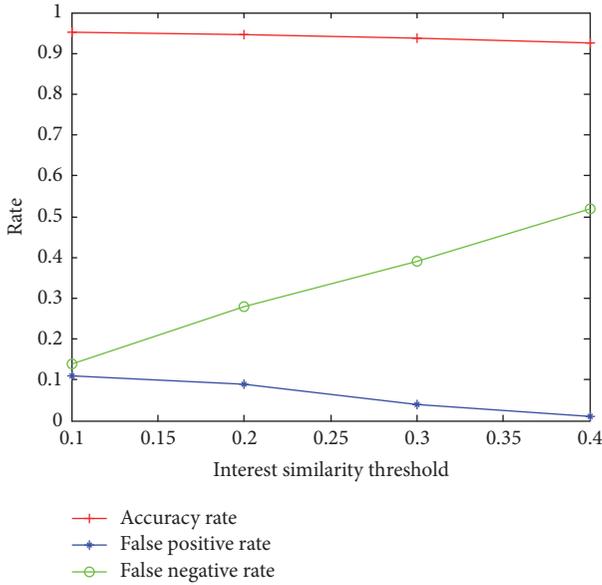


FIGURE 4: The false positive rate and false negative rate.

which helps to reduce the effects of poison attacks to some extent.

The false positive rate and false negative rate of CPSFS are shown in Figure 4. False negative represents spam that is classified as a legitimate email incorrectly. False positive represents a legitimate email that is classified as spam incorrectly. The interest similarity threshold is set to 0.1, 0.2, 0.3, and 0.4, respectively. The trust threshold is set to 0.5. The false positive rate decreases and the false negative rate increases as the interest similarity is increasing.

### 6.3. Different Trust Threshold and Interest Similarity Threshold.

For formula (2), the value of parameter  $b$  influences the trust calculation and the accuracy rate. In our evaluations, the value of  $b$  is set to 0.1, 0.2, and 0.3, respectively, and the accuracy rate results are shown in Figure 5. We can see that when the trust threshold is set to 0.5, all the three curves reach their accuracy peaks. In this figure, the accuracy rate changes dramatically as the trust threshold changes in all the three curves, which indicates that the trust threshold affects the accuracy significantly. When  $b$  is set to 0.1, the correlation between accuracy and trust is the best as shown in Figure 5. Therefore the value of  $b$  is set to 0.1 when calculating trust values.

The interest similarity threshold is a key factor which influences the performance of spam filtering. The results are shown in Figure 6, where the trust thresholds are set to 0.4, 0.5, 0.6, and 0.7, respectively. It shows that the accuracy rate changes when the interest similarity threshold increases from 0.1 to 0.4. The accuracy rate is always higher when the interest similarity threshold is set to 0.1 compared with other interest similarity thresholds. The accuracy decreases when the interest similarity threshold increases because users will not push their spam reports if the interest similarities are lower than the corresponding thresholds.

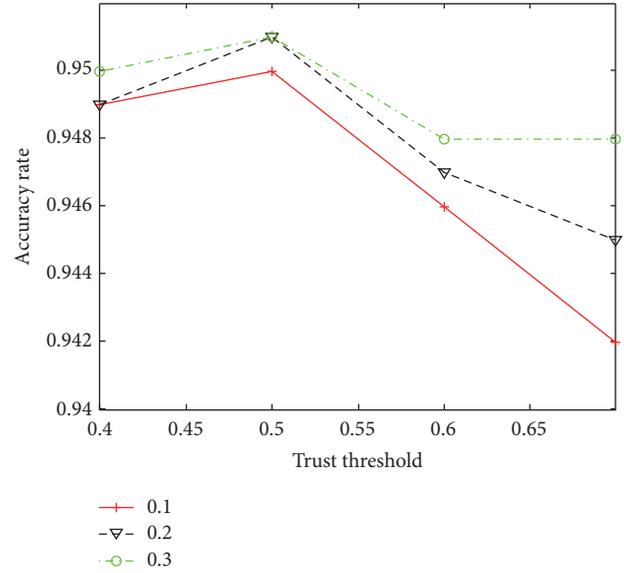


FIGURE 5: Accuracy rate under different trust threshold.

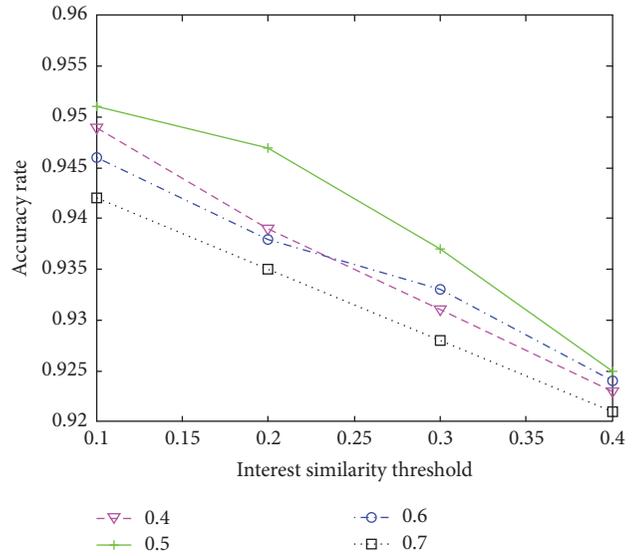


FIGURE 6: The accuracy rate under different interest similarity threshold.

## 7. Conclusion

To handle spam effectively, in this paper, we propose a credible and personalized spam filtering scheme (CPSFS) based on social trust and interest similarity, where users report their received spam emails to their contacts in social networks. We introduced local lists and social sensing mechanism for spam reports. The trust value and similarity are calculated to determine whether users should push spam reports to their friends. The social trust and similarity increase the credibility of the CPSFS filtering scheme. Our experiments showed that the accuracy of our CPSFS is better than the conventional Bayesian filter and some other approaches.

There is work on Copy Adjustable Incentive Scheme (CAIS) that adopts virtual credit concept to stimulate selfish nodes to cooperate in data forwarding [32]. We will consider incorporating this idea to the situation where some users are reluctant to share interests. In the future, we will also improve the performance by improving the network connectivity and throughput [33]. In addition, we will apply our CPSFS scheme to other social networks such as mobile social networking and vehicular social networks [34].

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

The work presented in this paper is supported by the Key Research Program of Shandong Province (no. 2017GGX10140), National Natural Science Foundation of China (no. 61309024, and no. 61772551), the Program on Innovative Methods of Work from Ministry of Science and Technology, China (no. 2015010300), Shandong Provincial Natural Science Foundation (no. ZR2015FM022), and the Fundamental Research Funds for the Central Universities.

## References

- [1] Z. Miller, B. Dickinson, W. Deitrick, W. Hu, and A. H. Wang, "Twitter spammer detection using data stream clustering," *Information Sciences*, vol. 260, pp. 64–73, 2014.
- [2] J. Shen, R. H. Deng, Z. Cheng, L. Nie, and S. Yan, "On robust image spam filtering via comprehensive visual modeling," *Pattern Recognition*, vol. 48, no. 10, pp. 3227–3238, 2015.
- [3] B. Cui, Z. Liu, and L. Wang, "Key-Aggregate Searchable Encryption (KASE) for Group Data Sharing via Cloud Storage," *IEEE Transactions on Computers*, vol. 65, no. 8, pp. 2374–2385, 2016.
- [4] Z. Liu, X. Chen, J. Yang, C. Jia, and I. You, "New order preserving encryption model for outsourced databases in cloud environments," *Journal of Network and Computer Applications*, vol. 59, pp. 198–207, 2016.
- [5] *China Internet Association Anti-Spam center: Anti-spam survey report*, <http://www.acma.gov.au/theACMA/spam-statistics>.
- [6] Y.-M. Li, H.-W. Hsiao, and Y.-L. Lee, "Recommending social network applications via social filtering mechanisms," *Information Sciences*, vol. 239, pp. 18–30, 2013.
- [7] W. Mason, J. W. Vaughan, and H. Wallach, "Computational social science and social computing," *Machine Learning*, vol. 95, no. 3, pp. 257–260, 2014.
- [8] Y. Ren and D. Ji, "Neural networks for deceptive opinion spam detection: An empirical study," *Information Sciences*, vol. 385–386, pp. 213–224, 2017.
- [9] H. Shen, Z. Li, J. Liu, and J. E. Grant, "Knowledge sharing in the online social network of Yahoo! Answers and its implications," *IEEE Transactions on Computers*, vol. 64, no. 6, pp. 1715–1728, 2015.
- [10] Y. Ren, J. Shen, J. Wang, J. Han, and S. Lee, "Mutual verifiable provable data auditing in public cloud storage," *Journal of Internet Technology*, vol. 16, no. 2, pp. 317–323, 2015.
- [11] D. Quinn, L. Chen, and M. Mulvenna, "Social network analysis: A survey," *International Journal of Ambient Computing and Intelligence*, vol. 4, no. 3, pp. 46–58, 2012.
- [12] Y. A. Kim and M. A. Ahmad, "Trust, distrust and lack of confidence of users in online social media-sharing communities," *Knowledge-Based Systems*, vol. 37, pp. 438–450, 2013.
- [13] S. J. Yu, "The dynamic competitive recommendation algorithm in social network services," *Information Sciences*, vol. 187, no. 1, pp. 1–14, 2012.
- [14] F. Liu, X. Li, Y. Ding et al., "A social network-based trust-aware propagation model for P2P systems," *Knowledge-Based Systems*, vol. 41, pp. 8–15, 2013.
- [15] X. Liu, L. Shi, Y. Wang, Z. Xin, and W. Fu, "A dynamic trust conference algorithm for social network," in *Proceedings of the 2013 8th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, 3PGCIC 2013*, pp. 340–346, France, October 2013.
- [16] X. Liu, Y. Wang, D. Zhao, W. Zhang, and L. Shi, "Patching by automatically tending to hub nodes based on social trust," *Computer Standards & Interfaces*, vol. 44, pp. 94–101, 2016.
- [17] J. Jaeyeon and S. Emil, "An Empirical Study of Spam Traffic and the Use of DNS Black Lists," in *Proceedings of the 4th ACM SIGCOMM Conference on Internet measurement*, pp. 370–375, October 2004.
- [18] C. O'Brien and C. Vogel, "Spam filters: Bayes vs. Chi-squared; letters vs," in *Proceedings of the in Proceedings of the 1st International Symposium on Information and Communication Technologies*, pp. 291–296, September 2003.
- [19] P. Haider, U. Brefeld, and T. Scheffer, "Supervised clustering of streaming data for email batch detection," in *Proceedings of the 24th International Conference on Machine Learning, ICML 2007*, pp. 345–352, USA, June 2007.
- [20] B. Scholkopf and J. B. Platt, "Dirichlet-Enhanced spam filtering based on biased samples," in *Proceedings of the International Conference on Neural Information Processing Systems*, pp. 161–168, MIT Press, January 2006.
- [21] D. Zisiadis, S. Kopsidas, A. Varalis, and L. Tassioulas, "Mailbook: A social network against spamming," in *Proceedings of the 2011 International Conference for Internet Technology and Secured Transactions, ICITST 2011*, pp. 245–249, are, December 2011.
- [22] P. O. Boykin and V. P. Roychowdhury, "Leveraging social networks to fight spam," *The Computer Journal*, vol. 38, no. 4, pp. 61–68, 2005.
- [23] M. Sirivianos, K. Kim, and X. Yang, "SocialFilter: Introducing social trust to collaborative spam mitigation," in *Proceedings of the IEEE INFOCOM 2011*, pp. 2300–2308, China, April 2011.
- [24] H. Shen and Z. Li, "Leveraging social networks for effective spam filtering," *IEEE Transactions on Computers*, vol. 63, no. 11, pp. 2743–2759, 2014.
- [25] C. Wilson, B. Boe, A. Sala, K. P. N. Puttaswamy, and B. Y. Zhao, "User interactions in social networks and their implications," in *Proceedings of the 4th ACM European Conference on Computer Systems, EuroSys'09*, pp. 205–218, Germany, April 2009.
- [26] M. Kobayakawa, S. Kinjo, M. Hoshi, T. Ohmori, and A. Yamamoto, "Fast computation of similarity based on jaccard coefficient for composition-based image retrieval," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 5879, pp. 949–955, 2009.
- [27] Z. Li and H. Shen, "SOAP: A Social network Aided Personalized and effective spam filter to clean your e-mail box," in *Proceedings of the IEEE INFOCOM 2011*, pp. 1835–1843, China, April 2011.

- [28] Z. Zhong and K. Li, "Speed up statistical spam filter by approximation," *IEEE Transactions on Computers*, vol. 60, no. 1, pp. 120–134, 2010.
- [29] S. Hameed, X. Fu, P. Hui, and N. Sastry, "LENS: Leveraging social networking and trust to prevent spam transmission," in *Proceedings of the 19th IEEE International Conference on Network Protocols, ICNP 2011*, pp. 13–18, Canada, October 2011.
- [30] Datatang., <http://www.datatang.com/data/796>.
- [31] "Spam Assassin," <http://spamassassin.apache.org/downloads.cgi?update=201504291720/>.
- [32] Z. Ning, L. Liu, F. Xia, B. Jedari, I. Lee, and W. Zhang, "CAIS: a copy adjustable incentive scheme in community-based socially aware networking," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 3406–3419, 2017.
- [33] Z. Ning, F. Xia, X. Hu, Z. Chen, and M. S. Obaidat, "Social-oriented adaptive transmission in opportunistic internet of smartphones," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 810–820, 2017.
- [34] Z. Ning, F. Xia, N. Ullah, X. Kong, and X. Hu, "Vehicular Social Networks: Enabling Smart Mobility," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 16–55, 2017.

## Research Article

# Color Distribution Pattern Metric for Person Reidentification

Yingsheng Ye, Xingming Zhang, and Wing W. Y. Ng

School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China

Correspondence should be addressed to Yingsheng Ye; 1543590032@qq.com

Received 18 July 2017; Accepted 27 November 2017; Published 18 December 2017

Academic Editor: Zhaolong Ning

Copyright © 2017 Yingsheng Ye 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.

Accompanying the growth of surveillance infrastructures, surveillance IP cameras mount up rapidly, crowding Internet of Things (IoT) with countless surveillance frames and increasing the need of person reidentification (Re-ID) in video searching for surveillance and forensic fields. In real scenarios, performance of current proposed Re-ID methods suffers from pose and viewpoint variations due to feature extraction containing background pixels and fixed feature selection strategy for pose and viewpoint variations. To deal with pose and viewpoint variations, we propose the color distribution pattern metric (*CDPM*) method, employing color distribution pattern (*CDP*) for feature representation and SVM for classification. Different from other methods, *CDP* does not extract features over a certain number of dense blocks and is free from varied pedestrian image resolutions and resizing distortion. Moreover, it provides more precise features with less background influences under different body types, severe pose variations, and viewpoint variations. Experimental results show that our *CDPM* method achieves state-of-the-art performance on both 3DPeS dataset and ImageLab Pedestrian Recognition dataset with 68.8% and 79.8% rank 1 accuracy, respectively, under the single-shot experimental setting.

## 1. Introduction

The person reidentification (Re-ID) task searches for a targeted person from images captured in different times and places under the assumption that the target person is wearing the same clothing as provided in source images. With wide deployments of public surveillance infrastructures and private surveillance products, people are paying more attention to Re-ID than ever before and are desperate for assistance from Re-ID methods when searching for an individual among massive amounts of surveillance videos. Although many methods have been proposed in the last decade with notable progress, Re-ID methods still face many challenges in real scenarios. Intra-class differences of pose and viewpoint variations, inter-class similarities of appearance, different camera settings, and significant environment changes together make the Re-ID task much more complicated and challenging. Movements of body parts cause intra-class differences in pose and viewpoints, while inter-class similarities include resemblances between 2 individuals sharing similar body types with similar or identical clothing. Furthermore, significant environmental changes like illumination changes and occlusions have great influence on a person's appearance and bring environmental background noises. To deal with

these challenges, most Re-ID algorithms focus on finding robustness features and a reliable similarity classification metric. For robustness features, unlike shape, position, soft-biometry, and body models, combination features of color and texture information based on dense blocks are very popular in many approaches well summarized in [1]. Besides these features, silhouette information also has its own challenges and opportunities in Re-ID. Silhouette information comes from background abstraction of surveillance videos. In IoT, it is possible to integrate background abstraction and device management service like the lightweight RESTful Web service [2] into IP cameras to help ease the load of IoT and make IP cameras smarter. In our work, we traded texture information for silhouette information using combination of color and silhouette information. As for finding a reliable similarity classification metric, an annoying problem is lack of sufficiently labeled training samples, especially positive training samples. Therefore, we proposed a grouping strategy called Pure-vs-Exception to separate training images into different groups and automatically generate sufficient and balanced labeled training samples.

In this paper, we propose a color distribution pattern metric (*CDPM*) method for Re-ID. *CDPM* consists of a color distribution pattern (*CDP*) feature model for feature

extraction and a SVM [3] for training and classification, under the assumption that the target person still wears the same clothing. Our *CDP* feature model is designed to deal with background interference, pose variations, and viewpoint variations. It extracts color distribution patterns from HSV and Lab color spaces, combining with silhouette information based on leg and torso-head (TH) body parts derived from the leg-torso-head body model [4]. In similarity classification metric learning, we aim to maximize differences of interclass variations and suppress differences of intraclass variations by proposing the Pure-vs-Exception strategy. We use Pure-vs-Exception strategy to group pedestrian images and generate training samples, which simulate significant difference patterns between 2 pedestrian images of 2 different individuals and significant difference patterns between 2 pedestrian images of the same individual. Experimental results show that our *CDPM* method outperforms other comparative approaches on both 3DPeS and ImageLab Pedestrian Recognition (IPR) datasets and possesses great tolerance for pose and viewpoint variations.

The remainder of this paper is as follows. Section 2 presents a review of related works. Section 3 is a detailed introduction of the proposed *CDPM*. Experimental results on 2 public datasets are covered in Section 4. Section 5 concludes this paper.

## 2. Related Work

Almost all Re-ID approaches are based on 2-step framework: (i) feature extraction from pedestrian image pairs and (ii) feature similarity calculation under a prelearned classification metric. Thus, most efforts are devoted to these 2 steps. According to the Re-ID taxonomy summarized by Vezzani et al. [1], local features and pedestrian body model-based features are very common because they capture detailed and localized information for matching. Among local features, it is very popular to use combination features of color and texture histograms, for example, color histograms (from HSV color space) and scale invariant local ternary pattern (SILTP) histograms [5], color histograms (from RGB, YUV, and HSV color spaces) and local binary pattern (LBP) histograms [6, 7], color histograms (from RGB and HSV color spaces) and LBP histograms [8], color histograms (from RGB, YCbCr, and HS color spaces) and texture histograms of Gabor filters and Schmid filters [9], and color histograms (from HSV and Lab color spaces) and LBP histograms [10, 11]. The implementation details of these color and texture histograms might be different. After feature extraction, it is very important to find a reliable similarity classification metric exploiting feature representativeness. Hence, a variety of strategies are introduced when searching for a reliable classification metric.

Some approaches focus on cross-domains classification metric learning. Hu et al. [7] employed deep learning in a transfer metric, which learned hierarchical nonlinear transformations of cross-domains by mapping discriminative knowledge between a labeled source domain and unlabeled target domain. Others [12] jointly learned a transfer metric in an asymmetric way by extracting discriminant shared components through multitask modeling to enhance target

interclass differences under shared latent space. Recently, Shi et al. [13] showed interest in attribute features extracted from semantic level for the Re-ID task and employed it in cross-domains transfer metric learning.

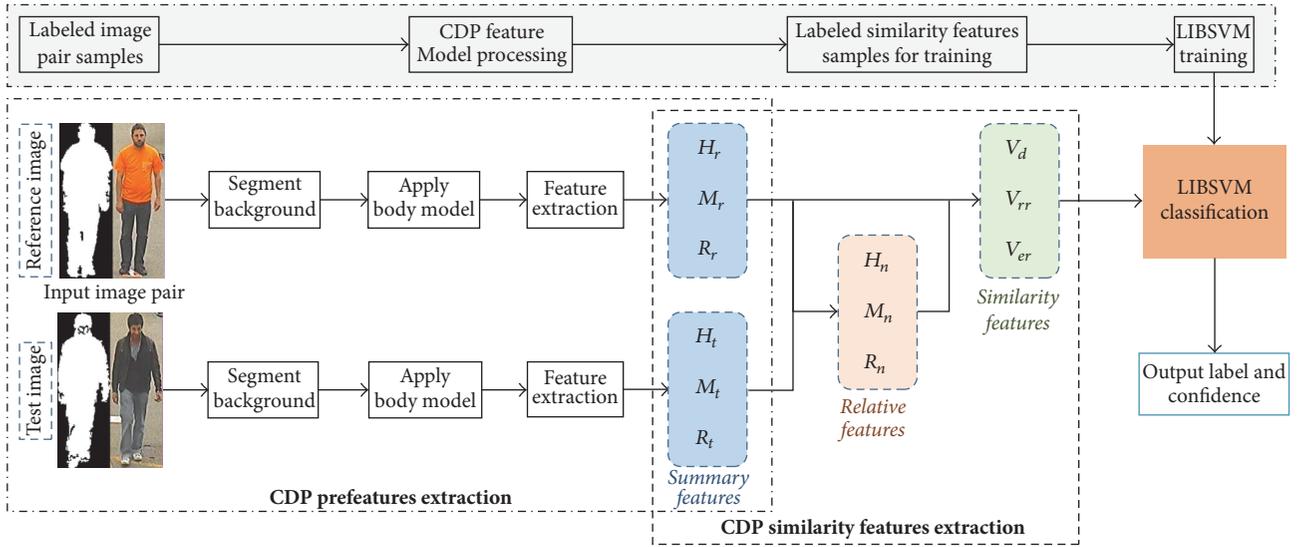
Most of these approaches focused on classification metric learning under the same domain. In [14], authors used an improved deep learning architecture with 2 novel layers to extract the relationships of the input image pair. One layer computed the cross-input neighborhood differences and the other subsequently summarized these differences. Köstinger et al. [15] presented KISS classification metric learned from equivalence constraints based on statistical inference rather than computational complex optimization. However, this classification metric learning could be unstable under a small-sized training set, mentioned in [10]. Thus, Tao et al. [10] integrated smoothing and regularization techniques in KISS for robust estimation of covariance matrices and stable performance and proposed regularized smoothing KISS. Pedagadi et al. [16] used local Fisher discriminant analysis (LFDA) to learn similarity classification metric from a set of labeled feature pairs, which consisted of features extracted from high-dimensional features by unsupervised PCA. However, unsupervised PCA undesirably compressed most of discriminative features under relatively small dataset. Hence, Xiong et al. [6] used kernel approach to replace the unsupervised dimensionality reduction by supervised dimensionality reduction and proposed kernel LDFA (kLDFA), avoiding eigenvalue decomposition of large scatter matrices and replacing kernel easily for better performance. Other researchers [11] used a reference set to learn a reference subspace by regularized canonical correlation analysis. Then, similarity was calculated by the features projected in reference subspace and reranked by saliency information. Conversely, Liao et al. [5] proposed cross-view quadratic discriminant analysis to learn similarity classification metric under a low-dimensional subspace. Furthermore, Zhang et al. [17] treated hashing learning as regularized similarity learning and used deep learning to train hashing codes and additional regularization term, which encoded adjacency consistency.

Besides approaches with a fixed classification metric for all image pairs, some approaches use multiple metrics in Re-ID. To address viewpoint variations, Wang et al. [8] proposed a data-driven distance metric method based on cross-view support consistency and cross-view projection consistency to readjust the distance metric for each image pair by exploiting the training data each time, which was quite computational for a large scale. Ma et al. [9] modeled a multitask distance metric for the Re-ID in camera networks and proposed multitask maximally collapsing metric learning method to learn distance metrics between different cameras. Others [19] proposed a spatiotemporal feature named optical flow energy image (OFEI) and a corresponding matching method named multiview relevance metric learning with listwise constraints (mvRMLLC) for video-based Re-ID. The mvRMLLC assumed that OFEI features of the same person from different views should be globally consistent with their similarities.

Different from these above 2-dimensional- (2D-) based methods, a 3-dimensional (3D) method called SARC3D [20]



FIGURE 1: Sample images from 3DPeS dataset.


 FIGURE 2: CDPM for person reidentification. The *CDP* feature model contains two parts: prefeatures extraction and similarity features extraction.

used averaged side, frontal, and top views silhouettes to learn an approximate graphical 3D body model, which then could compare people from even 1 image due to its precise 3D feature mapping via a selected set of vertices. This comparison not only allowed one to look for details and global features but also copped with partial data and occluded views.

However, methods based on dense block features [5–14, 16], which encode not only person pixel information but also background pixel information, cannot properly filter out background information due to pose variations and different body types. Furthermore, dense block features introduce distortion caused by resizing from different resolutions. Figure 1 shows different body types (including massive build, medium build, and slim build), pose variations, and viewpoint variations (note that viewpoint variations in this paper refer to front view, back view, left view, and right view or any state in-between but do not include top view). Matching accuracies of these aforementioned methods [5–14, 16] suffer from encoded background information due to inappropriate feature selection, in which case the fixed classification metric could not properly separate encoded person information from encoded background information, under significant pose variations and different body types. We refer to this kind of interference as background interference.

Our *CDP* feature model is designed to deal with background interference by filtering out pedestrian image background pixels using silhouette images provided or extracted from surveillance sequences and generates more precise pedestrian features than those obtained by other methods [5–11, 14]. In addition, our *CDP* feature model does not have to worry about body type variations, pose variations, and image resolution variations, which are common in real scenarios of video surveillance and video forensics, as long as the silhouette images are well provided.

### 3. Color Distribution Pattern Metric

Our *CDPM* is designed to deal with severe pose and viewpoint variations, as well as background influences. Furthermore, resolutions of pedestrian images are not required to be fixed in *CDPM*, different from fixed image resolutions in [5–14, 16]. This characteristic of *CDPM* avoids distortion interference brought by resizing.

Figure 2 shows the main processes of our *CDPM*, including *CDP* feature model, SVM training, and SVM classification. *CDP* feature model is used to extract similarity features from image pairs, while SVM is used for training and classification based on similarity features. We divide *CDP*

feature model into 2 phases: *CDP* prefeatures extraction and *CDP* similarity features extraction. Implementations of input images in the prefeatures extraction phase are independent from image pairs and could be preprocessed parallelly. For *CDP* prefeatures extractions in Figure 2, inputs include two pedestrian images (an image pair of test image and reference image) with corresponding silhouette mask images, which are in the form of binary pixels. Through successive implementations of background segmentation, applying body model, and features extraction, we obtain summary features for each pedestrian image at the end of *CDP* prefeatures extraction phase. In *CDP* similarity extraction phase from Figure 2, we use summary features of test and reference images to produce relative features. And then we extract similarity features from relative features and summary features of reference image only. Finally, we use similarity features as the SVM input for classification metric learning and prediction. In classification metric learning, shown in the top row of Figure 2, training samples are labeled similarity features extracted from labeled image pairs, where each image pair contains one test image and one reference image.

The following are the detailed implementations of *CDP* feature model and metric learning.

**3.1. *CDP* Prefeature Extraction.** We use background pixels ratios calculated by (1) to observe percentages of background pixels in pedestrian images.

$$R_{\text{background pixels}} = 1 - \frac{\sum_{x=0}^{r-1} \sum_{y=0}^{c-1} B(x, y)}{r * c}, \quad (1)$$

where  $r$  and  $c$  represent rows and columns of corresponding silhouette image, respectively, and  $B$  represents the binary image of corresponding silhouette image. According to 3DPeS dataset [21], background pixels ratios of pedestrian images vary from 34% to 89%, with a mean value of 59.4%. In IPR dataset [18], background pixels ratios vary from 36% to 80%, with a mean value of 59%. The average background pixels ratio is about 59.2% in these two datasets, which could bring great interference for feature robustness and classification if they are not filtered out before final classification. To avoid interference caused by background pixels, we use silhouette images to segment pedestrian from the background by setting background pixels values to 0. First, silhouette images are transformed into binary images using thresholds. Then, we multiply the binary image with corresponding original image at pixel level for each channel. We filter out background pixels of pedestrian images before feature extraction, saving a lot of work during classification metric learning by enhancing feature efficiency and robustness. So, we finish part of the work of feature selection at the beginning of feature extraction, unlike most of state-of-the-art methods, which implemented feature selection after applying dense block feature extraction.

After filtering out background pixels, we divide the pedestrian image into 2 parts, leg and torso-head (TH), based on the leg-torso-head [4] body model. This implementation aims to extract locations of body parts and enhance robustness of *CDP* model. The main reason we choose leg and TH parts is

that most clothing styles around the world share a commonality that there are differences between leg and TH parts. Even though the whole outfit may share the same color, it is still essential and practical to treat body as 2 parts, while the other one from input image pair shares a different clothing style. However, due to small proportion of head part and low resolution of pedestrian images in real surveillance scenarios and datasets, it is neither necessary nor significant to treat head part alone. Besides, it is difficult to automatically and accurately divide the head part. Therefore, we choose to put torso and head parts together as TH to reduce computation cost for more practical implementation.

When these 2 aforementioned implementations are ready, we implement color extraction. We use both HSV and lab color spaces to extract color distribution histogram matrices on leg and TH parts, respectively, while ignoring the pixels with a value of 0. Details of color distribution extraction are as follows. With 2 color spaces involved, we obtain 6 channels of data for each image and group them into 3 pairs: HS, ab, and LV. We then apply 2D histogram extraction on these 3 pairs of channels to extract color distribution histogram matrices with sizes (16, 32) for HS channels, (32, 32) for ab channels, and (32, 32) for LV channels. So, we obtain 6 histogram matrices, 3 histogram matrices each for the leg and TH parts, respectively. We mark these histogram matrices as  $H$ . For each histogram matrix, we generate a binary matrix with a threshold value of 3 for noises reduction. Then we mark these 6 binary matrices as occurrence masks  $M$ . The occurrence masks specify the domain boundaries for corresponding histogram matrices. We use occurrence masks  $M$  to update the corresponding histogram matrices  $H$  by multiplying  $H$  with  $M$  at pixel level as follows:

$$H(x, y) = H(x, y) M(x, y). \quad (2)$$

We use these 6 occurrence masks to calculate the corresponding occurrence ratios, plus mean occurrence ratios of occurrence ratios on leg and TH parts for each pair of channels, which produced 3 mean occurrence ratios. The occurrence ratio, not the mean occurrence ratio, is calculated as follows:

$$R_{\text{occurrence-ratio}} = \frac{\sum_{x=0}^{r-1} \sum_{y=0}^{c-1} M(x, y)}{r * c}. \quad (3)$$

In results, there are 9 occurrence ratios and we mark these 9 occurrence ratios as  $R$ . At this point, summary features including histogram matrices  $H$ , occurrence masks  $M$ , and occurrence ratios  $R$  are extracted, representing the color distribution of each pedestrian image without background pixels. Thus, for the reference image in Figure 2, we have its summary features marked as  $H_r$ ,  $M_r$ , and  $R_r$ . As for the test image in Figure 2, we have its summary features marked as  $H_t$ ,  $M_t$ , and  $R_t$ .

**3.2. *CDP* Similarity Features Extraction.** Obviously, summary features extracted above are not the final data that we want in the classification phase. There must be a pair of images in the matching phase: one as the reference pedestrian image and the other as the test pedestrian image. From last subsection,

we already get  $H_r$ ,  $M_r$ , and  $R_r$  for reference pedestrian image and  $H_t$ ,  $M_t$ , and  $R_t$  for test pedestrian image. Now, we use  $H_r$  and  $M_r$  to filter out some certain irrelevant patterns in  $H_t$  and  $M_t$  by multiplying them at pixel level, shown in (4). We treat the resulting features as parts of the relative features and mark them as  $H_n$  and  $M_n$ .

$$\begin{aligned} H_n(x, y) &= H_t(x, y) M_r(x, y), \\ M_n(x, y) &= M_t(x, y) M_r(x, y). \end{aligned} \quad (4)$$

Besides  $H_n$  and  $M_n$ , relative features also include occurrence ratios  $R_n$ . We calculate  $R_n$  directly from  $M_n$  using (3) in *CDP* prefeature extraction phase.  $H_n$  enables filtering out some certain irrelevant color distribution patterns while letting similarity matching focus on the rest.  $M_n$  enables performing a domain boundary match in later implementation. In other words,  $H_n$  focuses on the fine-grain distribution matching, while  $M_n$  and  $R_n$  focus on domain boundaries matching.

Next step is to use the summary features of reference pedestrian image and relative features to generate similarity features for SVM [3] training and classification. Firstly, we normalize both  $H_r$  and  $H_n$  by dividing the cumulated sum of their own, respectively. This normalization aims to reduce fluctuation noises caused by occlusions, pose variations, and viewpoint variations, making  $H_r$  and  $H_n$  more compatible for different scales of input images while well preserving color distribution patterns. Then, we compute the absolute difference matrices of  $H_r$  and  $H_n$ . Right after that, we project these difference matrices into a subspace by computing the corresponding cumulated sums of absolute difference matrices as distance values by (5). We mark these 6 distance values calculated from the corresponding absolute difference matrices as  $V_d$ .

$$V_d = \text{sum} \left( \text{abs} \left( \frac{H_n}{\text{sum}(H_n)} - \frac{H_r}{\text{sum}(H_r)} \right) \right). \quad (5)$$

Secondly, we get 6 occurrence relative ratios  $R_{r-o-r}$  from corresponding  $M_r$  and  $M_n$  calculated by (6), plus 1 summary relative ratio  $R_{s-r-r}$  calculated by (7), resulting in 7 relative ratios marked as  $V_{rr}$ .

$$R_{r-o-r} = \frac{\sum_{x=0}^{r-1} \sum_{y=0}^{c-1} (M_n(x, y) M_r(x, y))}{\sum_{x=0}^{r-1} \sum_{y=0}^{c-1} M_r(x, y)}, \quad (6)$$

$$R_{s-r-r} = \frac{\sum_{i=1}^6 \sum_{x=0}^{r-1} \sum_{y=0}^{c-1} (M_{n,i}(x, y) M_{r,i}(x, y))}{\sum_{i=1}^6 \sum_{x=0}^{r-1} \sum_{y=0}^{c-1} M_{r,i}(x, y)}, \quad (7)$$

where  $i$  stands for different occurrence mask in both  $M_n$  and  $M_r$ . Thirdly, for  $R_r$  and  $R_n$ , we calculate absolute difference ratios by (8) and mark them as  $V_{er}$ .

$$V_{er} = \text{abs}(R_n - R_r). \quad (8)$$

Finally, we concatenate  $V_d$ ,  $V_{rr}$ , and  $V_{er}$  into a 22-dimensional vector forming the similarity features as input of SVM for training and classification. This 22-dimensional vector is designed with the aim of avoiding background pixels and providing robust representativeness for classification.

**3.3. Classification Metric Learning.** In classification metric learning, we use 435 pedestrian images, which are automatically extracted from ViSOR [22] surveillance video sequences, to learn interclass variations between 2 pedestrian images of 2 individuals and intraclass variations between 2 pedestrian images of the same individual. We propose a Pure-vs-Exception strategy to separate these 435 pedestrian images into 4 groups. Three of them are Pure Groups, only containing 3 different individuals with each individual corresponding to 1 specific Pure Group. The last group, called Exception Group, contains pedestrian images of individuals different from those individuals in Pure Groups. Pure Groups have 130, 134, and 136 pedestrian images, respectively, while the Exception Group has 35 pedestrian images of 21 individuals. Positive training samples are generated from every combination of 2 images inside each Pure Group, considering combination order which matters for reference image and test image. So, Pure Groups have 16,900 ( $130^2$ ), 17,956 ( $134^2$ ), and 18,496 ( $136^2$ ) positive training samples, respectively, adding up to 53,352. Meanwhile, negative training samples are generated from image pairs where one is from one of the Pure Groups and the other is from any other group, resulting in an amount of 67,324. The number 67,324 comes from  $(130 * (134 + 136 + 35) + 134 * (136 + 35) + 136 * 35)$ . Thus, we generate a total of 120,676 labeled samples for training from these 435 pedestrian images.

With these training samples, we explore *CDPM*'s performance using the SVM provided by [3] under different kernels and different kernel degrees, while the other parameters of SVM are set as default. The kernels include linear kernel (Lin.), polynomial kernel (Poly.), radial basis kernel (Rad.), and sigmoid kernel (Sig.). The kernel degree varies from 1 to 5, which only works in polynomial kernel. We train the SVMs with different kernels and degrees on these 120,676 labeled samples.

After training, we also use 200 pedestrian images (with 50 individuals and 4 pedestrian images of different viewpoints for each individual) in IPR dataset to generate 20,400 labeled samples with 800 positives and 19,600 negatives for the classification test. Table 1 shows the test results of the trained *CDPM* methods. In Table 1, AC means overall accuracy on test samples, while PM and NM represent the false prediction rates inside positive samples and negative samples, respectively. The highest accuracies are highlighted in bold type. As the kernel degree varies from 1 to 5, test accuracy of polynomial *CDPM* descends from 94.89% to 91.93%, and test accuracy declines to 81.57% when the kernel degree reaches 10 in further tests. The classification results of all 4 kernels, where polynomial kernel's degree is 1, are very close, showing the stability of our *CDP* feature model. Even though the test samples are unbalanced on the distribution of positives and negatives, PMs and NMs are both below 9%, and the differences between them are less than 6%. This shows great and balanced classification performance of *CDPM* methods on both positives and negatives. However, it is hard to determine which kernel is better in the Re-ID ranking problem from Table 1 without any further experiments.

TABLE 1: Classification accuracies of *CDPM* methods.

Degree	1			2			3			4			5		
	AC	PM	NM	AC	PM	NM	AC	PM	NM	AC	PM	NM	AC	PM	NM
CDPM <sub>l</sub> (Lin.)	<b>95.31</b>	8.25	4.54	-	-	-	-	-	-	-	-	-	-	-	-
CDPM <sub>p</sub> (Poly.)	94.89	5.88	5.08	94.43	5.25	5.59	93.78	4.25	6.3	93.08	3.25	7.07	91.93	2.5	8.3
CDPM <sub>r</sub> (Rad.)	94.82	6.88	5.11	-	-	-	-	-	-	-	-	-	-	-	-
CDPM <sub>s</sub> (Sig.)	94.81	6	5.15	-	-	-	-	-	-	-	-	-	-	-	-



FIGURE 3: Sample pedestrian images from 3DPeS.

Thus, in further experiments, we use these *CDPM* methods with 4 kernel types (where the kernel degree for polynomial kernel is 1) to evaluate our *CDP* feature models capabilities and compare them with other state-of-the-art methods on 2 public datasets: 3DPeS dataset [21] and IPR dataset [18].

#### 4. Experiment

In this section, we evaluate our *CDPM* method on 2 public datasets: 3DPeS dataset [21] and IPR dataset [18]. These 2 datasets are based on real surveillance setup of University of Modena and Reggio Emilia campus. Surveillance setup contains 8 cameras at different zoom levels with a recording resolution of  $704 \times 576$  pixels. 3DPeS and IPR are 2 of the few person Re-ID datasets that provide silhouette images required by our *CDPM* method, while the others, like VIPeR [23], iLIDS [24], and CAVIAR [25], do not provide silhouette images. Note that, due to implementation limits, 3DPeS and IPR datasets provide silhouette images that filter out most background pixels rather than all background pixels. In addition, IPR dataset is better segmented and smaller than 3DPeS, while some silhouette images in 3DPeS contain shadow areas caused by sunlight.

Here, we present averaged comparison results with the widely used cumulative match characteristic (CMC) and CMC curves. We also report the proportion of uncertainty removed (PUR) value proposed in [16]:

$$\text{PUR} = \frac{\log(S) + \sum_r^S M(r) \log(M(r) + e)}{\log(S)}, \quad (9)$$

where  $S$  is the size of test data,  $M(r)$  is match characteristic instead of cumulative match characteristic, and  $e$  is an extremely small constant to avoid meaningless definition in logarithm when  $M(r)$  is 0. In our experiment,  $e$  is set to  $1e-18$ , which still preserves 16 precisions after decimal point

for  $S = 95$ . The PUR value evaluates the entropy information of output rank and is invariant to the logarithm base used. Additionally, PUR value ranges from 0 to 1. A larger PUR value indicates that more values are concentrated in fewer  $M(r)$ s since all  $M(r)$ s add up to 1. For example, assume that  $M(1)$  is 1 and the remaining  $M(r)$ 's are all 0, and PUR value is 1 according to (9). When all  $M(r)$ 's are equal, PUR value is 0. As mentioned in [16], PUR value describes the behaviors of all  $M(r)$ 's in the entire rank instead of the behavior of any single  $M(r)$ . To obtain a precise evaluation of different ranks performances, we must combine PUR scores with CMC scores during assessment. This is exceptionally helpful for evaluating close CMC scores at different ranks.

The following subsections explain the details of experimental settings and results.

##### 4.1. Experiment on 3DPeS Dataset

**4.1.1. Dataset and Experimental Protocol.** 3DPeS [21] dataset provides 1,011 snapshots of 192 individuals with both appearance images and silhouette images. These 1,011 pedestrian images contain significant pose viewpoint variations, as well as illumination changes. Examples are shown in Figure 3.

We adopt the single-shot experiment setting [6] to randomly divide 3DPeS dataset into 2 subsets with  $P$  individuals in test subset, where  $P$  is set to 95. In the test subset, we randomly choose 1 image of each individual to form the reference set and randomly choose another image of each individual to form the test set. We repeat this partition 10 times to generate 10 groups of test data for evaluation.

**4.1.2. Features and Implementation Details.** In our experiment, we loosely crop out background areas, as shown in Figure 4, instead of scaling pedestrian images into a fixed resolution like [5–7, 12].

We employ 4 different kernels, linear kernel, polynomial kernel with kernel degree of 1, radial basis kernel, and sigmoid



FIGURE 4: Cropping out background extension area.

kernel in our *CDPM*. Thus, there are 4 corresponding *CDPM* methods: *CDPM<sub>l</sub>*, *CDPM<sub>p</sub>*, *CDPM<sub>r</sub>*, and *CDPM<sub>s</sub>*, in which the postfixes *l*, *p*, *r*, and *s* represent linear kernel, polynomial kernel, radial basis kernel, and sigmoid kernel, respectively. Our *CDP* feature model extracts  $32 \times 32$  2D histogram matrices from 6 channels of Lab and HSV color spaces based on 2 body parts, as well as  $32 \times 32$  occurrence mask matrices, plus a 7-bin occurrence ratio. We project their concatenation into a 22-dimensional feature vector.

In *cAMT-DCA* [12], color, LBP, and HOG features are extracted from 75 overlapping  $16 \times 16$  blocks. Block step sizes in both horizontal and vertical directions are 8 pixels. The color histogram is a 16-bin histogram from 8 color channels of RGB, YCbCr, and HS. For each block, a 484-dimensional vector is generated, including color, LBP, and HOG features. Since there are 75 blocks in each pedestrian image, a 36,300-dimensional vector is generated, which is then compressed into a 100-dimensional vector by PCA before being applied in *cAMT-DCA*.

*DSTML* [7] uses features extracted from 6 nonoverlapping horizontal stripes of each pedestrian image. Features consist of 16-bin histograms from 8 channels of RGB, YUV, and HS channels, as well as uniform LBP histograms with 8 neighbors and 16 neighbors, respectively. Then, histograms of each channel are normalized and concatenated into a 2,580-dimensional feature vector. Before being applied in *DSTML*, PCA learned from source data is used to project the target feature vector into a low-dimensional feature vector.

*PCCA* [26], *LDFA* [16], *SVMML* [27], *KISSME* [10], *rPCCA* [6], *kLFDA* [6], and *MFA* [6] use the same set of features. The features used in *PCCA* are extracted from  $32 \times 32$  overlapping blocks with a step size of 16 pixels in both horizontal and vertical directions, while the features used in the other 6 methods are extracted from  $16 \times 16$  overlapping blocks with a step size of 8 pixels in both horizontal and

vertical directions. Features, containing 16-bin histograms of 8 color channels (RGB, YUV, and HS) and 2 kinds of uniform LBP histograms with 8 neighbors and 16 neighbors, are extracted from these blocks. Then, histograms are normalized and concatenated into a feature vector. In *PCCA*, *rPCCA*, *LFDA*, and *kLFDA*, the concatenated feature vector space is projected into 40-dimensional space, while it is projected into 30-dimensional space in *MFA* and 70-dimensional space in *KISSME*.

*REV* [6] conducts ranking ensemble voting of *rPCCA*, *kLFDA*, and *MFA* by adding rankings in a simple voting scheme.

The first 10 methods in Table 2 use features extracted over dense blocks from resized pedestrian images with a resolution of  $128 \times 48$  pixels. In the parentheses of first row in Table 2, the first parameter in numeric form indicates the region size of pedestrian image when extracting features, while the second parameter indicates the kernel type in which  $R_{x^2}$  means *RBF*- $x^2$  kernel. The parameter settings of these 10 methods are the best settings reported in [6, 7, 12].

Note that our training setting is different from these other methods. We use a training set extracted from *ViSOR* surveillance video sequences, which is the source of *3DPeS* dataset. Our training set only contains 435 pedestrian images of 24 individuals. Among these 435 images, there are 400 pedestrian images coming from 3 individuals in the Pure Groups. In *cAMT-DCA*, the corresponding training set contains all images in the source dataset (*VIPEr*, 1,264 pedestrian images of 632 individuals) and the images of the other 97 individuals from *3DPeS* dataset. *DSTMLs* corresponding training set consists of the label information of source dataset *CAVIAR*, with 1,220 pedestrian images of 72 individuals. The other methods mentioned in this section all use the rest 97 individuals in *3DPeS* dataset as training set, with around half of 1,011 pedestrian images. Thus, compared to those methods,

TABLE 2: PUR and CMC scores on 3DPeS dataset with  $P = 95$  test individuals.

$r$	cAMT-DCA	DSTML	SVMML (75)	PCCA (14, $R_{x^2}$ )	LFDA (75)	KISSME (75)	rPCCA (75, $x^2$ )	kLFDA (75, $x^2$ )	MFA (75, $x^2$ )	REV	CDPML	CDPMP	CDPMr	CDPMs
1	31.9	32.5	34.7	42.2	45.5	41.3	47.3	54.0	48.4	54.2	67.9	<b>68.8</b>	68.6	68.7
5	53.5	54.3	66.4	71.1	69.2	66.2	75.0	77.7	72.4	77.7	79.9	<b>80.5</b>	80.0	<b>80.5</b>
10	63.9	65.3	78.8	82.1	78.0	76.3	84.5	85.9	81.5	<b>86.1</b>	85.2	85.7	85.1	85.7
20	75.1	N/A	88.5	90.5	86.1	85.3	91.9	92.4	89.8	<b>92.8</b>	92.3	<b>92.8</b>	92.1	91.8
PUR	N/A	N/A	9.7	45.1	43.2	40.1	49.3	53.5	47.6	53.8	62.3	<b>63.6</b>	62.9	63.2

our training set is the smallest and hardest with least number in both images and individuals.

**4.1.3. Experimental Results.** Table 2 and Figure 5 show the average results of our 4 CDPM methods and several other methods. The highest CMC scores at every rank and highest PUR value are highlighted in bold type. For fair comparison, we use the results provided by the authors or by corresponding cited papers, under the single-shot experiment setting with 95 test individuals. Results in the first and second columns in Table 2 are from Wang et al. [12] and Hu et al. [7], respectively, and results of the third to tenth columns come from the fantastic work reported in literature [6]. These results are the best reported results of 3DPeS in the existing literatures [6, 7, 12, 16, 21, 28]. Table 2 shows that all 4 of our CDPM methods outperform the other methods at rank 1 and rank 5 CMC scores, achieving at least 13.7% and 2.2% improvement, respectively. Meanwhile, our CDPMP achieves 14.6% and 2.8% improvement at rank 1 and rank 5 CMC scores, respectively. At rank 10 CMC score, REV performs best with 0.4% to 1% improvement compared to our 4 methods. As for rank 20 CMC score, our CDPM and REV achieve the highest figure, 92.8%. Note that REV is a fusion solution of kLFDA, rPCCA, and MFA and is more complicated and computational than any of them. Although REV is slightly better than CDPMP in rank 10 CMC score, our methods have higher PUR scores than those of methods reported by Xiong et al. [6]. Among PUR scores, CDPMP possesses the highest score, with at least 9.8% improvement over the PUR scores reported in the other 8 methods. Under the consideration of both CMC and PUR, our CDPMP has the best performance among all methods in Table 2.

In Figure 5, our methods show great advantage with rank 1 and rank 2 matching accuracies. The other 10 methods extracted features from dense overlapping regions containing both background and person information and do not alter their feature selection strategies automatically according to different poses and different body types between test and reference pedestrian images. This means they are unable to properly filter out background features in matching phase when significant pose and body type variations occur. Our methods possess great tolerance for different poses and body types because our CDP feature model has already filtered out most background pixels with silhouette images before features extractions. Furthermore, it projects the extracted features into a uniform space for similarity matching. The slopes of our methods in Figure 5 decrease faster than those of other methods from rank 2 to rank 10, which means the

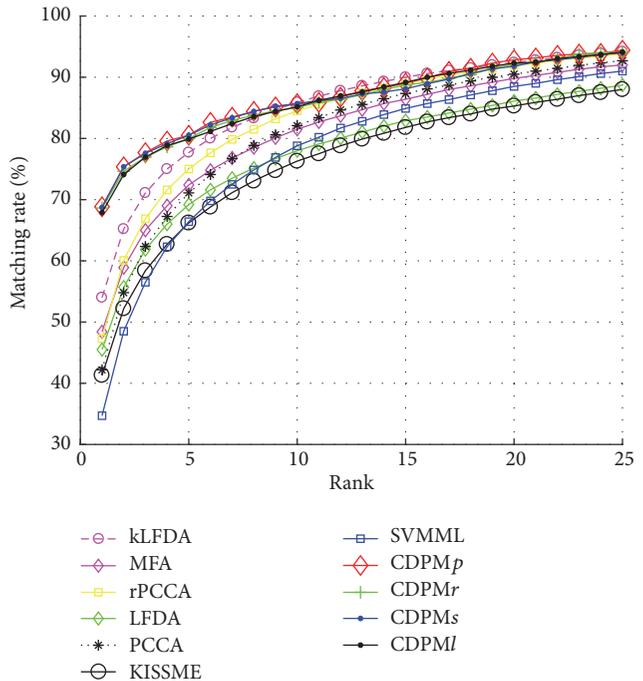


FIGURE 5: CMC curves on 3DPeS dataset.

growth of matching accuracies of our methods in top 10 ranks is smaller than those of the other methods. Normally, the slope of CMC curve decreases faster when the rank 1 accuracy is higher due to smaller growth of CMC scores in subsequent ranks. Despite lack of competitiveness after rank 5, our 4 methods remain in top ones, still holding some of the best reported results.

The cAMT-DCA and DSTML methods are designed under the circumstance of learning a similarity classification metric for the target scenario from another existing annotated dataset, since manually annotating a large dataset of pairwise pedestrian images is costly and impractical for efficiently deploying a Re-ID system to a completely new scenario [12]. Even though the results of these 2 methods are the lowest at rank 1, rank 5, and rank 10, they are still good trials in transfer metric learning.

Our CDPM provides a practical alternative to automatically collecting pedestrian images from video sequences and uses Pure-vs-Exception strategy to group these images for similarity classification metric learning. The experimental results of our methods on 3DPeS dataset show the effectiveness of our similarity classification metric learning.

TABLE 3: PUR and CMC scores on IPR dataset with  $P = 25$  test individuals.

$r$	Reference or model: 3 images for each individual					Reference: 2 images for each individual				Reference: 1 image for each individual			
	SARC3D	CDPMI	CDPM $p$	CDPM $r$	CDPM $s$	CDPMI	CDPM $p$	CDPM $r$	CDPM $s$	CDPMI	CDPM $p$	CDPM $r$	CDPM $s$
1	76	87.5	<b>89.5</b>	88	89	84.3	<b>86.8</b>	86.2	86.3	76.8	79.7	79	<b>79.8</b>
2	85	<b>95</b>	<b>95</b>	94.5	<b>95</b>	92.7	<b>94.3</b>	94	93.8	88.8	<b>90.8</b>	89.8	<b>90.8</b>
5	93	<b>99.5</b>	<b>99.5</b>	<b>99.5</b>	<b>99.5</b>	98.2	<b>98.3</b>	98.2	<b>98.3</b>	97.5	98.2	97.7	<b>98.3</b>
10	98 <sup>a</sup>	<b>99.5</b>	<b>99.5</b>	<b>99.5</b>	<b>99.5</b>	98.8	99.2	98.8	<b>99.3</b>	98.7	<b>98.8</b>	98	98.7
PUR	68.8 <sup>b</sup>	84.0	<b>85.9</b>	84.1	85.4	79.6	<b>81.8</b>	81.3	81.7	72.2	75	74.1	<b>75.5</b>

<sup>a</sup>The accuracy of rank 10 in SARC3D is read from Figure 6 in [18]. <sup>b</sup>The PUR score is calculated using figures read from Figure 6 in [18].

Our 4 methods, *CDPMI*, *CDPM $p$* , *CDPM $r$* , and *CDPM $s$* , corresponding to 4 different kernels had close results on 3DPeS dataset. Although *CDPM $p$*  and *CDPM $s$*  are very close, *CDPM $p$*  is slightly better than *CDPM $s$* . *CDPM $r$*  is third best, and *CDPMI* is last. Despite these rankings, such close results are due to more precise and stable features provided in our *CDP* feature model by filtering out most background information.

#### 4.2. Experiment on IPR Dataset

**4.2.1. Dataset and Experimental Protocol.** IPR dataset is smaller than 3DPeS dataset and does not contain many illumination variations among images of the same individual. However, it is a suitable dataset to evaluate the robustness of viewpoint variations due to its specific settings. IPR dataset is a specialized dataset that contains 50 individuals with 4 specific images for each individual, resulting in a total of 200 pedestrian images. There are 4 different viewpoints for each individual, as shown in Figure 6: front view, back view, left view, and right view. These kinds of variations are challenging for methods based on dense blocks over fixed resolutions because of their asymmetric and nonuniform distribution of body. Furthermore, due to its specialization and small quantity of images, IPR dataset is not well known to the public. Thus, we only compared our *CDPM* method with SARC3D [18], which is intrinsically independent of viewpoint variations as mentioned in [18], and evaluated *CDPM*'s capability of dealing with these viewpoint variations.

We employ 3 different test settings with 1, 2, and 3 images of each individual to form the reference set but only 1 image of each individual to form the test set. We replicate 24, 24, and 8 partitions for these 3 experimental settings, respectively, by choosing different images forming different reference and test sets. Half partitions come from the first 25 individuals, and the other half come from the last 25 individuals. The reference set with 3 images is the easiest setting for testing, while the reference set with 1 image is the hardest for testing. Since SARC3D method requires 3 images to generate feature model, SARC3D is only evaluated in the reference setting with 3 images. Conversely, our methods are presented in all 3 reference settings.

**4.2.2. Features and Implementation Details.** We evaluate *CDPMI*, *CDPM $p$* , *CDPM $r$* , and *CDPM $s$*  with the same model setting in 3DPeS on IPR dataset, since IPR dataset is also generated from ViSOR sequences.



FIGURE 6: Sample images from IPR dataset. Rows from up to bottom are front views, back views, left views, and right views, respectively.

SARC3D method uses 3 images to generate features through a graphical 3D body model with a selected set of vertices. Note that the 3D body model is learned prior to extract feature, which is an exclusive and advantageous setting. SARC3D extracts a 16-bin color histogram from HSV with 8 bins for H channel and 4 bins each for S and V channels, over  $10 \times 10$  pixel blocks for each vertex. Besides color histogram, location, mean color, optical reliability, and saliency are also extracted for each vertex. Then, global similarity distance  $D_H$  and local similarity distance  $D_S$  are computed from weighted average of vertex-wise distances by using different weighting strategies. During matching, the final distance measure  $D_{HS}$  is computed by the product of  $D_H$  and  $D_S$ .

**4.2.3. Experimental Results.** Averaged results are shown in Table 3 and Figure 7. The highest CMC and PUR scores

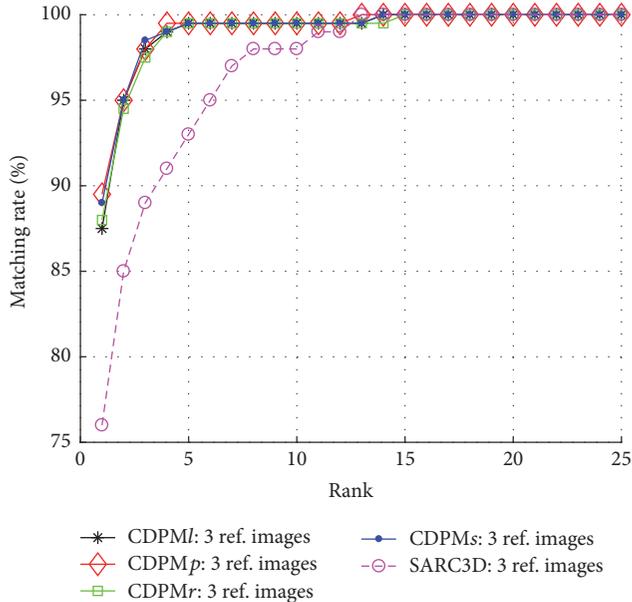


FIGURE 7: CMC curves on IPR dataset.

of each reference setting are highlighted in bold type in Table 3. For better observation, we only draw results of the first reference setting in Figure 7. With less test individuals and less illuminations variations, CMC scores are much higher on IPR dataset than those on 3DPeS dataset. However, IPR dataset still possesses challenges on viewpoint and pose variations, as shown in Figure 6.

In the first reference setting of Table 3, SARC3D’s results represent the best setting in [18], and PUR score is calculated using reading figures with maximum error less than 0.2%. All 4 of our methods outperform SARC3D. In particular, *CDPMp* achieves 13.5%, 10%, 6.5%, and 1.5% improvements at rank 1, rank 2, rank 5, and rank 10 CMC scores, respectively, while PUR score improvement reaches 17.1%. The higher CMC and PUR scores of our *CDPM* methods on this reference setting can be explained in 3 aspects. Firstly, test set is small with only 25 individuals, and pedestrian images are well segmented with nearly no background information. Meanwhile, there is no illumination variance among the pedestrian images of the same individual. Secondly, our *CDPM* methods are free from background influences caused by pose and body type variations, especially for the well-segmented pedestrian images. Thirdly, our *CDPM* methods use similarity scores computed from 3 reference images to evaluate the final similarity measure for classification.

In the second reference setting, compared to SARC3D, improvements of *CDPMp* at rank 1, rank 2, rank 5, and rank 10 CMC scores are 10.8%, 9.3%, 5.3%, and 1.2%, respectively, and PUR score improvement is 13%. In the third reference setting, improvements of *CDPMs* are 3.8%, 5.8%, 5.3%, 0.7%, and 6.7%, respectively. All our methods in these 2 reference settings outperform SARC3D in the first reference setting according to CMC and PUR scores. The results on these two harder settings further prove that our *CDPM* methods have better tolerance for pose and viewpoint variations in IPR dataset than SARC3D. Furthermore, our experimental results

prove the effectiveness of our metric learning strategy under the same domain.

Through these 3 reference settings, our experimental results show that our *CDPM* methods achieve significant improvements with multireference images. The behaviors of our 4 methods on IPR dataset are very similar to those on 3DPeS dataset. But the lower number of test individuals, well-segmented silhouette images, and multiple reference images not only make our 4 methods more different at rank 1 CMC scores and PUR scores but also swap the rankings of *CDPMs* and *CDPMp* in the hardest setting (with 1 reference image).

**4.3. Analysis across Tables.** Test results from Table 1 in distance metric learning are subject to a classification problem, while the experiment results from Tables 2 and 3 on 3DPeS and IPR datasets are subject to a ranking problem. Thus, the rankings of our 4 methods in Table 1 are different from those in Tables 2 and 3. However, there is a link between Table 1 and the other tables. From the first column in Table 1, the sums of PM and NM for *CDPMI*, *CDPMp*, *CDPMr*, and *CDPMs* are 12.79%, 10.96%, 11.99%, and 11.15%, respectively, and the corresponding mean variances of PM and NM are 3.44, 0.16, 0.78, and 0.18 with a scale of  $10^{-4}$ , respectively. The ascending orders of the sums and mean variances are consistent with the rankings on 3DPeS and IPR datasets. Thus, under identical or very close ACs, *CDPM* tended to perform better on the Re-ID task with both less sum and less mean variance of PM and NM. This explains why *CDPMp* had better performance than the other 3 *CDPM* methods. Additionally, the close ACs, sums, and mean variances of PN and NM between *CDPMp* and *CDPMs* explain the close results between *CDPMp* and *CDPMs* on 3DPeS and IPR datasets.

## 5. Conclusion

We have proposed a *CDP* feature model and evaluated the performances of our Re-ID *CDPM* methods on 3DPeS and IPR datasets. Experimental results show that our *CDPM* methods have better performance compared to other state-of-the-art approaches and possess great tolerance for pose and viewpoint variations. In addition, we provide an effective similarity classification metric learning strategy for our *CDP* feature model to maximize interclass differences and suppress intra-class differences. Although our *CDP* feature model relies on silhouette images, which can be automatically extracted from video sequences or provided, it is compatible with varied image resolutions and is free of resizing distortion.

However, due to strong reliance on silhouette images, unfiltered background pixels can directly affect the classification results in our *CDPM* method. Thus, we intend to improve segmentations of the silhouette images provided or autoextracted, by filtering out more background pixels with less loss of person pixels in the future. In particular, we would like to automatically filter out shadow pixels cast by sunlight. In addition, we will also focus on finding integrable texture features and enriching the feature dimension of *CDP* to enhance robustness for significant illumination variations. By applying our method to surveillance systems in future work, we will find a more reliable matching strategy to exploit

CDP features in the similarity classification metric learning and multireference matching.

## Conflicts of Interest

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

## References

- [1] R. Vezzani, D. Baltieri, and R. Cucchiara, "People reidentification in surveillance and forensics: a survey," *ACM Computing Surveys*, vol. 46, no. 2, 2013.
- [2] Z. Sheng, H. Wang, C. Yin, X. Hu, S. Yang, and V. C. M. Leung, "Lightweight Management of Resource-Constrained Sensor Devices in Internet of Things," *IEEE Internet of Things Journal*, vol. 2, no. 5, pp. 402–411, 2015.
- [3] C. Chih-Chung and L. Chih-Jen, "LIBSVM: a Library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, 2011.
- [4] L. Michel, P. Marc, and B. Robert, "Vip: Vision tool for comparing images of people," in *Proceedings of the 16th International Conference on Vision Interface*, pp. 35–42, June 2003.
- [5] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, "Person re-identification by Local Maximal Occurrence representation and metric learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 2197–2206, Mass, USA, June 2015.
- [6] F. Xiong, M. Gou, O. Camps, and M. Sznai, "Person re-identification using kernel-based metric learning methods," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 8695, no. 7, pp. 1–16, 2014.
- [7] J. Hu, J. Lu, Y.-P. Tan, and J. Zhou, "Deep transfer metric learning," *IEEE Transactions on Image Processing*, vol. 25, no. 12, pp. 5576–5588, 2016.
- [8] Z. Wang, R. Hu, C. Liang et al., "Zero-shot person re-identification via cross-view consistency," *IEEE Transactions on Multimedia*, vol. 18, no. 2, pp. 260–272, 2016.
- [9] L. Ma, X. Yang, and D. Tao, "Person re-identification over camera networks using multi-task distance metric learning," *IEEE Transactions on Image Processing*, vol. 23, no. 8, pp. 3656–3670, 2014.
- [10] D. Tao, L. Jin, Y. Wang, Y. Yuan, and X. Li, "Person re-identification by regularized smoothing kiss metric learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 10, pp. 1675–1685, 2013.
- [11] L. An, M. Kafai, S. Yang, and B. Bhanu, "Person Reidentification with Reference Descriptor," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 4, pp. 776–787, 2016.
- [12] X. Wang, W.-S. Zheng, X. Li, and J. Zhang, "Cross-scenario transfer person reidentification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 8, pp. 1447–1460, 2016.
- [13] Z. Shi, T. M. Hospedales, and T. Xiang, "Transferring a semantic representation for person re-identification and search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 4184–4193, USA, June 2015.
- [14] E. Ahmed, M. Jones, and T. K. Marks, "An improved deep learning architecture for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 3908–3916, USA, June 2015.
- [15] M. Köstinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 2288–2295, June 2012.
- [16] S. Pedagadi, J. Orwell, S. Velastin, and B. Boghossian, "Local fisher discriminant analysis for pedestrian re-identification," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2013*, pp. 3318–3325, USA, June 2013.
- [17] R. Zhang, L. Lin, R. Zhang, W. Zuo, and L. Zhang, "Bit-scalable deep hashing with regularized similarity learning for image retrieval and person re-identification," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 4766–4779, 2015.
- [18] D. Baltieri, R. Vezzani, and R. Cucchiara, "SARC3D: a new 3D body model for people tracking and re-identification," in *Proceedings of the 16th Conference on Analysis and Processing: Part I (ICIAP '11)*, vol. 6978 of *Lecture Notes in Computer Science*, pp. 197–206, Springer, Heidelberg, Germany, 2011.
- [19] J. Chen, Y. Wang, and Y. Y. Tang, "Person Re-identification by Exploiting Spatio-Temporal Cues and Multi-view Metric Learning," *IEEE Signal Processing Letters*, vol. 23, no. 7, pp. 998–1002, 2016.
- [20] D. Chen, Z. Yuan, G. Hua, N. Zheng, and J. Wang, "Similarity learning on an explicit polynomial kernel feature map for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 1565–1573, USA, June 2015.
- [21] D. Baltieri, R. Vezzani, and R. Cucchiara, "3dps: 3d people dataset for surveillance and forensics," in *Proceedings of the the 2011 joint ACM workshop on Human Gesture and Behavior Understanding (J-HGBU '11)*, pp. 59–64, NY, USA, December 2011.
- [22] <http://imagelab.ing.unimore.it/visor/index.asp>.
- [23] D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features," in *Proceedings of the 10th European Conference on Computer Vision: Part I (ECCV '08)*, vol. 5302 of *Lecture Notes in Computer Science*, pp. 262–275, Springer, Berlin, Germany, 2008.
- [24] W. S. Zheng, S. Gong, and T. Xiang, "Associating groups of people," in *Proceedings of the 20th British Machine Vision Conference (BMVC'09)*, September 2009.
- [25] D. C. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino, "Custom pictorial structures for re-identification," in *Proceedings of the 22nd British Machine Vision Conference, BMVC'11*, pp. 1–11, September 2011.
- [26] A. Mignon and F. Jurie, "PCCA: a new approach for distance learning from sparse pairwise constraints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 2666–2672, June 2012.
- [27] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith, "Learning locally-adaptive decision functions for person verification," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2013*, pp. 3610–3617, USA, June 2013.
- [28] S. Paisitkriangkrai, C. Shen, and A. Van Den Hengel, "Learning to rank in person re-identification with metric ensembles," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 1846–1855, USA, June 2015.

## Research Article

# Optimal Channel Selection Based on Online Decision and Offline Learning in Multichannel Wireless Sensor Networks

**Mu Qiao, Haitao Zhao, Shengchun Huang, Li Zhou, and Shan Wang**

*College of Electronic Science, National University of Defense Technology, Changsha, China*

Correspondence should be addressed to Mu Qiao; [qiaomu16@nudt.edu.cn](mailto:qiaomu16@nudt.edu.cn)

Received 31 August 2017; Accepted 15 November 2017; Published 13 December 2017

Academic Editor: Kun Bai

Copyright © 2017 Mu Qiao 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.

We propose a channel selection strategy with hybrid architecture, which combines the centralized method and the distributed method to alleviate the overhead of access point and at the same time provide more flexibility in network deployment. By this architecture, we make use of game theory and reinforcement learning to fulfill the optimal channel selection under different communication scenarios. Particularly, when the network can satisfy the requirements of energy and computational costs, the online decision algorithm based on noncooperative game can help each individual sensor node immediately select the optimal channel. Alternatively, when the network cannot satisfy the requirements of energy and computational costs, the offline learning algorithm based on reinforcement learning can help each individual sensor node to learn from its experience and iteratively adjust its behavior toward the expected target. Extensive simulation results validate the effectiveness of our proposal and also prove that higher system throughput can be achieved by our channel selection strategy over the conventional off-policy channel selection approaches.

## 1. Introduction

Multichannel communication enables terminals to transmit on different channels simultaneously without mutual interferences. It has been widely used in Wireless Sensor Networks (WSNs) or Internet of Things (IoT) to support large and dense networks [1]. Since different channels may result in different transmission qualities, the channel selection plays crucial role in multichannel WSNs.

Owing to the constraints of energy budget and memory size of WSN nodes, centralized approaches are usually considered to conduct channel selection. In these approaches, a central node, for example, access point (AP) or sink node, performs all the necessary computations and informs reasonable channel selection decision to other sensor nodes. Wu et al. [2] adopt a static tree-based channel selection approach where the sink node can operate on attribute sensor node to switch to a channel with minimum interference. Li et al. [3] extend the typical two-level architecture by using aggregator coordinate associated with sensor nodes to avoid the transmission of huge collected data. However,

centralized approaches have limited performances in large-scale networks. Therefore, the distributed approaches have attracted more interests, since they allow better flexibility and scalability in node deployment. Tang et al. [4] design a counter-based approach in which nodes select channels based on the channel quality. Nevertheless, information exchange and negotiation among nodes in this approach require tight synchronization.

In order to implement self-decision and self-learning, the approaches based on game theory and reinforcement learning have been introduced to improve channel selection or other resource allocation problems, for example, [5–11]. Game theory has been a powerful tool to model decentralized networks to obtain an equilibrium state. Its common drawbacks lie in the huge instant computational costs. Félegyházi et al. [5] define two-tier noncooperative medium access game which composes of a channel allocation and a multiple access subgames. Han and Kawanishi [6] provide two types of game strategies to adapt to different collision probabilities. Canzian et al. [7] design an equilibrium game between the pricing and intervention to achieve the maximum efficiency

in the perfect monitoring scenario. On the other hand, reinforcement learning can be used to help each individual node learn from a sequence of their individual feedback history and adjust their behaviors towards expected state, gradually. Nie and Haykin [8] provide a classical reinforcement learning framework to solve the channel assignment problem. Naddafzadeh-Shirazi et al. [9] and Zhou et al. [10] investigate reinforcement learning schemes to help secondary users capture the state of the primary user and learn the satisfactory feedback to improve its own utility. Zame et al. [11] design a statistical count learning scheme to make secondary stations learn from and coordinate their own histories, while simultaneously teaching other stations about these histories of counter. Unfortunately, reinforcement learning approaches have a common disadvantage that they usually require lots of learning iterations to converge to an acceptable solution. Furthermore, most existing work is based on information exchange and negotiation among users, which may cause computational complexity and communication overhead.

In this paper, we propose an intelligent channel selection strategy with hybrid architecture which benefits from the combination of centralized methods and distributed method. Our work requires neither central control nor any exchange or negotiation messages among sensors. Most importantly, we make use of the intelligent technique, for example, game theory and reinforcement learning, to find a solution to the application limitation problem of optimal channel selection with different communication overhead. To achieve this goal, we formulate two algorithms from the perspective of sensors which, respectively, named online decision algorithm and offline learning algorithm. We consider the proposed online strategy and offline strategy have their own merits and are targeted at different application scenarios.

The online decision algorithm based on noncooperative game is to help each individual sensor immediately select optimal channel when the network can satisfy the requirements of energy and computational costs. In terms of the computational complexity, the online strategy is based on the noncooperative game and is less complex than the cooperative game. The “online” here means real-time computation that sensors can obtain immediate results. This approach is focused on how to find the optimal equilibrium state through local computation by each individual sensor.

The offline learning algorithm based on reinforcement learning is to address the iterative channel selection to decrease energy consumption. Each sensor can learn from a sequence of its individual feedback history and adjust its behavior towards the expected target. This approach emphasizes the learning ability that sensor learns its behavior and picks optimal choices while converging to an acceptable and stable solution. Different from the online decision, the offline learning algorithm cannot affect node’s selection behavior immediately, but in an iterative way. Therefore, the main contributions of this paper are twofold:

- (i) We propose a hybrid architecture, in which centralized processing and distributed processing are jointly considered in order to alleviate overhead of AP node and allow more flexible deployment.

- (ii) In this architecture, we present two types of optimal channel selection algorithm based on intelligent decision and learning. They can be used to adapt to different requirements of the communication overhead. It requires no central control, no information exchange, or negotiation among individual nodes, which allows low computational complexity, communication overhead, and storage requirement.

The rest of this paper is organized as follows. Section 2 briefly reviews related work. Section 3 introduces the system model. In Sections 4 and 5, the online decision and offline learning method for channel selection are, respectively, explained in detail. Section 6 validates our proposal via simulation. Section 7 concludes this research.

## 2. Related Work

Intelligent methods have been extensively investigated on the channel selection problem in the past decade. They are different from conventional channel selection strategies by taking advantage of the sense information around external environment to make a decision in the channel selection process. The intelligent methods can be broadly divided into two categories: the game-based category that focuses on applying game theoretical tools to make a real-time decision [5–7, 12–23] and the learning-based category that focuses on introducing reinforcement learning techniques to select optimized resource [8–11, 24–31].

For the game-based category, a host of game strategies are presented to fit the optimal channel selection process, including cooperative game and noncooperative game solutions. The cooperative game can improve the performance of resource allocation protocol, while it needs more information exchange and negotiation among nodes, which incurs high communication overhead and computational complexity. Nuggehalli et al. in [12] exploit AP node to manage the priority of other nodes, which guarantees the fairness of the bargaining process. This thread is enhanced by penalizing and pricing mechanisms [13–16]. In specific, Shrestha et al. [13] and Chatterjee and Wong [14] investigate punishment mechanism to promote cooperation. Wang et al. [15] and Cui et al. [16] propose a pricing model based on Stackelberg game, in which the leader node formulates price list for follower node to access certain desirable channel. Recently, a new incentive scheme, called intervention, has gained utilization in [17–19]. These approaches aim at formulating several incentive mechanisms to achieve higher utility. However, these methods need designers or coordinators to price and formulate reasonable rules during the initial phase. Nodes should perform strictly with the rules during the execution phase, which brings an inevitable problem for dynamic network scenarios that designers need to monitor and adjust rules constantly. All of the aforementioned works depend on a centralized server to solve the resource allocation issue and inform the decision to each individual node. However, in many cases, the synchronization information may not be available for all nodes, and some nodes may deviate from cooperation. The second thread in this category uses

noncooperative game-based algorithms and policies for the distributed scenario, in which each node makes real-time decisions considering only to maximize its individual utility. Cho and Tobagi [20] indicate that the noncooperative game has lower computational complexity than cooperative game. The work suggests that noncooperative system converges the individual benefits with an appropriate selfish strategy that can lead to a global network optimal result. In [21], each node maintains some local counters to collect the states of packet transmission, based on which the conditional collision probability and the transmission probability of the opponents' behavior can be calculated without negotiation. Zheng et al. [23] extend the works in [22] and investigate the problem of channel selection where no information exchange is available among users without the centralized controller. Each user adaptively updates its channel selection strategy relying on the individual experienced action-reward. It can be noted that the prior solutions in [22, 23] are similar to our design as they do not need to exchange information in dynamic and distributed networks. Nevertheless, the prior work needs a mechanism to distinguish active users from inactive ones, which is not required in our design.

For the learning-based category, reinforcement learning approaches are firstly proposed to achieve low energy consumption and low computational complexity in WSN [24–26]. Subsequently, more and more investigations are performed by combining it with the other mechanisms [29–33] in cognitive networks. Teng et al. [27] have discussed a scheme which adopts a Q-learning-based auction game to help nodes compete channel access opportunity. Kakalou et al. [28] and Saleem et al. [29] use cluster-based architectures instead of the central entity, in which cluster head observes the traffic of primary user (PU) to avoid collisions while keeping other member nodes synchronized. In [30], Lin et al. have investigated a novel dynamic spectrum access framework with control information exchange through beacons. In [31], a novel distributed Q-learning algorithm with heuristically accelerated scheme has been shown to be a powerful approach to solve dynamic spectrum access problem. The main insight of these contributions is employing interactive information among nodes; however, an excess of information exchange may oppose the alleviation of communication overhead. Motivated by these observations, in our solution, the information exchange is not necessary in our approach.

### 3. System Model

There are two types of node in the considered system: AP and sensor node. Assume that there are  $k$  orthogonal channels and  $n$  selfish sensors in the networks ( $k \leq n$ ). Orthogonal Frequency Division Multiple Access (OFDMA) is applied so that each sensor can access different channels by utilizing the feedback information (e.g., the channel gain) from AP [33, 34]. The sensors in the network are related to diverse channel gains. We further assume that the interference comes only from the sensors that are intended to contend the same channel. Figure 1 shows the considered network model. There are two crucial problems in this system, that is, how the sensors select proper channels and how they compete

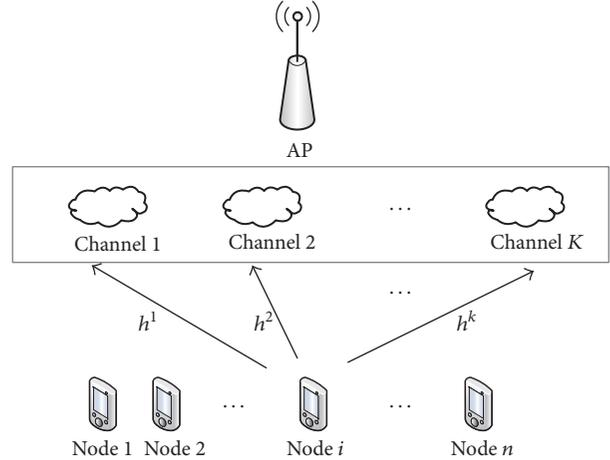


FIGURE 1: System model.

to access if two or more sensors select the same channel simultaneously. We will discuss these problems in detail in the followed subsections.

**3.1. Channel Selection.** We consider a noncooperative channel selection scenario in Figure 1, where  $N = \{1, 2, \dots, n\}$  is the set of fully distributed sensors,  $K = \{1, 2, \dots, k\}$  is the set of the available channels, and  $h = \{h^1, h^2, \dots, h^k\}$  denotes the set of channel gains. We assume that each sensor is equipped with a single radio transceiver, and it can dynamically access to any channel. It is worth noting that there is a unique policy in the channel selection process that the sensors individually select the channels with maximum channel gain. Due to the selfish behavior among the sensors, there are no negotiation messages exchanged in this process. However, the request and acknowledgement interacting messages between the sensors and the AP are still existed. According to the different strategies, we divide the on-policy channel selection method into two categories.

One category belongs to the real-time strategy, which is named online decision method based on noncooperative game. During the online decision process, the sensor first transmits a random access request to AP and then AP will send the feedback acknowledgement message. After that, the sensor can calculate its own maximum utility via the feedback from AP, which covers the information on channel gains of different channels. The optimal channel, that is, usually the one with the highest channel gain, will be selected.

The other category depends on the history of sensor states, and we name it offline learning method based on reinforcement learning. The offline learning process is actually an iterative exploration and exploitation process, in which each sensor will evaluate its current behavior and then improve it greedily. By this means, the random selection will gradually converge to the optimal one. It is noteworthy that the reinforcement learning algorithm is generally known as a real-time machine learning approach due to the immediate reward back from the environment. However, the immediate reward requires a number of learning iterations and may

bring unaffordable overhead to the network. Therefore, we revised it to gain an “offline” learning approach which will be affordable for the system with limited computational capability and power supply. The detailed procedures will be explained in Sections 4 and 5, and it is validated that this “offline” algorithm can prolong the network lifetime significantly.

**3.2. Multiple Access Contention.** The multiple access model is to determine the opportunity of channel access when some sensors select the same channel. We consider a CSAMA/CA scheme (e.g., the 802.11 DCF protocol) which is used in this distributed application to resolve the channel contention. In the carrier sensing phase, each sensor detects whether the channel is idle, and then, the sensor takes binary exponential backoff (BEB) algorithm to access the channel in the collision avoidance phase. According to the well-known research [35], in saturation conditions, the conditional transmission probability of sensor can be calculated as

$$\tau = \frac{2 \cdot (1 - 2p)}{(1 - 2p) \cdot (CW_{\min} + 1) + p \cdot CW_{\min} \cdot (1 - (2p)^m)}, \quad (1)$$

where  $\tau$ ,  $p$ ,  $CW_{\min}$ , and  $m$ , respectively, denotes transmission probability, collision probability, minimum contention window, and backoff stage.

Based on (1), probability that at least one sensor transmits packets can be expressed as

$$p_{\text{tr}} = 1 - (1 - \tau)^n. \quad (2)$$

The transmission success probability of each sensor can be written as

$$p_s = \frac{N \cdot \tau \cdot (1 - \tau)^{n-1}}{p_{\text{tr}}}. \quad (3)$$

Accordingly, the achievable rate of sensor  $i$  is given as follows

$$\begin{aligned} r &= W \log_2 (1 + \beta \cdot \text{SINR}) \\ &= W \log_2 \left( 1 + \frac{\beta \cdot P_{\text{tx}} \cdot h_i}{P_{\text{tx}} \cdot h_i \cdot \rho + \sigma^2} \right), \end{aligned} \quad (4)$$

where  $W$  is the bandwidth. SINR is the signal-to-interference-plus-noise ratio.  $\beta$  denotes the ratio of the current data rate to the Shannon capacity.  $P_{\text{tx}}$  is the transmission power.  $h_i$  denotes the channel gain of sensor node  $i$ .  $\rho$  is the correlation coefficient between sensor node and AP.  $\sigma^2$  is the noise power.

Next, we can calculate the time consumption by different events in the transmission process as follows:

(i) The time consumption due to conflict when more than two sensors contend for one transmission opportunity.

$$\begin{aligned} T_c &= \frac{(\text{PHY\_header} + \text{MAC\_header} + \text{packet\_length})}{r} \\ &+ \text{difs} + \delta, \end{aligned} \quad (5)$$

where  $\delta$  is the propagation delay. PHY\_header denotes the length of header for the physical layer. MAC\_header denotes the length of header for the MAC layer. packet\_length denotes the length of one data packet.  $r$  denotes the achievable rate. difs denotes the length of Short Interframe Space.

(ii) The time consumption associated with a successful transmission,

$$\begin{aligned} T_s &= \frac{(\text{PHY\_header} + \text{MAC\_header} + \text{packet\_length})}{r} \\ &+ \text{sifs} + \text{difs} + \text{ack} + 2 \cdot \delta, \end{aligned} \quad (6)$$

where sifs denotes the length of Short Interframe Space. ack denotes the length of acknowledgement message.

(iii) Therefore, the average time consumption can be calculated respectively as

$$\begin{aligned} T_{\text{fail}} &= (1 - p_{\text{tr}}) \cdot \text{slot} + p_{\text{tr}} \cdot (1 - p_s) \cdot T_c, \\ T_{\text{success}} &= p_s \cdot p_{\text{tr}} \cdot T_s, \end{aligned} \quad (7)$$

where slot denotes the length of one slot.

And the average throughput can be formulated as

$$\begin{aligned} T_r &= \frac{\text{packet\_length}}{\text{PHY\_header} + \text{MAC\_header} + \text{packet\_length}} \\ &\cdot \frac{T_{\text{success}}}{T_{\text{success}} + T_{\text{fail}}} \cdot r, \end{aligned} \quad (8)$$

where  $T_{\text{success}}$  denotes the average time consumption of transmission success.  $T_{\text{fail}}$  denotes the average time consumption of transmission fail.

From the above expression, we observe that the throughput of sensor is influenced by the achievable data rate  $r$ , which is essentially affected by the channel gain according to (4). Due to this fact, the channel with the maximum gain will be selected through the channel selection procedure.

The frequently used notations in this paper are summarized in Notations.

## 4. Online Decision Algorithm for Channel Selection

Taking account of the selfish behavior of each sensor in the network, we formulate the channel selection problem in multichannel WSN as a noncooperative game. The benefit for noncooperative game is that it requires no coordination control or information exchange among nodes. Based on this model, we propose an online decision algorithm for channel selection.

**4.1. Noncooperative Game Model.** Let us denote the game as  $\Gamma = \{N, \{S_i\}_{i \in N}, \{U_i\}_{i \in N}\}$ , where  $N$  is the player set (i.e., the set of sensor nodes),  $\{S_i\}_{i \in N}$  is the channel selection strategy set for player  $i$ , and  $\{U_i\}_{i \in N}$  is the utility of player  $i$ .

The utility function reflects the throughput of a sensor in the selected channel  $k$ , which can be defined as

$$u_i = T_r(h_i^k), \quad (9)$$

```

// Initialization
(1) AP evaluates the  $h^k$ 
// Each sensor obtains the channel gain
(2) Sensor  $i$  transmits request to AP
(3) Sensor  $i$  obtains channel gain  $\{h_i^k\}$ ,  $i \in N$ ,  $k \in K$  from the feedback
// Each sensor select the optimal channel through the Non-cooperative game
(4) while ( $u_{i,k}(S_i^*, S_{-i}^*) < u_{i,k}(S_i', S_{-i}^*)$ ) do
(5)   for  $i = 1$  to  $n$ 
(6)     for  $k = 1$  to  $k$ 
(7)       Sensor  $i$  calculates the utility function  $u_{i,k}$  on each channel according to (9)
(8)     end for
(9)   end for
(10) end while
(11)  $S_i^*(h_i^k) = \arg \max_{S_i} u_{i,k}(S_i, S_{-i}^*)$ 
(12) Take greedy strategy  $S_i^*$  to select the  $k$ th channel

```

ALGORITHM 1: Channel selection based on noncooperative game.

where  $h_i^k$  denotes the channel gain obtained by player  $i$  in channel  $k$ .

*Definition 1.* A channel selection profile  $S^* \in \{S_i\}_{i \in N}$  is a pure strategy Nash Equilibrium (NE) of the game  $\Gamma$  if and only if no player can improve its utility by deviating unilaterally,

$$\begin{aligned}
 & u_i(S^*) \geq u_i(S') \\
 \text{s.t. } & \forall i \in N, \forall S \in \{S_i\}_{i \in N}, S^* \neq S'.
 \end{aligned} \tag{10}$$

Note that NE can be generally classified into pure strategy NE and mixed strategy NE. The mixed strategy NE usually seeks for a stable equilibrium state in which sensors select channel with negotiation. In this paper, we employ pure strategy NE in which each sensor selects channel in an on-policy manner. If player  $i$  decides to deviate from its individual NE, its utility will be degraded if the system is at such NE. Therefore, this property is particularly desirable. However, the sum-utility optimal channel selection problem is NP-hard. Thus, conventional optimization techniques cannot be applied directly and even centralized algorithms cannot guarantee the globally optimal solution. We propose Theorem 2 to characterize the game.

**Theorem 2.** *With the maximization of an individual node utility the global benefit of the system is also maximized.*

*Proof.* Dov Monderer and Shapleyb [32] have proven that the individual or global NE is the maximization of the potential function. According to the concept of NE and (9), we can find that the utility  $u_i$  is the best response for node  $i$  with strategy  $S^*$ , either individually or globally.

In terms of the global optimization, the sum-utility optimal channel selection problem can be formalized as

$$U = \text{maximize} \quad \sum_{i \in N, S^* \in S} u_i, \tag{11}$$

where  $U$  denoted the sum of each individual node's utility. According to Theorem 2, we should develop an effective algorithm to obtain the global optimal NE.  $\square$

*4.2. Algorithm Description.* Each sensor is regarded as an online decision automaton agent, which selects the channel according to greed strategy  $S^*$  ( $S^*$  is the strategy that selects the channel with the highest channel gain). In other words, each sensor will maximize its utility function  $u_{i,k}$  in a greedy way. And the algorithm can be described as below.

In the initialization phase, the AP evaluates the channel gain in each channel. Each sensor first transmits the required message to the AP. Next, sensor  $i$  obtains channel gains of different channels in feedback acknowledgement message from the AP, based on which the utility function of sensor  $i$  is locally calculated. The above operations will be performed iteratively until the expected utility function  $u_{i,k}$  converges to the unique NE. Finally, sensor  $i$  takes greedy strategy  $S_i^*$  to select the corresponding channel which has the optimal channel gain value. Algorithm 1 describes the channel selection process based on noncooperative game.

*4.3. Convergence and Complexity Analysis.* The convergence of Algorithm 1 is guaranteed since the expected utility function  $u_{i,k}$  converges to the unique NE, and the number of iteration is  $O(n)$ . Within each iteration, the maximum computation of a sensor is  $O(n^k)$ . Therefore, the total computational complexity of Algorithm 1 is  $O(n^k)$ . In terms of storage requirement, each sensor needs to cache channel gains of different channels in feedback acknowledgement message; thus  $k$  memory units are required to store the immediate feedback acknowledgement messages in the case of the online decision algorithm based on noncooperative game. Obviously, the computational complexity and storage requirement of this immediate algorithm will be increased with the number of sensors and channels. In the following section, we will present an alternative channel selection algorithm with reduced computational complexity.

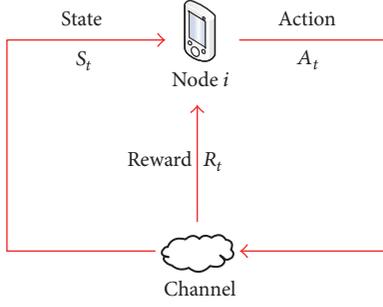


FIGURE 2: A simplified RL model.

## 5. Offline Learning Algorithm for Channel Selection

In this section, we present a decentralized channel selection algorithm by using the reinforcement learning framework. This framework selects the channel “offline” in a simpler way than the online algorithm.

**5.1. Decentralized Reinforcement Learning Framework.** Reinforcement learning (RL) is usually adopted to solve the problem that a learning agent is interacting with its environment to achieve goals related to the state of the environment [36]. Such an agent should be able to observe the state of the environment and take actions according to the feedback of the observation that affects the state in next time. As illustrated in Figure 2, at time  $t$ , sensor  $i$  observes the channel state and obtains the current channel state  $S_t$ , and then sensor  $i$  takes action  $A_t$  and obtains the reward  $R_t$ .

In a general reinforcement learning framework, there are two interacting objects which are agent and environment. And three types of exchange information are included in the learning process: state  $S_t$ , action  $A_t$ , and reward  $R_t$ . For each sensor, the learning process is called the exploration-exploitation tradeoff with two important characteristics: trial-and-error search and delayed reward. The current reward may not affect the next time state immediately, and the expected learning target will be obtained after a period of time. Therefore, we called this learning algorithm “offline” to distinguish it from the online decision algorithm in Section 4.

We propose a Q-learning algorithm for sensors to optimally select the channels according to the histories of observed states and the rewards accumulated into the current choice of action. Some definitions of the algorithm are given as follows.

**State.** We define that state  $S_t$  is observed by node  $i$  at time  $t$ . We use  $S = \{S_1, \dots, S_t\}$  to denote the finite set of state space,  $S_t \in S$ . The state transition from  $S_t$  to  $S_{t+1}$  depends on the action, and accordingly the next state  $S_{t+1}$  can be observed when the next action occurs.

**Reward.** We use  $R = \{R_1, \dots, R_t\}, \forall R_t = \{-1, +1\}$  denotes the finite set of reward space. If node  $i$  at time  $t$  selects optimal channel, the reward value will be equal to “1.” Otherwise, the reward value equals “-1.”

**Action.** We define that the action  $A_{i,t}$  is implemented by node  $i$  at time  $t$ . We use  $A = \{A_{i,1}, \dots, A_{i,t}\}, \forall A_{i,t} = \{A_{t,R_t=1}, A_{t,R_t=-1}\}$  to denote the finite set of action space, where  $A_{t,R_t=+1}$  and  $A_{t,R_t=-1}$ , respectively, denote that the node obtained positive reward and negative reward after action  $A_t$  are taken.

**Action-Value Function.** The action-value function  $Q(S, A)$  is associated with the action  $A$  and state  $S$  at time  $t$ . In Section 4, it is equivalent to the utility function.

**5.2. Channel Selection with Reinforcement Learning.** In the above reinforcement learning framework, each sensor interacts with the channel environment. At each discrete time step  $t$ , sensor observes current state  $S_t$ , takes action  $A_t$ , and obtains feedback reward  $R_t$ . As we assume that there are  $k$  orthogonal channels allocated by the AP in the system model, each sensor selects the optimal channel with the probability of  $1/k$  and other channels with the probability of  $(k-1)/k$ . The channel selection process is a memory less random process and obtains a sequence of random states with the Markov property. Therefore, it can be modeled as the Markov Decision Process (MDP) with a sequence of state information. The “state information” includes actions, states, and rewards. According to policy  $\pi(A | S)$ , sensor observes state  $S$  which is affected by action  $A$ . The task for a sensor is to learn the policy  $\pi(A | S)$  to maximize the expectation of action-value function  $Q_\pi(S, A)$ .

Since  $S_t$  is a Markov process, that is, the information related to past states  $[S_1, \dots, S_{t-1}]$  is covered by state  $S_t$ . We only need to store the current state  $S_t$ , which allows a considerable reduction of storage requirement.

**Definition 3.** A history  $H_t$  is a sequence of actions, states, and rewards

$$H_t = A_0, S_1, R_1, \dots, A_{t-1}, S_t, R_t. \quad (12)$$

**Definition 4.** A channel selection process is a tuple  $\langle S, A, R, P, \gamma \rangle$ , where  $S, A, R$ , and  $P$ , respectively, denote a finite set of states, actions,  $R$  rewards, and the transition probability matrix.  $\gamma$  ( $\gamma \in [0, 1]$ ) is the discount factor in order to avoid infinite returns in cyclic Markov processes.

As illustrated in Figure 3, sensor  $i$  fulfills the channel selection process through two discrete time steps. White point is the initial state, red point implies sensor  $i$  selected optimal channel with the probability  $1/k$ , and black point implies sensor  $i$  selected other channel with the probability  $(k-1)/k$ .

Let us consider the operation at time  $t+1$  as an example, and the optimal state transition probability is defined as

$$P_{S_{t+1}, \text{RED}} = P[S_{t+1} = \text{RED} | S_t = \text{WHITE}], \quad (13)$$

and the other state transition probability is defined as

$$P_{S_{t+1}, \text{BLACK}} = P[S_{t+1} = \text{BLACK} | S_t = \text{WHITE}]. \quad (14)$$

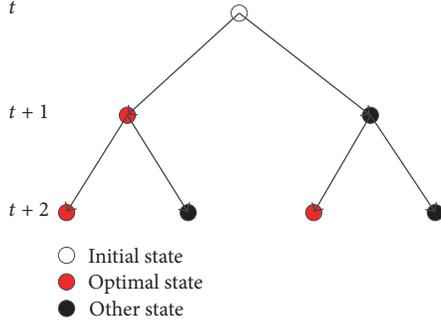


FIGURE 3: Two steps state transition model.

**Definition 5.** The optimal action-value function  $Q_*(S, A)$  is the maximum action-value function over all policies

$$Q_*(S, A) = \max_{\pi} Q_{\pi}(S, A). \quad (15)$$

$Q_*(S, A)$  can be recursively computed by adopting the Bellman optimality equation:

$$Q_*(S, A) = R_S^A + \gamma \sum_{S_{\text{RED}} \in S} P_{S, \text{RED}}^A \max_{A_{\text{RED}}} Q_*(\text{RED}, A_{\text{RED}}). \quad (16)$$

$Q_*(S, A)$  always specifies the best possible performance in the MDP. For the MDP of channel selection, this property is hold according to the following theorems.

**Theorem 6.** *There exists an optimal policy  $\pi_*$  that is better than or equal to all other policies,  $\pi_* \geq \pi, \forall \pi$ .*

**Theorem 7.** *All optimal policies achieve the optimal action-value function,  $Q_{\pi_*}(S, A) = Q_*(S, A)$ .*

Now, we use the Q-learning algorithm to solve the MDP problem of channel selection. Each sensor decides its next action  $A_{t+1}$  based on the trend of a sequence of actions, states, and rewards, and then it observes current reward  $R_t$  and state  $S_t$  to update the Q-value of the action-value function  $Q(S, A)$ . The updated Q-value will affect the next-round channel selection. The Q-value can be expressed as

$$Q(S_t, A_t) \leftarrow (1 - \alpha) Q(S_t, A_t) + \alpha \left( R_t + \gamma \max_{a_t} Q(S_{t+1}, A_{t+1}) \right) \quad (17)$$

$$\text{s.t. } 0 \leq \alpha \leq 1, 0 \leq \gamma \leq 1,$$

where  $\alpha$  is the learning rate, which specifies the updating speed of the Q-value.  $\gamma$  is the discount factor that determines the present value of future rewards.  $Q(S_t, A_t)$  denotes current Q-value and  $Q(S_{t+1}, A_{t+1})$  denotes the expected Q-value.

As shown in Figure 4, we improve the learning policy  $\pi$  by acting greedily with respect to optimal policy  $\pi_{* \text{greedy}}$

in each turn of iteration. Figure 4(a) represents the learning policy  $\pi$  that will converge to the optimal policy  $\pi_{* \text{greedy}}$  after  $t$  times iteration. In Figure 4(b), we use contracting mapping theorem to represent this channel selection process. The red solid line denotes optimal policy  $\pi_{* \text{greedy}}$ , and the blue solid line denotes learning policy  $\pi$ . The green dash arrow denotes stochastic action, and this action will be improved gradually by a series of feedback information according to the optimal policy  $\pi_{* \text{greedy}}$ . The rose dash arrow and purple dash arrow, respectively, denote reward information and state information. They are used to evaluate the action according to the learning policy  $\pi$  after  $n'$ -step returns ( $n' = 1, 2, \dots, \infty$ ).

Using the offline algorithm to select channel is an exploration-exploitation process that requires no prior knowledge for each sensor. Through the learning process, the stochastic behavior of sensors can gradually converge to the optimal channel selection.

**5.3. Algorithm Description.** Each sensor is regarded as an offline learning automaton agent, and the relevant task is to learn the policy  $\pi$  maximizing the expectation of Q-value  $Q(S_t, A_t)$ . The algorithm is described as follows.

In the initialization phase, each sensor initializes its action space and Q-value. Then, sensor  $i$  first takes an random channel selection action  $A_t$  and the AP feedback reward  $R_t$  to sensor  $i$ . Next, sensor  $i$  observes the current reward value and the next time state. If sensor  $i$  selects optimal channel at time  $t$ , the corresponding reward value will be equal to "1"; otherwise the reward value equals "−1." Subsequently, sensor  $i$  updates Q-value. This process is repeated until the learning policy  $\pi$  that converges to optimal policy  $\pi_{* \text{greedy}}$ . Finally, sensor  $i$  chooses channel selection action  $A_{t+1}$  from  $S_{t+1}$  by using policy  $\pi$  which derives from  $Q(S_t, A_t)$ . Algorithm 2 describes the channel selection process based on Q-learning.

**5.4. Complexity and Convergence Analysis.** The number of iterations for Algorithm 2 is  $O(n)$ . Within each iteration, the maximum computation of a sensor is  $O(1)$ . Therefore, the total computational complexity of Algorithm 2 is  $O(n)$ . In terms of storage requirement, each sensor only needs to use one memory unit to store the current state information. In this way, the computational complexity and storage requirement can experience signification reduction compared to Algorithm 1.

The convergence of this channel selection algorithm is characterized by the following theorem.

**Theorem 8.** *The learning policy  $\pi$  converges to optimal policy  $\pi_{* \text{greedy}}$  if the following conditions are met:*

- (i) *The optimal policy  $\pi_{* \text{greedy}}$  has unique point.*
- (ii) *Robbins-Monro sequence [37] of step-sizes  $\alpha$  satisfies  $\sum_{t=0}^{\infty} \alpha_t = \infty, \sum_{t=0}^{\infty} \alpha_t^2 < \infty$ .*
- (iii) *The state  $S$  and action  $A$  spaces are finite.*

From Theorem 8, we know that the learning policy  $\pi$  converges to the optimal policy  $\pi_{* \text{greedy}}$ . Since a finite channel

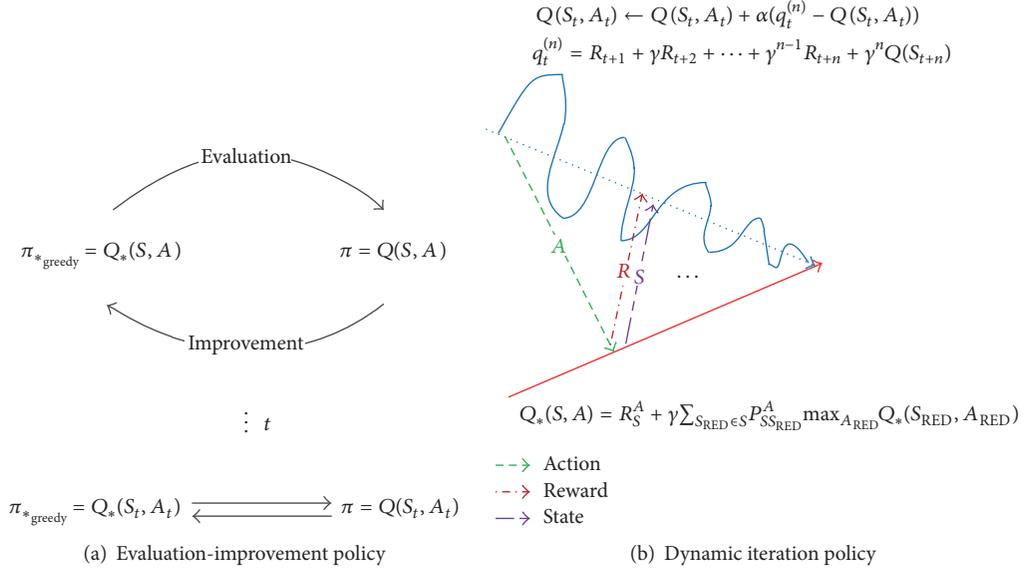


FIGURE 4: Policy iteration process model.

```

// Initialization
(1) Each sensor initializes its action space A
(2) Each sensor initializes its Q-value  $Q(S_t, A_t) = 0, \forall S_t \in S, \forall A_t \in A$ 
//Learning
(3) Sensor  $i$  takes random action  $A_t$ 
(4) Sensor  $i$  observes reward  $R_t$  and state  $S_{t+1}$ 
(5) while ( $Q(S_t, A_t) = Q_*(S_t, A_t)$ ) do
(6)   for  $i = 1$  to  $n$ 
(7)     for  $t = 1$  to  $t$ 
(8)       if  $S_{t+1}$  is good
(9)          $R_t = +1$ 
(10)      else
(11)         $R_t = -1$ 
(12)      end if
(13)      Sensor  $i$  updates its Q-value according to (17)
(13)       $Q(S_t, A_t) \leftarrow (1 - \alpha_t)Q(S_t, A_t) + \alpha_t(R_t + \gamma \max_{\alpha_{t+1}} Q(S_{t+1}, A_{t+1}))$ 
(14)       $\pi(S_t, A_t) = \arg \max(Q(S_t, A_t))$ 
(15)    end for
(16)  end for
(17) end while
(18) Choose  $A_{t+1}$  from  $S_{t+1}$  using policy  $\pi$  derived from  $Q(S_t, A_t)$ 

```

ALGORITHM 2: Channel selection based on Q-learning.

selection process has only a finite number of policies, this process must converge to an optimal policy and optimal action-value function within a finite number of iterations. The proof is given as follows.

*Proof.* We can rewrite the updating rule of the Q-value with learning policy in (18) as follows:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha(q_t^{(n')} - Q(S_t, A_t)), \quad (18)$$

where  $q_t^{(n')}$  denotes the  $n'$ -step Q-value return value and it can be expanded as

$$q_t^{(n')} = R_{t+1} + \gamma R_{t+2} + \dots + \gamma^{n'-1} R_{t+n'} + \gamma^{n'} Q(S_{t+n'}). \quad (19)$$

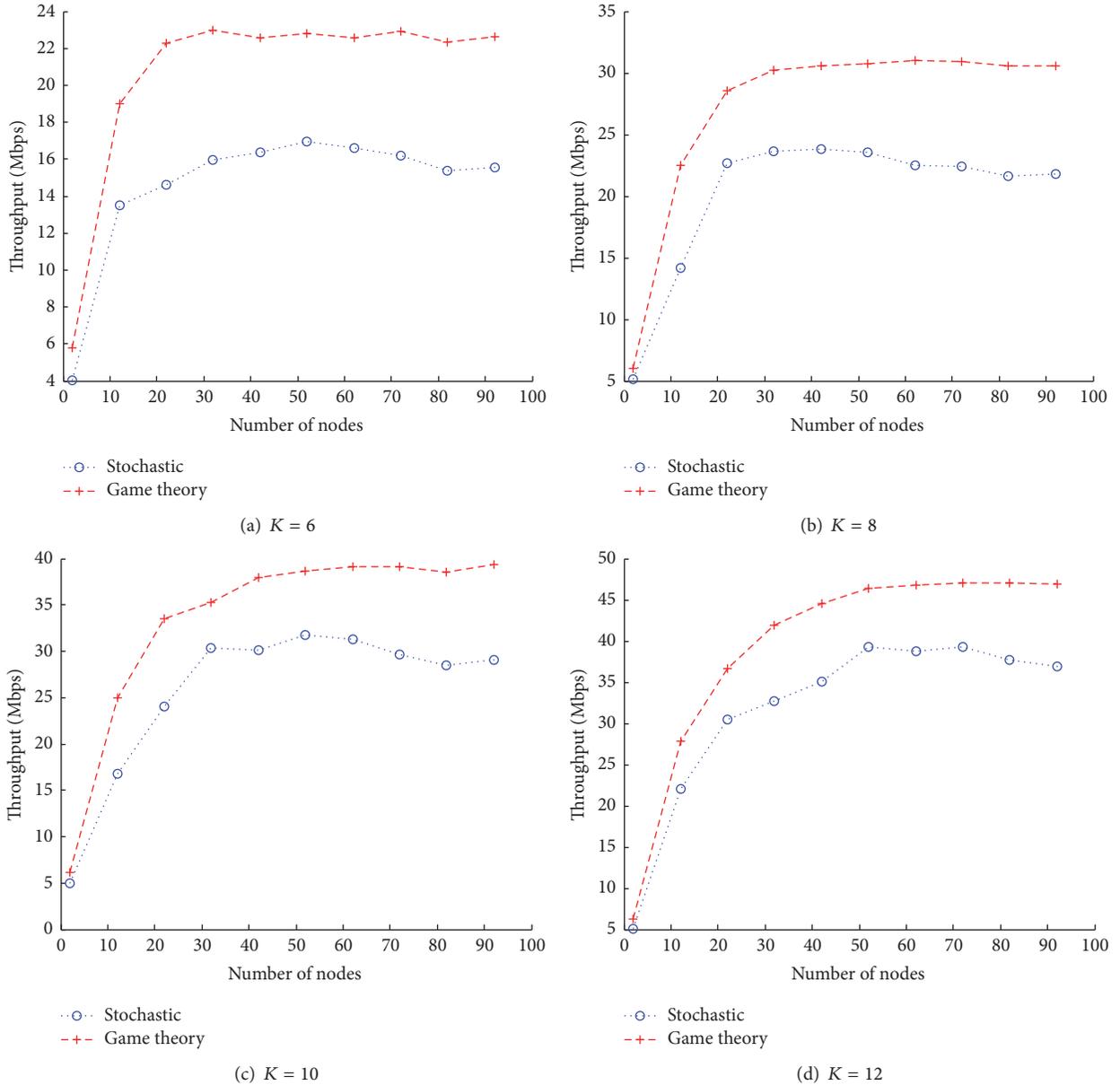


FIGURE 5: Throughput performance of online noncooperative game algorithm.

The  $n'$ -step ( $n' = 1, 2, \dots, \infty$ ) return values can be considered as follows:

$$\begin{aligned}
 n' = 1 & \quad q_t^{(1)} = R_{t+1} + \gamma Q(S_{t+1}) \\
 n' = 2 & \quad q_t^{(2)} = R_{t+1} + \gamma R_{t+2} + \gamma^2 Q(S_{t+2}) \\
 & \quad \vdots \\
 n' = \infty & \quad q_t^{(\infty)} = R_{t+1} + \gamma R_{t+2} + \dots + \gamma^{T-1} R_T.
 \end{aligned} \tag{20}$$

In (18), the  $Q$ -value at time  $t$  is a determinate value. For the return value  $q_t^{(n')}$  of the  $n'$ -step  $Q$ -value, we can see that it is finite and based on the number of evaluation  $T$ , which suggests the convergence property of  $q_t^{(n')}$  is irrelevant to the number of step.

Therefore, the  $Q$ -value in (18) must be a finite value. Theorem 8 is proved.  $\square$

## 6. Evaluation

**6.1. Simulation Setup.** When multiple sensors select one same channel, we assume that all sensor nodes resolve contention based on IEEE 802.11 standard, that is, the BEB backoff algorithm. The parameters are set as  $CW_{\min} = 32$ ,  $m = 5$ ,  $P_{\text{tx}} = 12$  dBm,  $\rho = 0.1$ ,  $\sigma^2 = -80$  dBm,  $W = 1$  Hz, and  $\beta = 0.5$ . The AP randomly assigns the channel gain in each turn of iteration according to  $h = \text{rand}(n, k) \times 0.3 + 0.6$ . OFDMA technique is used in the physical layer, where the packet length is 512 bits, the acknowledgement message length is 304 bits, and the headers for the physical layer and

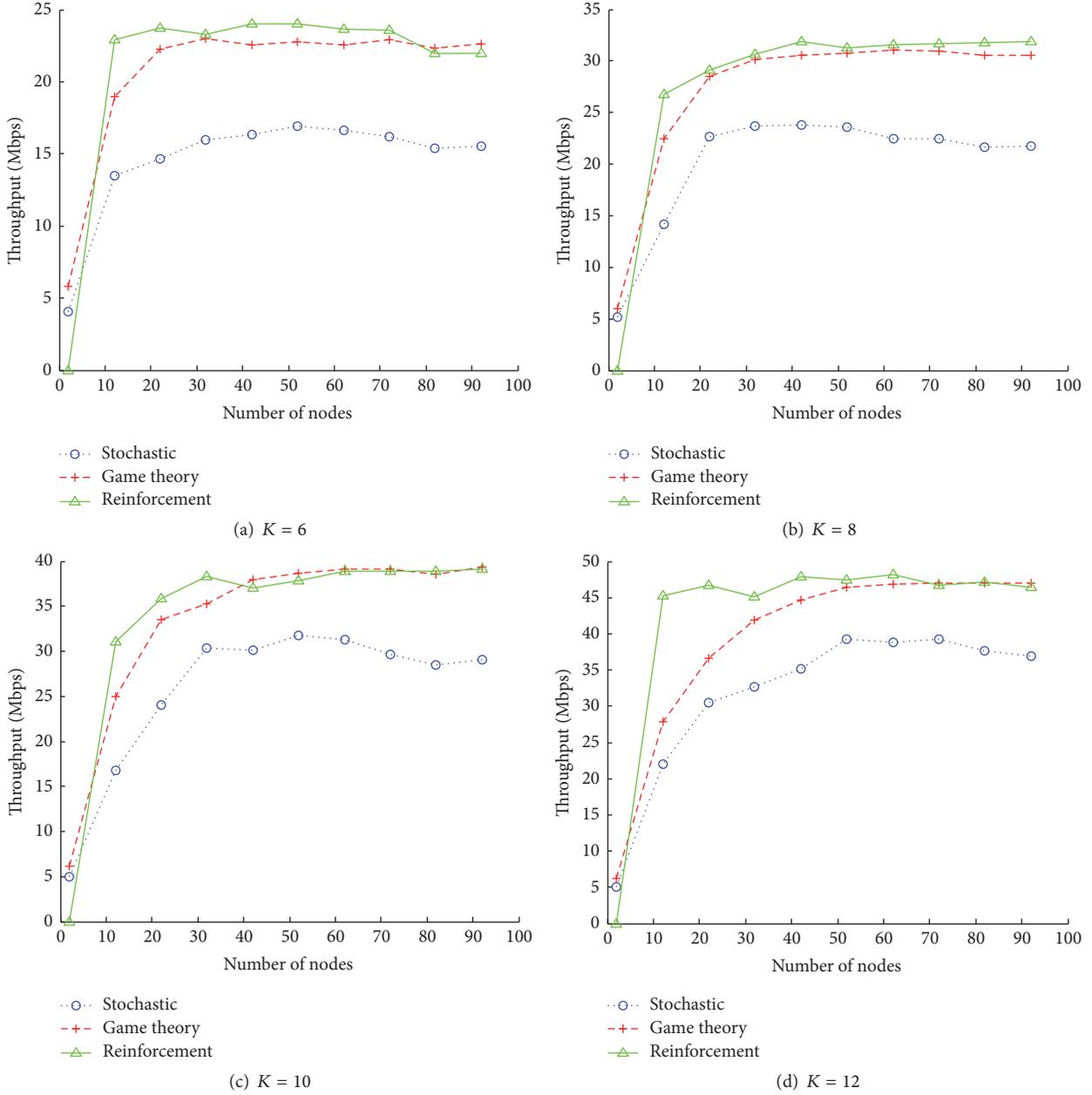


FIGURE 6: Throughput performance of offline reinforcement learning algorithm.

MAC are, respectively, 192 bits and 272 bits. And the slot time is  $20 \mu\text{s}$ , the Short Interframe Space (SIFS) and the DCF Interframe Space (DIFS) are, respectively,  $10 \mu\text{s}$  and  $50 \mu\text{s}$ , and the propagation delay is  $1 \mu\text{s}$ . The discount factor  $\gamma$  was chosen to be 0.9 and the learning rate  $\alpha$  was designed to be 0.6. To simulate the dynamic network environment, we set the initial number of sensor to 2, and then 10 sensors are added in each turn of iteration until the total number of sensors reaches 92. In the simulation, we vary the number of channels from 6 to 12. The presented results are obtained by 5000 independent Monte Carlo simulations.

**6.2. Evaluation of Online Decision Based on Game Theory.** Figure 5 shows the performance comparison in terms of the global throughput, which is the sum of individual throughput achieved by each sensor. The stochastic algorithm acts as the competing scheme as it is a typical off-policy channel selection algorithm. As shown in Figure 5, the noncooperative game algorithm performs better in throughput under various numbers of channels. In particular, the throughput is higher and more stable than the stochastic algorithm. We can also observe that the increasing tendency of the curves becomes slower when the number of sensors becomes larger. It can

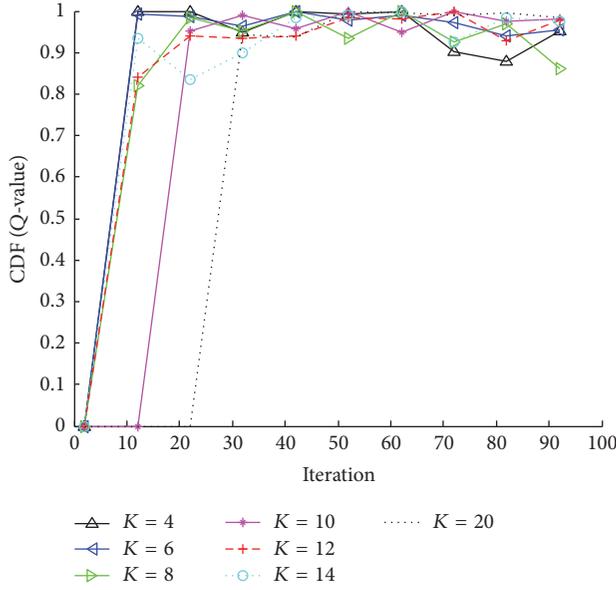


FIGURE 7: Convergence performance of offline reinforcement learning algorithm under different numbers of channels.

be concluded that the throughput gets saturation in each channel, and more sensors bring more severe contention.

**6.3. Evaluation of Offline Learning Based on Reinforcement Learning.** When the offline reinforcement learning algorithm is adopted, as shown in Figure 6, a better performance in the global throughput is attained compared to the stochastic channel selection algorithm. Similar to the simulation results given by Figure 5, the throughput will get saturation with the increasing number of sensors, which also stems from the severer contention. As shown in Figure 6, we can see that the initial process of reinforcement learning algorithm performance curve is unstable. This is because the learning algorithm is used to help sensors to reduce stochastic behavior. Therefore, at the beginning the sensor behaves stochastically, while it eventually converges to the stable and optimal behavior. The performance gaps between stochastic algorithm and our proposed algorithm increase with the number of sensors. However, the gaps between our proposed algorithms are minor. The reasons can be as follows: (1) more sensors incur severe contention, and (2) our learning algorithm gradually converges toward a pure NE employed in Section 4.

Figure 7 plots the cumulative distribution function (CDF) of  $Q$ -value versus the number of sensors. It can be seen that the convergence performance becomes stable with the increasing number of sensors. In addition, the convergence can be gradually achieved when the number of sensors is larger than the number of channels. Moreover, the increasing tendency of the curves becomes stable when the number of sensors added in each turn of iteration is larger than the number of channel. The reasons can be as follows: (1) the samples space becomes larger on each channel when the total number of sensor increases, and (2) sensors have been

accumulated enough historical observations and decision experiences.

## 7. Conclusions

In this paper, two channel selection algorithms based on online self-decision and offline self-learning, respectively, have been investigated in a multichannel wireless sensor networks. Sensor nodes in both algorithms behave selfishly and do not mutually negotiate information among other sensors. The online self-decision is made based on noncooperative game, and the offline self-learning is done based on the reinforcement learning. The online self-decision can be made immediately and is suitable for the real-time application. By contrast, the offline self-learning algorithm can iteratively converge to the optimal channel selection with lower occupation of computational and storage resources; thus it is available for the applications with low computational complexity, communication overhead, and storage requirement. Theoretical analysis and simulation results demonstrated that the proposed channel selection methods can improve the throughput performance compared to the existing off-policy strategies.

## Notations

$K$ :	Set of the available channels
$N$ :	Set of fully distributed sensor nodes
$n$ :	Number of sensor nodes
$k$ :	Number of channels
$h$ :	Set of channel gains
$h^k$ :	Channel gain in the $k$ th channel
$h_i^k$ :	Channel gain of sensor $i$ in the $k$ th channel
$S_i^*$ :	The optimal strategy for sensor $i$
$u_i$ :	The utility function of sensor $i$
$S_t$ :	The channel state at $t$ time
$R_t$ :	The feedback reward at $t$ time
$A_t$ :	The sensor node take action at $t$ time
$t$ :	The discrete time steps
$n'$ :	Number of steps
$Q_\pi(S, A)$ :	The action-value function with policy $\pi$
$Q_*(S, A)$ :	The optimal action-value function
$P_{S_{t+1}, \text{RED}}$ :	State transition probability of sensor selected optimal state from $t$ to $t + 1$
$P_{S_{t+1}, \text{BLACK}}$ :	State transition probability of sensor selected other state from $t$ to $t + 1$
$\pi_{* \text{greedy}}$ :	The optimal policy with greedy algorithm.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research was supported in part by the National Natural Science Foundation of China (Grants no. 61471376, no. 61601482, and no. 61501482) and by the foundation from

Science and Technology on Information Transmission and Dissemination in Communication Networks Lab.

## References

- [1] N. C. Luong, D. T. Hoang, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Data Collection and Wireless Communication in Internet of Things (IoT) Using Economic Analysis and Pricing Models: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2546–2590, 2016.
- [2] Y. Wu, J. A. Stankovic, T. He, and S. Lin, "Realistic and efficient multi-channel communications in wireless sensor networks," in *Proceedings of the 27th Conference on Computer Communications (INFOCOM '08)*, pp. 1193–1201, IEEE, Phoenix, Ariz, USA, April 2008.
- [3] C. Li, P. Wang, H.-H. Chen, and M. Guizani, "A cluster based on-demand multi-channel MAC protocol for wireless multimedia sensor networks," in *Proceedings of the IEEE International Conference on Communications (ICC '08)*, pp. 2371–2376, Beijing, China, May 2008.
- [4] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "EM-MAC: a dynamic multichannel energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 12th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '11)*, ACM, Las Vegas, Nev, USA, May 2011.
- [5] M. Félegyházi, M. Čagalj, and J.-P. Hubaux, "Efficient MAC in cognitive radio systems: A game-theoretic approach," *IEEE Transactions on Wireless Communications*, vol. 8, no. 4, pp. 1984–1995, 2009.
- [6] B. Han and K. Kawanishi, "Using game theory to investigate stochastic channel selection for multi-channel MAC protocol," in *Proceedings of the 2012 IEEE International Conference on Communication Systems, ICCS 2012*, pp. 172–176, Singapore, November 2012.
- [7] L. Canzian, M. Zorzi, and M. van der Schaar, "Game theoretic design of MAC protocols: Pricing and intervention in slotted-Aloha," in *Proceedings of the 2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 707–714, Monticello, Ill, USA, October 2013.
- [8] J. Nie and S. Haykin, "A Q-learning-based dynamic channel assignment technique for mobile communication systems," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 5, pp. 1676–1687, 1999.
- [9] G. Naddafzadeh-Shirazi, P.-Y. Kong, and C.-K. Tham, "Distributed reinforcement learning frameworks for cooperative retransmission in wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 8, pp. 4157–4162, 2010.
- [10] P. Zhou, Y. Chang, and J. A. Copeland, "Learning through reinforcement for repeated power control game in cognitive radio networks," in *Proceedings of the 53rd IEEE Global Communications Conference, GLOBECOM 2010*, December 2010.
- [11] W. Zame, J. Xu, and M. Van Der Schaar, "Cooperative multi-agent learning and coordination for cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 3, pp. 464–477, 2014.
- [12] P. Nuggehalli, M. Sarkar, K. Kulkarni, and R. R. Rao, "A Game-theoretic analysis of QoS in wireless MAC," in *Proceedings of the INFOCOM 2008: 27th IEEE Communications Society Conference on Computer Communications*, pp. 46–50, April 2008.
- [13] B. Shrestha, D. Niyato, Z. Han, and E. Hossain, "Wireless access in vehicular environments using BitTorrent and bargaining," in *Proceedings of the 2008 IEEE Global Telecommunications Conference, GLOBECOM 2008*, pp. 5254–5258, December 2008.
- [14] D. Chatterjee and T. F. Wong, "Resource allocation and cooperative behavior in fading multiple-access channels under uncertainty," in *Proceedings of the 2009 IEEE Military Communications Conference, MILCOM 2009*, October 2009.
- [15] B. Wang, Z. Han, and K. J. R. Liu, "Distributed relay selection and power control for multiuser cooperative communication networks using stackelberg game," *IEEE Transactions on Mobile Computing*, vol. 8, no. 7, pp. 975–990, 2009.
- [16] H. Cui, Y. Wang, Q. Guan, and H. Zhang, "Distributed Interference-Aware Cooperative MAC Based on Stackelberg Pricing Game," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 9, pp. 4124–4134, 2015.
- [17] Y. Chen and S. Kishore, "A game-theoretic analysis of decode-and-forward user cooperation," *IEEE Transactions on Wireless Communications*, vol. 7, no. 5, pp. 1941–1951, 2008.
- [18] M. Janzamin, M. Pakravan, and H. Sedghi, "A game-theoretic approach for power allocation in bidirectional cooperative communication," in *Proceedings of the IEEE Wireless Communications and Networking Conference 2010, WCNC 2010*, April 2010.
- [19] P. Ju and W. Song, "Repeated Game Analysis for Cooperative MAC With Incentive Design for Wireless Networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 7, pp. 5045–5059, 2016.
- [20] Y. Cho and F. A. Tobagi, "Cooperative and non-cooperative aloha games with channel capture," in *Proceedings of the 2008 IEEE Global Telecommunications Conference, GLOBECOM 2008*, pp. 4533–4538, December 2008.
- [21] L. Zhao and H. Zhang, "Using incompletely cooperative game theory in wireless LANs," in *Proceedings of the IET Conference on Wireless, Mobile and Sensor Networks 2007, CCWMSN'07*, pp. 129–132, December 2007.
- [22] L. Zheng, Y. Cai, and Y. Xu, "MAC-layer interference mitigation in dynamic and distributed environment: Dynamic graphic game with stochastic learning," in *Proceedings of the 2014 5th International Conference on Game Theory for Networks, GameNets 2014*, November 2014.
- [23] J. Zheng, Y. Cai, N. Lu, Y. Xu, and X. Shen, "Stochastic Game-Theoretic Spectrum Access in Distributed and Dynamic Environment," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 10, pp. 4807–4820, 2015.
- [24] J. Niu, "Self-learning scheduling approach for wireless sensor network," in *Proceedings of the 2010 2nd International Conference on Future Computer and Communication*, pp. V3-253–V3-257, Wuhan, China, May 2010.
- [25] Z. Liu and I. Elhanany, "RL-MAC: A reinforcement learning based MAC protocol for wireless sensor networks," *International Journal of Sensor Networks*, vol. 1, no. 3-4, pp. 117–124, 2006.
- [26] M. Mihaylov, Y.-A. Le Borgne, K. Tuyls, and A. Nowé, "Reinforcement Learning for Self-organizing Wake-Up Scheduling in Wireless Sensor Networks," *Communications in Computer and Information Science*, vol. 271, pp. 382–396, 2013.
- [27] Y. Teng, Y. Zhang, F. Niu, C. Dai, and M. Song, "Reinforcement learning based auction algorithm for dynamic spectrum access in cognitive radio networks," in *Proceedings of the 2010 IEEE 72nd Vehicular Technology Conference Fall, VTC2010-Fall*, September 2010.

- [28] I. Kakalou, G. I. Papadimitriou, P. Nicopolitidis, P. G. Sariannidis, and M. S. Obaidat, "A Reinforcement learning-based cognitive MAC protocol," in *Proceedings of the IEEE International Conference on Communications, ICC 2015*, pp. 5608–5613, June 2015.
- [29] Y. Saleem, K.-L. A. Yau, H. Mohamad, N. Ramli, and M. H. Rehmani, "Joint channel selection and cluster-based routing scheme based on reinforcement learning for cognitive radio networks," in *Proceedings of the 2nd International Conference on Computer, Communications, and Control Technology, I4CT 2015*, pp. 21–25, April 2015.
- [30] Y. Lin, C. Wang, J. Wang, and Z. Dou, "A novel dynamic spectrum access framework based on reinforcement learning for cognitive radio sensor networks," *Sensors*, vol. 16, no. 10, article no. 1675, 2016.
- [31] N. Morozs, T. Clarke, and D. Grace, "Distributed Heuristically Accelerated Q-Learning for Robust Cognitive Spectrum Management in LTE Cellular Systems," *IEEE Transactions on Mobile Computing*, vol. 15, no. 4, pp. 817–825, 2016.
- [32] D. Monderer and L. S. Shapleyb, "Potential Games," *Games & Economic Behavior*, vol. 14, no. 44, pp. 124–143, 1996.
- [33] 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.
- [34] G. Li and H. Liu, "Resource allocation for OFDMA relay networks with fairness constraints," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 11, pp. 2061–2069, 2006.
- [35] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.
- [36] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, The MIT Press, New York, NY, USA, 1998.
- [37] H. Robbins and S. Monro, "A stochastic approximation method," *Annals of Mathematical Statistics*, vol. 22, pp. 400–407, 1951.

## Research Article

# A Crowdsensing-Based Real-Time System for Finger Interactions in Intelligent Transport System

Chengqun Song,<sup>1,2</sup> Jun Cheng,<sup>1,2</sup> and Wei Feng<sup>1,2</sup>

<sup>1</sup>Guangdong Provincial Key Laboratory of Robotics and Intelligent System, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China

<sup>2</sup>The Chinese University of Hong Kong, Shatin, Hong Kong

Correspondence should be addressed to Jun Cheng; [jun.cheng@siat.ac.cn](mailto:jun.cheng@siat.ac.cn)

Received 3 July 2017; Revised 8 August 2017; Accepted 7 September 2017; Published 12 October 2017

Academic Editor: Zhaolong Ning

Copyright © 2017 Chengqun Song 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.

Crowdsensing leverages human intelligence/experience from the general public and social interactions to create participatory sensor networks, where context-aware and semantically complex information is gathered, processed, and shared to collaboratively solve specific problems. This paper proposes a real-time projector-camera finger system based on the crowdsensing, in which user can interact with a computer by bare hand touching on arbitrary surfaces. The interaction process of the system can be completely carried out automatically, and it can be used as an intelligent device in intelligent transport system where the driver can watch and interact with the display information while driving, without causing visual distractions. A single camera is used in the system to recover 3D information of fingertip for hand touch detection. A linear-scanning method is used in the system to determine the touch for increasing the users' collaboration and operability. Experiments are performed to show the feasibility of the proposed system. The system is robust to different lighting conditions. The average percentage of correct hand touch detection of the system is 92.0% and the average time of processing one video frame is 30 milliseconds.

## 1. Introduction

The World Health Organization in 2017 years reported that approximately 1.25 million people die every year as a result of a road traffic crash [1]. Distracted driving is a major factor of road traffic crash. The distraction caused by mobile phones, vehicle mounted central control display, and navigation display is a growing concern for road safety. Drivers watching the displays while driving slows braking reaction time and makes it difficult to keep in the correct lane and to keep the correct following distances.

In order to reduce the distractions caused by watching display as driving, a projector-camera interactive system can be used to project the display image on transparent glass on the vehicle while users can create and send messages by bare hand touching the projection content directly without any physical clicking devices. People can watch the display information while driving, without causing visual distractions.

The increasing social networking services based on mobile devices are a remarkable trend in mobile

computing [2, 3]. Social networking services can be fused with real-world sensing. Crowdsensing is one kind of fusing [4]. Crowdsensing, as a special form of crowdsourcing [5, 6], leverages human intelligence/experience from the general public and social interactions to create participatory sensor networks to solve specific problems collaboratively [7].

Projector-camera interactive system is also with potential applications to projection-based augmented reality (AR) [8] which directly project augmenting information upon objects in the real world. The vehicle mounted projection-based AR can project traffic information onto the glass screen and overlap the reality. Through the crowdsensing method, drivers can report specific traffic information, such as outside buildings, and store information, road conditions in remote areas, accidents, and so on by the AR interactive system. Then, their information can be automatically aggregated and uploaded to other users in real time.

For the projector-camera interactive systems, many single camera vision-based researches have been carried out. Shah et al. realized finger clicking detection using fingertip path

based on Camshift tracker [9]. However, the method detects clicks using a delay-based scheme, which is not suitable for applications that require fast response. Dai and Chung proposed a touch detection method in which the projection is embedded with imperceptible structured light [10]. He and Cheng realized touch detection method in which a kind of self-adaptive structured light is encoded and embedded into the projection [11]. However, special synchronized and high-speed camera and projector must be used to lock the phase of the embedded structured light. Hu et al. realized a kind of bare finger touch projection interaction by a novel approach based on button distortion [12]. However, the touch precision much depended on the size of button. Cheng et al. proposed a projector-camera interactive system in which touch was detected using a matched pairs of feature points by a projected white circle around the fingertip [13]. However, the additional projected white circle and the shadow of the finger on the projector screen will reduce the accuracy of foreground extraction. Park et al. implemented and evaluated a touch interface for a projection surface in which a depth camera was used for hand recognition and for extracting the hand area from the surrounding scene [14]. But the depth camera is expensive and its precision is affected by surrounding illumination. Zhou et al. proposed a method enabling estimation of the finger depth information using perceptible black-and-white stripe pattern [15]. However, the perceptible black-and-white stripe pattern on the projection screen disturbed the normal display image.

The information cues of the finger's shadow has been used in many kinds of projector-camera interactive systems. Song et al. proposed a handwriting recognition system in which the finger and its shadow were tracked using a camera. Tracks of the finger were used to indicate the direction and location of a document to simulate typical operations [16]. Xu et al. introduced an interactive system based on the shadow derived by the projector [17]. Shadows can provide a simple interface between human and computer systems. Huang et al. proposed a FSM grammar to recognize finger gestures and a shadow-based fingertips detection method [18].

Cai et al. proposed a fingertip touching approach based on the geometric relationship between fingertip and its shadow to estimate the distance from fingertip to projection surface [19]. While these methods require ideal illumination to create shadows that can be well recognized, Dung et al. provided a touch system that used the distance between the finger and its shadow to detect the touch-timing and location [20, 21]. However, an infrared camera and an infrared ray source are necessary in the system which makes the system complicated and costly.

In this paper, a real-time projector-camera interactive system based on the crowdsensing is proposed in which a triangulation method and a linear-scanning method are used to determine the touch for increasing the users' collaboration and operability. These approaches can enhance the robustness and accuracy of the system. The interaction process of the system can be completely carried out automatically, and it can be used as an intelligent device in intelligent transport system. The rest of the paper is organized as follows. In Section 2, the overview of the system is outlined, and

our proposed method for the system including modified multiframe difference method, linear-scanning method, and fingertip detection method is presented. Some experimental results are given in Section 3. Concluding remarks are discussed in Section 4.

## 2. Crowdsensing-Based Interactive System

*2.1. System Overview.* The proposed crowdsensing-based real-time finger interactions system for intelligent transport system consists of a computing device, a projector, a camera, and vehicle sensing devices. For different tasks such as the data collection of outside buildings and store information, road conditions in remote areas, and accidents, specific distributed crowdsourcing platform is established in which incentive mechanism with privacy protection is adopted to attract users to participate in the data collection. A hybrid incentive approach is used in this system for crowdsensing data collection. Monetary reward incentive method is used for the data collection of outside buildings and store information. Entertainment and gamification incentive method is used for the data collection of road conditions in remote areas. Virtual credit incentive method is used for the data collection of accidents.

The collected data is uploaded to the cloud service platform after being preprocessed. The cloud service platform analyzes and processes all the uploaded data. Other users download the information from the cloud service platform and project the information on the front screen of the car by projection-based AR mode. A registration method is used in the projection-based AR system based on projective reconstruction technique using natural features. At first, four points are specified to build the world coordinate which is used to superimpose virtual objects. Next, the live video natural feature is tracked using the Kanade-Lucas-Tomasi feature tracker [22]. The corresponding projective matrix is estimated using the tracked natural features in the image sequence [23]. Then, the registration matrix is computed by transforming the four specified points by the projective reconstruction technique for AR.

The hand casts its shadow on the display screen because of the projector light source, and the movements of the hand and its shadow are tracked by the camera. The system extracts the area of the hand and its shadow from the projection image and estimate the fusion degree of the hand and its shadow when the user interacts with the projector screen. When the hand and its shadow have been detected as being fused completely, the location of the fingertip will be detected and used to determine the location of the touch.

*2.2. Foreground and Shadow Extraction.* There are three approaches for detecting moving foreground target in image sequences: optical flow [24], background subtraction [25], and frames difference [26]. The optical flow method uses the brightness change of the image to extract the motion information of the object. The motion vector of each pixel is used to generate a complete vector field. However, the projection images changed randomly, which makes the optical flow method not suitable in our projection interactive application.

In addition, the optical flow method is quite complex and has poor noise immunity, so it cannot be applied to real-time processing. The background subtraction method use the subtracting between the current frame and a reference image called background to detect the foreground target. The background is found by means of an image selection process. The frames difference approach has the same principle with the background subtraction method, but the previous frame is treated as the reference image. Because of its small computation and high real-time performance, the frames difference method is more commonly used in foreground detection. However, the detection results of frames difference are prone to image smear and holes. In this paper, combining the background subtraction method, we proposed a modified multiframes difference method to detect the foreground and its shadow in the system.

At first, the camera captured image is geometric calibrated to the corresponding projection image in order to obtain the location relationship between the projection image and the camera view. The process can be carried out automatically using geometric calibration on camera captured image. A  $3 \times 3$  matrix  $H$  homography [27] can be used to describe the mapping relationship between a point in the camera view and the projection image:

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix}. \quad (1)$$

The main steps of geometric calibration are described as follows [28]. (1) Project a chessboard to the display screen. (2) Capture the projected images using the camera. (3) Detect the chessboard corners in projection image and captured image, as shown in Figure 1. (4) Estimate the homography between the projection image and captured image using the corresponding detected corner locations in the two chessboard image. To estimate the homography, only 4 points of one rectangle in chessboard are necessary. In this paper as shown in Figure 1, 40 corner points are used to achieve higher accuracy.

Next, the foreground and its shadow will be segmented from the calibrated camera captured image. If image is projected onto the screen directly, the captured current image should be equal to the reference captured front image. If there is a moving foreground in front of the screen, the surface albedo will change. In order to increase the robustness of extracting the foreground and its shadow, block searching is used to obtain the albedo change of pixel  $[x, y]$  which is computed by estimating a set of albedo ratios  $a[x \pm i, y \pm j]$ :

$$a[x \pm i, y \pm j] = \frac{I_k(x \pm i, y \pm j)}{I_{k-1}(x, y)} \quad (2)$$

$(i = 1, 2, \dots, m. j = 1, 2, \dots, n),$

where  $I_k$  is the current frame image and  $I_k(x \pm i, y \pm j)$  is the gray value of pixel  $[x \pm i, y \pm j]$  in  $I_k$ .  $I_{k-1}$  is the previous frame image and  $I_{k-1}(x, y)$  is the gray value of pixel  $[x, y]$  in  $I_{k-1}$ .

Pixel  $[x, y]$  belongs to the foreground region if any one of  $a[x \pm i, y \pm j]$  satisfies

$$a[x \pm i, y \pm j] < 1 - s, \quad (3)$$

or

$$a[x \pm i, y \pm j] > 1 + s, \quad (4)$$

where  $s$  is a constant of 0.5–0.8 as a tolerant scale of the albedo change.

If the hand movement is fast enough, as described above, the hand and its shadow can be extracted using two consecutive frames' difference. If the hand movement is slow, using the difference of two consecutive frames cannot extract the hand and its shadow completely. So, multiframes  $I_k(x, y), I_{k-5}(x, y), I_{k-10}(x, y), I_{k-15}(x, y)$  are used to make differential operation. The extracted foreground and its shadow image can be obtained by

$$T_{kl(l=5,10,15)}(x, y) = \begin{cases} 255, & \text{if } a_l[x \pm i, y \pm j] < 1 - s \text{ or } a_l[x \pm i, y \pm j] > 1 + s \\ 0, & \text{else,} \end{cases} \quad (5)$$

where  $a_l$  is the albedo ratio between the current frame image  $I_k(x, y)$  and previous reference image  $I_{k-l}(x, y)$ , ( $l = 5, 10, 15$ ).

Background subtraction method is combined with multiframes difference method for the proposed interactive projection system, which is basically similar to the frame difference method, but the reference image is background image. The projection image is used as the background image in the modified multiframes difference method. According to the difference between the background image and the geometric calibrated camera captured image, the extracted foreground and its shadow image can be obtained by

$$T'_k(x, y) = \begin{cases} 255, & \text{if } a_b[x \pm i, y \pm j] < 1 - s \text{ or } a_b[x \pm i, y \pm j] > 1 + s \\ 0, & \text{else,} \end{cases} \quad (6)$$

where  $a_b$  is the albedo ratio between the current frame image  $I_k(x, y)$  and background image.

The multiframes difference results  $T_{kl(l=5,10,15)}(x, y)$  are used to perform AND logical operations with  $T'_k(x, y)$ , respectively,

$$\begin{aligned} D_1 &= T_{k5}(x, y) \cap T'_k(x, y), \\ D_2 &= T_{k10}(x, y) \cap T'_k(x, y), \\ D_3 &= T_{k15}(x, y) \cap T'_k(x, y). \end{aligned} \quad (7)$$

Then, perform OR logical operations on  $D_1, D_2,$  and  $D_3$  to obtain  $D$ .

The extracted foreground and its shadow result  $D$  are processed by morphological erosion and dilation [29]. Because of the noise interference, there is often a small amount of color

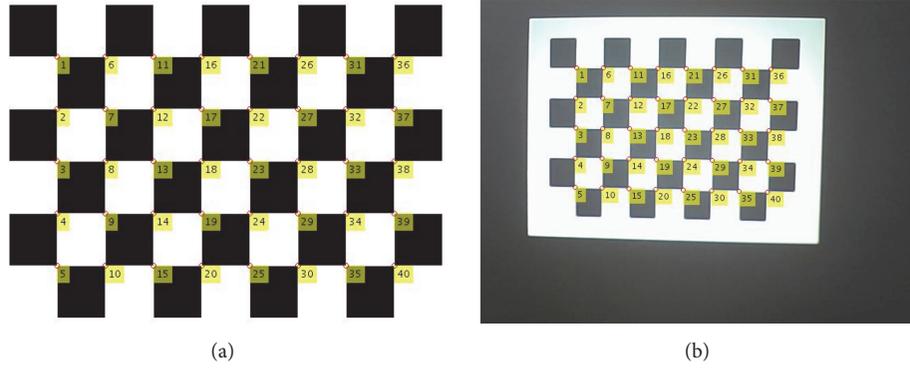


FIGURE 1: (a) The chessboard in projection image and (b) the chessboard in captured image. The detected corners of chessboard are marked with circle for both the chessboard in projection and captured image.

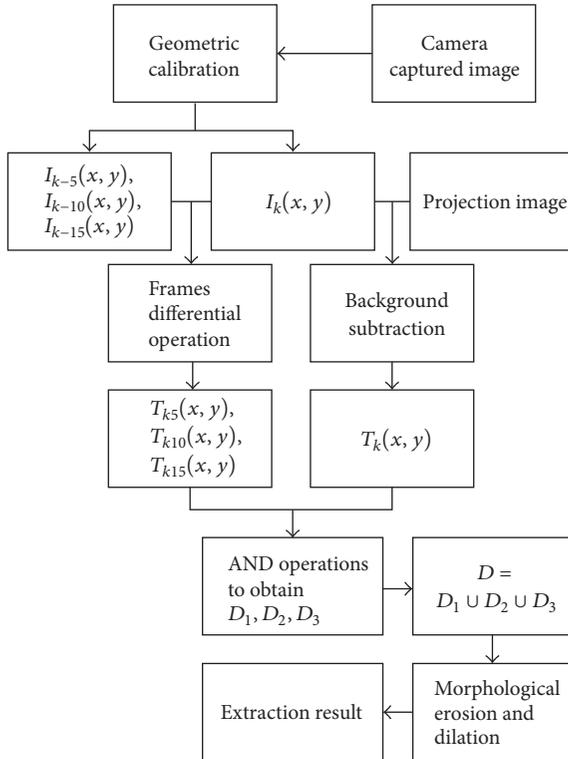


FIGURE 2: Flow chart of the foreground and shadow extraction.

or gray similarity between the target and the reference image. Image  $D$  often has many isolated points and holes, which will interfere the detection of foreground and its shadow. Through erosion and dilation treatment, isolated spots are removed and holes are filled.

The flow chart of the modified multiframes difference method as described above for foreground and shadow extraction is shown in Figure 2. Figure 3(a) is the captured current image with foreground and its shadow; Figure 3(b) is the captured previous 10th image with foreground and its shadow. Figure 3(c) is the captured previous 15th image with foreground and its shadow. Figure 3(d) shows the foreground

and its shadow extraction result using the modified multi-frames difference method.

**2.3. Touch Detection.** The principle of touch detection is based on the triangulation [30] in our interactive projection system. A linear-scanning method is proposed to detect the triangulation of the finger and its shadow without separating the hand from its shadow, which can increase the robustness of the system and reduce the computation cost. Before detecting the triangulation fusion degree of the finger and its shadow, the hand in the extracted image is demarcated to improve the accuracy and processing speed of the fusion detection for the disturbance from the rest of the captured image can be avoided. The pixels of the foreground and its shadow can be detected on the right edge of the extracted images if the user is on the right side of the projector display screen, while they cannot be detected on the left edge of the extracted images, as shown in the left parts of Figures 4(a) and 4(b). Then, a section including the hand from the far-left pixels line of the foreground and its shadow in the extracted image to the right, as shown in the right parts of Figures 4(a) and 4(b), is intercepted as the demarcated area for fusion detection of the finger and its shadow. The pixels of the foreground and its shadow can be detected on the left edge of the extracted images if the user is on the left side of the projector display screen, while they cannot be detected on the right edge of the extracted images, as shown in the left parts of Figures 4(c) and 4(d). Then, a section including the hand from the far-right pixels line of the foreground and its shadow in the extracted image to the left, as shown in the right parts of Figures 4(c) and 4(d), is intercepted as the demarcated area for fusion detection of the finger and its shadow.

A vertical scanning line is used to scan the demarcated area to estimate the fusion degree. If a vertical line is detected on which a 0-pixel segment exists and the pixel value at both ends of the segment is 255, the finger and its shadow have not completely fused, as shown in the right parts of Figures 4(a) and 4(c). Otherwise, if no such vertical line is detected, the finger and its shadow have fused, as shown in the right parts of Figures 4(b) and 4(d).

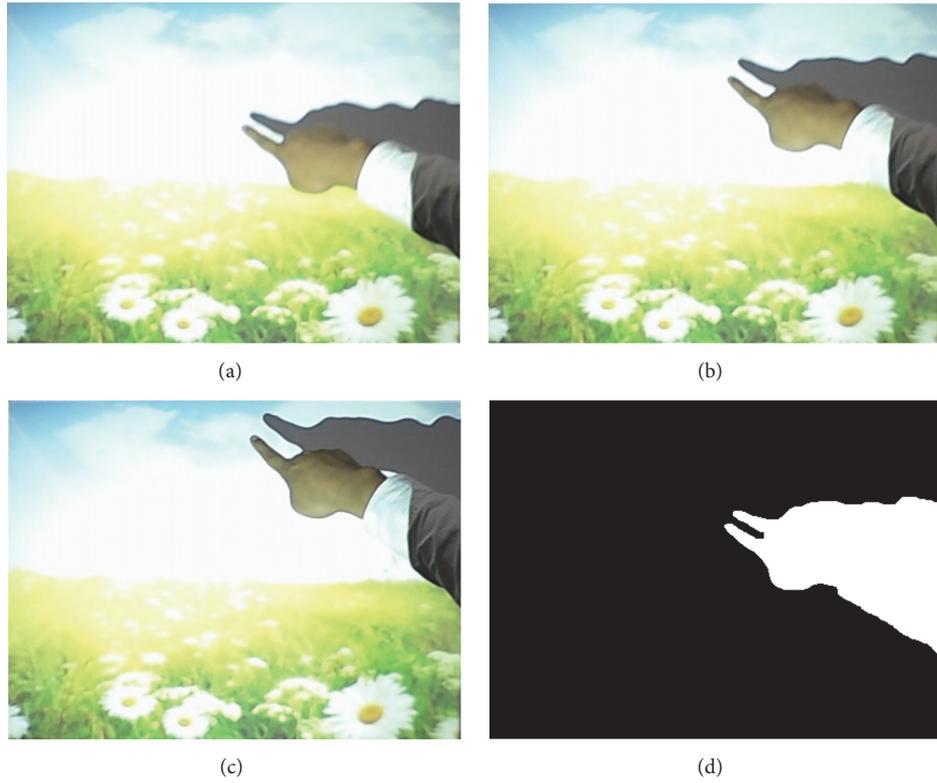


FIGURE 3: Foreground and shadow extraction. (a) The captured current image. (b) The captured previous 10th image. (c) The captured previous 15th image. (d) Foreground and its shadow extraction result.

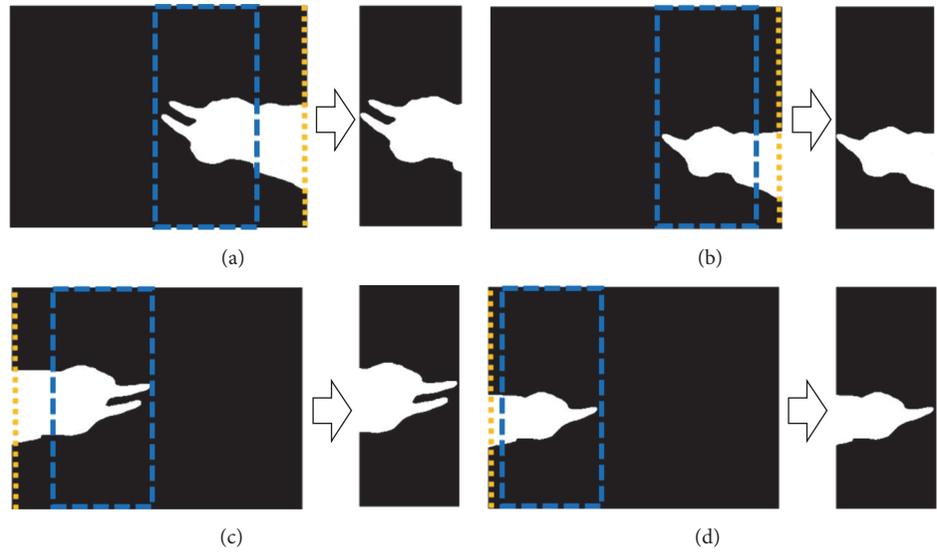


FIGURE 4: Schematic diagram of the linear-scanning method. (a) The finger has not fused with its shadow when the user is on the right. (b) The finger has fused with its shadow when the user is on the right. (c) The finger has not fused with its shadow when the user is on the left. (d) The finger has fused with its shadow when the user is on the left.

When the fingertip and its shadow are detected as being completely fused using the linear-scanning method, the location of the fingertip should be found in the camera captured image to obtain the touch location on the projection

image. According to the characteristic of the user, when the user is on the right side of the projector display screen, the far-left pixel of the foreground in the extracted image is as the fingertip, as shown in Figure 4(b). When the user is on

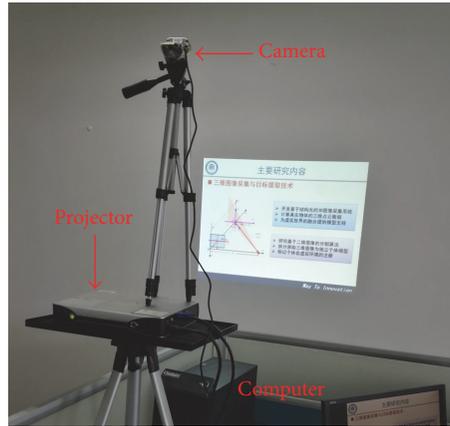


FIGURE 5: The proposed projector-camera interactive system.

the left side of the projector display screen, the far-right pixel of the foreground and its shadow in the extracted image is as the fingertip, as shown in Figure 4(d).

### 3. Result and Discussion

An simulation experiment is conducted to evaluate the performance of our proposed projector-camera interactive system. The experiment platform, as shown in Figure 5, includes a Lenovo computer with 3.6 Ghz CPU and 4.0 G RAM, a  $640 \times 480$  resolution camera, a SONY projector with resolution of  $1440 \times 1050$ , and a projector screen.  $Lps$  is 1.5 m, and  $Lcp$  is 0.5 m.

Figure 6 shows the experimental results of the foreground and its shadow extraction under different lighting. Figures 6(a) and 6(b) are the foreground and its shadow extraction results in natural lighting with bright and dark projection, respectively. Figures 6(c) and 6(d) show experimental results in LED (Light Emitting Diode) lighting with bright and dark projection, respectively. It can be seen that our method is robust in the foreground and its shadow extraction.

The touch depth accuracy is studied in the experiment. When a touch has been detected, the maximum vertical distance from the fingertip to the screen plane is defined as maximum effective touch depth (METD). 9 points on the screen are selected as our METD detection points which are represented with red asterisks as shown in Figure 7. Figure 8 illustrated the METD detection results at the selected 9 points on the projection screen in natural lighting condition with a bright projection. It can be seen from Figure 8 that most of the METDs are below 10 mm except one of 12 mm. The average METD is 8.3 mm. The results demonstrate that our proposed method is effective and it will lead to an accurate fingertip depth detection during interaction.

The randomness and accuracy of touch position on the projection screen are verified in the system. As shown in Figure 9, 220 random points are measured after the finger traverse through the screen. Most of the touch position deviation on the  $X$ - $Y$  screen plane is below 8 mm only except one of 21 mm. The average touch position deviation is 3.3 mm.

The results imply the touch position detection is accurate and unaffected whether the points locate in the center region or at the edge of the screen during the interaction.

Experiment is conducted for touch detection under different lighting conditions, as shown in Table 1. In the experiment, when the METD is below 8 mm and the touch position deviation is below 8 mm, the touch detection is supposed to be correct in normal projector interactions. In Table 1, the touch detection accuracy rates of our method are shown in comparison with Song et al.'s [16] and He and Cheng's [11] method. Among 200 sampled frames, we show the touch detection accuracy rates experimental results in bright and dark projection on natural lighting condition and LED lighting condition, respectively. The touch detection accuracy rate in bright projection on natural lighting condition is the same between our method and Song et al.'s method; both are 98.0%, which is higher than 96.0% of He and Cheng's method. The touch detection accuracy rate in dark projection on natural lighting condition is also the same between our method and Song et al.'s method; both are 88.0%, which is higher than 86.0% of He and Cheng's method, while the touch detection accuracy rate of our method is much higher than Song et al.'s and He and Cheng's method in bright or dark projection on LED lighting condition. The average percentage of correct detection of our method (92.0%) is higher than that of their methods (88.0% and 88.5%).

The execution time of the blocks of the projector-camera interactive system is shown in Table 2. The average time of processing one video frame is 30 milliseconds; the total time variance is 0.6; the total time range is 4. We can detect the touch-timing and location at 33 frames per second. That is to say, our method can achieve real-time computing in an ordinary computing system.

### 4. Conclusion

In this paper, we present a real-time projector-camera interactive system based on the crowdsensing which enables user to transform any flat surface into a virtual touch panel and interact with computer by finger touching. The

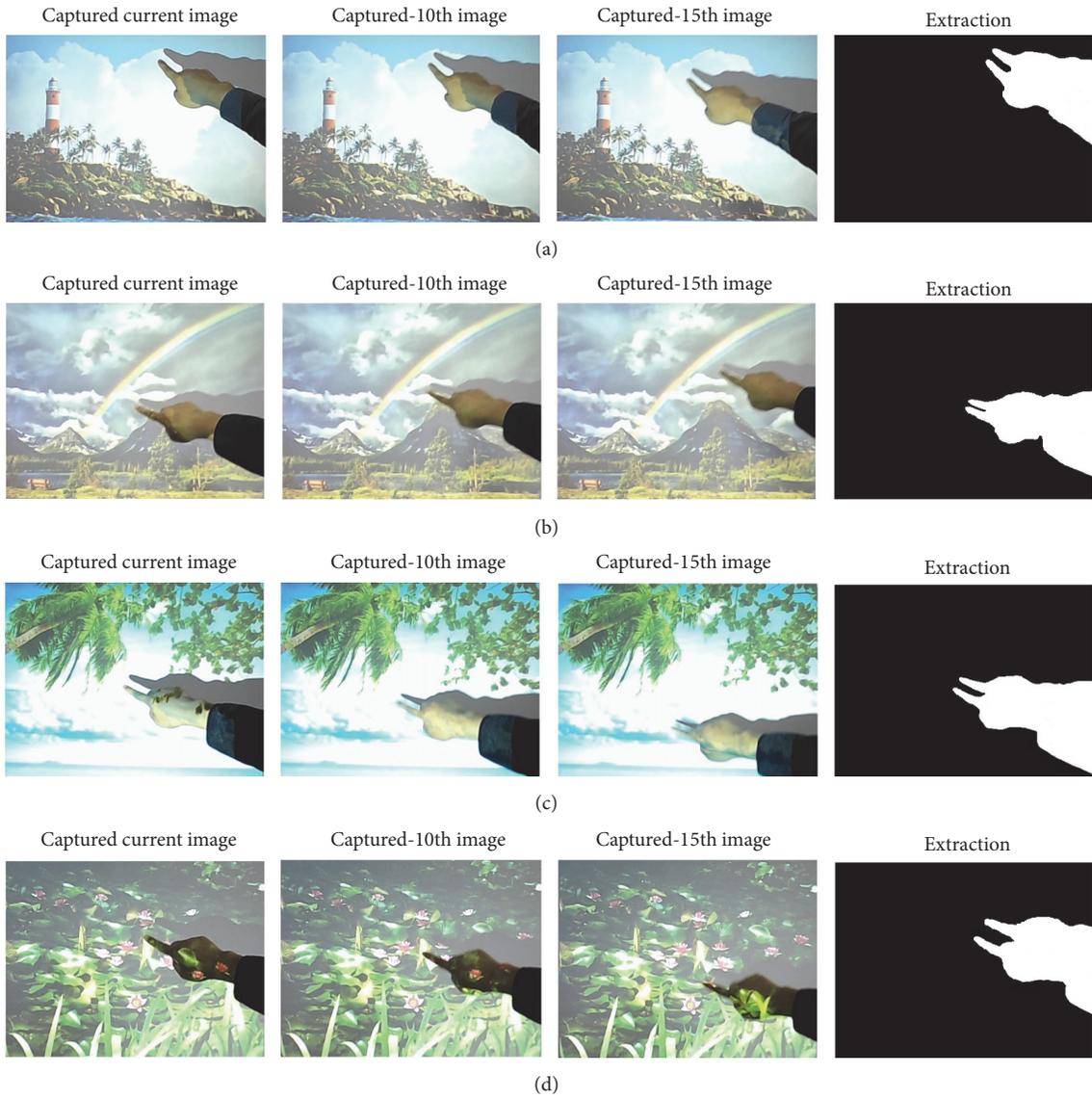


FIGURE 6: The foreground and shadow extraction results under different lighting. (a) is in bright projection on natural lighting, (b) is in dark projection on natural lighting, (c) is in bright projection on LED lighting, and (d) is in dark projection on LED lighting.

interaction process of the system can be completely carried out automatically; it can be used as an intelligent device in intelligent transport system where the driver can watch and interact with the display information while driving, without causing visual distractions. A single camera is used in the system to recover 3D information of fingertip for hand touch detection by the finger and its shadow triangulation. Foreground and shadow are extracted based on a modified multiframes difference method which has both the advantage of the adaptability for projection dynamic environment in the frame difference method and advantage of obtaining complete target information in background subtraction method. Therefore, the interactive system can hardly be affected by the lighting environment, which leads to an accurate interactive gestures extraction and makes the system more adaptable. The fusion degree of the hand and its shadow is detected using a linear-scanning method which avoided the errors caused

by inadequate separation of the finger and its shadow and increased the robustness of the system. Finally, a simple and effective method for detecting the position of the fingertip is proposed according to the user's habit of using the projection interactive system. The experiment results indicate that the projector-camera interactive system can achieve a robust and effective performance using a single camera.

### 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. 61403365 and

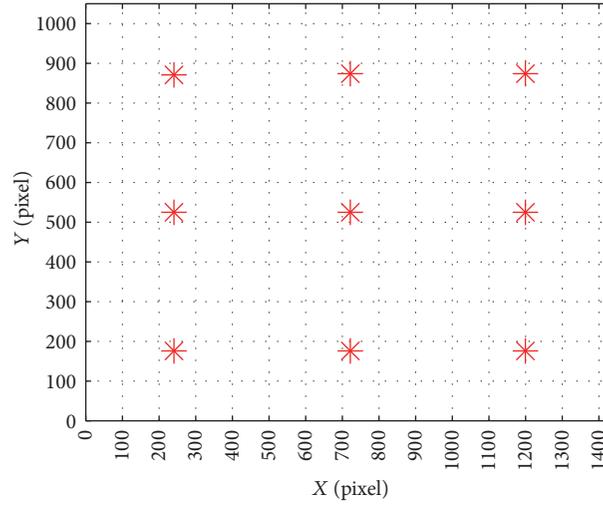


FIGURE 7: Selected 9 points on the projection screen for touch depth detection.

TABLE 1: Touch detection accuracy rates.

Methods	Types	Frames	Each type's accuracy	Average accuracy	
Song et al.'s method	Bright projection on natural lighting	Correct	49	98.0%	88.0%
		No	1		
	Dark projection on natural lighting	Correct	44	88.0%	
		No	6		
	Bright projection on LED lighting	Correct	44	88.0%	
		No	6		
Dark projection on LED lighting	Correct	39	78.0%		
	No	11			
He and Cheng's method	Bright projection on natural lighting	Correct	48	96.0%	88.5%
		No	2		
	Dark projection on natural lighting	Correct	43	86.0%	
		No	7		
	Bright projection on LED lighting	Correct	45	90.0%	
		No	5		
Dark projection on LED lighting	Correct	41	82.0%		
	No	9			
The proposed method	Bright projection on natural lighting	Correct	49	98.0%	92.0%
		No	1		
	Dark projection on natural lighting	Correct	44	88.0%	
		No	6		
	Bright projection on LED lighting	Correct	48	96.0%	
		No	2		
Dark projection on LED lighting	Correct	43	86.0%		
	No	7			

TABLE 2: Execution time measured.

Block	Average time (milliseconds)	Variance	Range
Foreground and shadow extraction	25	0.8	4
Touch detection	5	0.1	1
Total	30	0.6	4

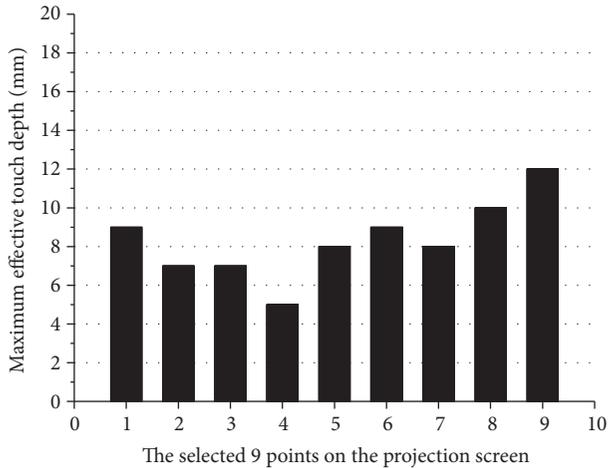


FIGURE 8: Touch depth detection results at the selected 9 points on the projection screen.

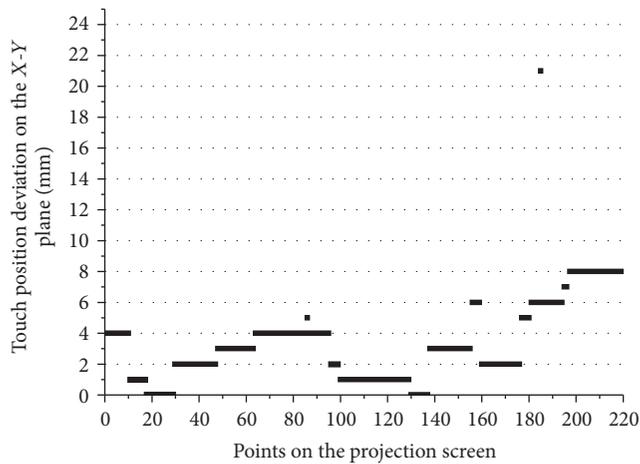


FIGURE 9: Touch position deviation at 220 random points on the X-Y screen plane.

61402458), the China Postdoctoral Science Foundation (Grant no. 2016M602543), the Natural Science Foundation of Guangdong Province, China (Grant nos. 2015A030313744 and 2016A030313177), Shenzhen Technology Project (Grant nos. JSGG20160331185256983 and JSGG20160229115709109), Guangdong Technology Project (Grant nos. 2016B010108010 and 2016B010125003), Shenzhen Engineering Laboratory for 3D Content Generating Technologies (Grant no. [2017]476), Key Laboratory of Human-Machine Intelligence-Synergy Systems, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences (Grant no. 2014DPI73025), State Joint Engineering Laboratory for Robotics and Intelligent Manufacturing funded by National Development and Reform Commission (Grant no. 2015581), and CAS Key Technology Talent Program.

## References

- [1] World Health Organization, "Road traffic injuries," 2017, <http://www.who.int/mediacentre/factsheets/fs358/en/>.

- [2] X. Hu, T. H. S. Chu, V. C. M. Leung, E. C.-H. Ngai, P. Kruchten, and H. C. B. Chan, "A Survey on Mobile Social Networks: Applications, Platforms, System Architectures, and Future Research Directions," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1557–1581, 2015.
- [3] Z. Ning, L. Liu, F. Xia, B. Jedari, I. Lee, and W. Zhang, "CAIS: a copy adjustable incentive scheme in community-based socially aware networking," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 3406–3419, 2017.
- [4] C. T. Calafate, C. Wu, E. Natalizio, and F. J. Martínez, "Crowdsensing and Vehicle-Based Sensing," *Mobile Information Systems*, vol. 2016, Article ID 2596783, p. 2, 2016.
- [5] J. Howe, "The rise of crowdsourcing," *Wired Magazine*, vol. 14, no. 6, pp. 1–4, 2006.
- [6] Q. Li, P. Yang, S. Tang, M. Zhang, and X. Fan, "Equilibrium is priceless: selfish task allocation for mobile crowdsourcing network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, pp. 1–8, 2016.
- [7] X. Hu, V. C. Leung, Y. Kwok et al., "SafeDJ: a Crowd-Cloud Co-design Approach to Situation-aware Music Delivery for Drivers," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 12, no. 1s, p. 21, 2015.
- [8] P. Beardsley, J. V. Baar, R. Raskar, and C. Forlines, "Interaction using a handheld projector," *IEEE Computer Graphics and Applications*, vol. 25, no. 1, pp. 39–43, 2005.
- [9] S. A. Shah, A. Ahmed, I. Mahmood, and K. Khurshid, "Hand gesture based user interface for computer using a camera and projector," in *Proceedings of the 2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pp. 168–173, Kuala Lumpur, Malaysia, November 2011.
- [10] J. Dai and R. Chung, "Making any planar surface into a touch-sensitive display by a mere projector and camera," in *Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*, pp. 35–42, Providence, RI, USA, June 2012.
- [11] M. He and J. Cheng, "Self-adaptive coding-based touch detection for interactive projector system," in *Proceedings of the 2014 4th IEEE International Conference on Information Science and Technology (ICIST)*, pp. 656–659, Shenzhen, China, April 2014.
- [12] J. Hu, G. Li, X. Xie, Z. Lv, and Z. Wang, "Bare-fingers Touch Detection by the Button's Distortion in a Projector-Camera System," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 4, pp. 566–575, 2014.
- [13] J. Cheng, Q. Wang, R. Song, and X. Wu, "Fingertip-based interactive projector-camera system," *Signal Processing*, vol. 110, pp. 54–66, 2015.
- [14] Y.-S. Park, S.-H. Park, T.-G. Kim, and J.-M. Chung, "Implementation of Gesture Interface for Projected Surfaces," *KSII Transactions on Internet and Information Systems*, vol. 9, no. 1, pp. 378–390, 2015.
- [15] Q. Zhou, X. Qiao, K. Ni, X. Li, and X. Wang, "Depth detection in interactive projection system based on one-shot black-and-white stripe pattern," *Optics Express*, vol. 25, no. 5, pp. 5341–5351, 2017.
- [16] P. Song, S. Winkler, S. O. Gilani, and Z. Y. Zhou, in *Proceedings of the IEEE International Workshop on Human-Computer Interaction*, vol. 4796, pp. 49–58, Springer, Rio de Janeiro, Brazil.
- [17] H. Xu, D. Iwai, S. Hiura, and K. Sato, "User Interface by Virtual Shadow Projection," in *Proceedings of the 2006 SICE-ICASE International Joint Conference*, pp. 4814–4817, IEEE, Busan, Korea, October 2006.

- [18] J.-Y. Huang, Y.-Z. Yeo, L. Huei, and C.-H. Tsai, "Image-Based Wearable Tangible Interface," in *Advances in Intelligent Systems and Applications - Volume 2*, vol. 21 of *Smart Innovation, Systems and Technologies*, pp. 455–464, Springer, Berlin, Germany, 2013.
- [19] X. Cai, X. Xie, G. Li, W. Song, Y. Zheng, and Z. Wang, "A new method of detecting fingertip touch for the projector-camera HCI system," in *Proceedings of the 2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 526–529, IEEE, Melbourne, Australia, June 2014.
- [20] L.-R. Dung, G.-Y. Lai, and Y.-Y. Wu, "Shadow touching for interactive projectors," in *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1798–1802, IEEE, Vancouver, Canada, May 2013.
- [21] L.-R. Dung and R.-Y. Huang, "A Shadow Touching Technique for Interactive Projector Devices," in *HCI International 2013 - Posters' Extended Abstracts*, vol. 373 of *Communications in Computer and Information Science*, pp. 308–312, Springer, Berlin, Germany, 2013.
- [22] B. Georgescu and P. Meer, "Point matching under large image deformations and illumination changes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 674–688, 2004.
- [23] M. L. Yuan, S. K. Ong, and A. Y. C. Nee, "Registration using natural features for augmented reality systems," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 569–580, 2006.
- [24] A. Agarwal, S. Gupta, and D. K. Singh, "Review of optical flow technique for moving object detection," in *Proceedings of the 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, pp. 409–413, Greater Noida, India, October 2016.
- [25] G. Gemignani and A. Rozza, "A Robust Approach for the Background Subtraction Based on Multi-Layered Self-Organizing Maps," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5239–5251, 2016.
- [26] K. Wang, L. Xu, Y. Fang, and J. Li, "One-against-all frame differences based hand detection for human and mobile interaction," *Neurocomputing*, vol. 120, pp. 185–191, 2013.
- [27] Z. Zhang, "Computer Vision Technologies for Remote Collaboration Using Physical Whiteboards, Projectors and Cameras," in *Proceedings of the Computer Vision for Interactive and Intelligent Environment (CVIIIE'05)*, IEEE, Lexington, KY, USA, November 2005.
- [28] M. Liao, R. Yang, and Z. Zhang, "Robust and Accurate Visual Echo Cancellation in a Full-duplex Projector–Camera System," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 10, pp. 1831–1840, 2008.
- [29] P. Soille, *Morphological Image Analysis—Principles and Applications*, Springer, Berlin, Heidelberg, Germany, 2nd edition, 2004.
- [30] Y. Wang and S. Zhang, "Optimal fringe angle selection for digital fringe projection technique," *Applied Optics*, vol. 52, 7094, no. 29, p. 7098, 2013.

## Research Article

# Optimal Computing Resource Management Based on Utility Maximization in Mobile Crowdsourcing

Haoyu Meng,<sup>1</sup> Ying Zhu,<sup>2</sup> and Ruilong Deng<sup>3</sup>

<sup>1</sup>Computer Science Department, Wenzhou Vocational & Technical College, Wenzhou 325035, China

<sup>2</sup>Department of Science & Technology, Zhejiang Institute of Economics and Trade, Hangzhou 310018, China

<sup>3</sup>State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China

Correspondence should be addressed to Ruilong Deng; dengruilong@gmail.com

Received 7 June 2017; Accepted 18 July 2017; Published 17 September 2017

Academic Editor: Zhaolong Ning

Copyright © 2017 Haoyu Meng 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 crowdsourcing, as an emerging service paradigm, enables the computing resource requestor (CRR) to outsource computation tasks to each computing resource provider (CRP). Considering the importance of pricing as an essential incentive to coordinate the real-time interaction among the CRR and CRPs, in this paper, we propose an optimal real-time pricing strategy for computing resource management in mobile crowdsourcing. Firstly, we analytically model the CRR and CRPs behaviors in form of carefully selected utility and cost functions, based on concepts from microeconomics. Secondly, we propose a distributed algorithm through the exchange of control messages, which contain the information of computing resource demand/supply and real-time prices. We show that there exist real-time prices that can align individual optimality with systematic optimality. Finally, we also take account of the interaction among CRPs and formulate the computing resource management as a game with Nash equilibrium achievable via best response. Simulation results demonstrate that the proposed distributed algorithm can potentially benefit both the CRR and CRPs. The coordinator in mobile crowdsourcing can thus use the optimal real-time pricing strategy to manage computing resources towards the benefit of the overall system.

## 1. Introduction

With the explosion of mobile devices, mobile computing has become an overwhelming trend in the development of mobile networks and Internet of things (IoT) [1–3]. However, mobile devices are facing some limitations on various resources, for example, computation, memory, and energy [4–6]. To overcome these limitations, mobile cloud computing has become a promising solution to enable mobile devices to use cloud server resources via wireless communications and networking [7–9]. Such mobile devices, that is, mobile as a computing resource requestor (CRR), can improve the computation capability and energy efficiency by offloading computation tasks to cloud servers.

On the other hand, with the proliferation of increasingly powerful mobile devices, mobile users can collaboratively form a mobile cloud to provide pervasive services, such as data collection, processing, and computing [10–12].

Empowered by these capabilities, mobile devices shift from service consumers to service providers, that is, mobile as a computing resource provider (CRP) [13–16]. With the unprecedentedly powerful mobile cloud, mobile crowdsourcing has been gaining momentum as a viable platform for solving very large-scale problems with the help of the masses [17–19]. By outsourcing computation tasks to the crowd, cost-effective and pervasive computing services can be achieved at a societal scale, using a possibly large number of mobile devices to collaborate together in a distributed way.

Resource management is one of the fundamental issues in mobile networks and IoT for network utility maximization (NUM) [20–22]. In mobile crowdsourcing, the CRR wants to adjust its demand amount of computing resources so as to pursue the maximum benefit, while each CRP wants to adjust its supply amount of computing resources so as to pursue the maximum benefit. Given the importance of computing resource management, in this paper, we focus

on the real-time interaction among the CRR and CRPs and introduce an optimal real-time pricing strategy. At first, we analytically model the CRR and CRPs behaviors in form of carefully selected utility and cost functions, based on concepts from microeconomics. Then, we formulate the computing resource management in mobile crowdsourcing as an optimization problem to maximize the satisfaction of the CRR while minimizing the aggregate expense of all CRPs in the system. We show that there exist real-time prices that can align individual optimality with systematic optimality, that is, under such prices, when the CRR and CRPs selfishly optimize their own benefits, they also automatically maximize the system welfare. In order to preserve everyone's privacy, we propose a distributed algorithm for the CRR and CRPs to jointly determine the optimal real-time prices and computing resource demand/supply. The coordinator in mobile crowdsourcing can thus use the optimal real-time pricing strategy to manage computing resources towards the benefit of the overall system. Finally, we take account of the interaction among CRPs and formulate the computing resource management as a game with Nash equilibrium achievable via best response.

To the best of our knowledge, this is an early effort towards providing a systematic framework of optimal computing resource management in mobile crowdsourcing. We hope that this pioneering work can throw light on coordinating the real-time interaction among the CRR and CRPs via the optimal real-time pricing strategy. Specifically, the original contributions of this paper are summarized as follows:

- (1) We model the CRR and CRPs behaviors and formulate the computing resource management in mobile crowdsourcing as an optimization problem. The proposed distributed algorithm does not require anyone to reveal its private information.
- (2) We take account of the interaction among CRPs and formulate the computing resource management as a game with Nash equilibrium achievable via best response.
- (3) Simulation results demonstrate that both the CRR and CRPs will benefit from the proposed algorithm.

The remainder of this paper is organized as follows. The system model is introduced in Section 2. We solve the problem of the one-CRR and one-CRP case in Section 3 and that of the one-CRR and multi-CRP case in Section 4. Numerical results are illustrated in Section 5 before conclusions drawn in Section 6.

## 2. System Model

Consider a mobile crowdsourcing system consisting of one CRR, one or multiple CRPs, and a coordinator. The time cycle is divided into  $T$  time slots. In a certain time slot, let  $d$  denote the computing resource demand of the CRR; Similarly, let  $s_i$  denote the computing resource supply of the CRP  $i$  ( $i = 1, 2, \dots, n$ , where  $n$  is the total number of CRPs).

(1) *The Utility Function for the CRR.* The computing resource demand of the CRR depends on the resource price and the priority of the task, which can be modeled by the utility function. Specifically, the utility function  $U(\cdot)$  represents the satisfaction obtained by the CRR as a function of the computing resource demand  $d$ , which is nondecreasing and concave. In this paper, we consider the following quadratic utility function:

$$U(d) = \begin{cases} \omega d - \frac{\alpha}{2} d^2 & 0 \leq d \leq \frac{\omega}{\alpha} \\ \frac{\omega^2}{\alpha} & d \geq \frac{\omega}{\alpha}, \end{cases} \quad (1)$$

where  $\omega, \alpha > 0$  are predetermined parameters. It corresponds to a linear decreasing marginal benefit  $V(d) \triangleq \partial U(d)/\partial d = \omega - \alpha d \geq 0$  when  $0 \leq d \leq \omega/\alpha$ .

(2) *The Cost Function for the CRP.* The cost function  $C_i(\cdot)$  represents the expense of supplying the computing resource  $s_i$  by the CRP  $i$ , which is increasing and strictly convex. In this paper, we consider the following quadratic cost function:

$$C_i(s_i) = a_i s_i^2 + b_i s_i + c_i, \quad (2)$$

where  $a_i > 0$  and  $b_i, c_i \geq 0$  are predetermined parameters.

## 3. One-CRR and One-CRP Case

At the beginning, consider a simple case for a mobile crowdsourcing system consisting of one CRR, one CRP, and a coordinator. We formulate the interaction among the CRR, CRP, and coordinator as local and global optimization problems and obtain the solution in a distributed way. The resource price is taken as an incentive to reach the balance between the supply and demand, as well as the maximum benefit of both the CRR and CRP.

*3.1. Problem Formulation.* Firstly, in a certain time slot, under the resource price  $p$ , the benefit of the CRR by demanding the computing resource  $d$  is calculated as  $B_r(d) = U(d) - pd$ . The CRR wants to adjust its demand amount of computing resources so as to pursuit the maximum benefit. Thus the local optimization problem from the CRR perspective is

$$\max_d U(d) - pd. \quad (3)$$

Similarly, in a certain time slot, under the resource price  $p$ , the benefit of the CRP by supplying the computing resource  $s$  is calculated as  $B_p(s) = ps - C(s)$ . The CRP wants to adjust its supply amount of computing resources so as to pursuit the maximum benefit. Thus the local optimization problem from the CRP perspective is

$$\max_s ps - C(s). \quad (4)$$

From the system perspective, it is desirable that the satisfaction of the CRR is maximized and the expense of the CRP is minimized. Mathematically, we define the system welfare as  $W(d, s) = U(d) - C(s)$ , where  $s \geq d$ . The individually optimal solution may not be systematically optimal under an arbitrary resource price. We take the utility function minus the cost function as the objective with the constraint that the supply should be at least equal to the demand. Thus the global optimization problem from the system perspective is

$$\begin{aligned} \max_{d,s} \quad & U(d) - C(s) \\ \text{s.t.} \quad & s \geq d. \end{aligned} \quad (5)$$

Note that the problem is a concave maximization problem, which can be solved by convex optimization techniques in a centralized way [23]. For example, we take the constraint into the objective by a Lagrangian multiplier. Thus the Lagrangian is defined as

$$\begin{aligned} L(d, s, \lambda) &= U(d) - C(s) + \lambda(s - d) \\ &= [U(d) - \lambda d] + [\lambda s - C(s)], \end{aligned} \quad (6)$$

where  $\lambda$  denotes the Lagrangian multiplier associated with the constraint in (5). By means of  $\partial L / \partial d = \partial L / \partial s = \partial L / \partial \lambda = 0$  for (6), the coordinator wants to adjust the resource price (later we will show that  $\lambda$  can be interpreted as the resource price) so as to pursuit the maximum benefit of both the CRR and CRP.

However, the arising challenge is that the coordinator needs to know the exact utility function of the CRP and cost function of the CRP. Since such information is private and no one wants to reveal any, the coordinator may not have sufficient information to solve problem (5). Nevertheless, the distributed algorithm will not require the coordinator to know the exact utility function of the CRP and cost function of the CRP and thus preserves their privacy. Therefore, in the following, we will present the distributed algorithm to approach the optimal resource price for computing resource management in mobile crowdsourcing.

**3.2. Lagrange Duality.** In order to solve problem (5) in a distributed way, we define the dual function as the maximum value of (6) over  $d, s$ :

$$\begin{aligned} D(\lambda) &= \sup_{d,s} L(d, s, \lambda) \\ &= \sup_d [U(d) - \lambda d] + \sup_s [\lambda s - C(s)]. \end{aligned} \quad (7)$$

Furthermore, the Lagrange dual problem is

$$\begin{aligned} \min_{\lambda} \quad & D(\lambda) \\ \text{s.t.} \quad & \lambda > 0. \end{aligned} \quad (8)$$

We can solve the dual problem (8) instead of the primal problem (5) [23].

Comparing (7) with (3) and (4), we find that the Lagrangian multiplier  $\lambda$  can be replaced by the resource price  $p$  to pursuit the global optimum. In this way, the dual problem can be decomposed into two separable subproblems: one is in the form of (3), which can be locally solved by the CRR, while the other is in the form of (4), which can be locally solved by the CRP. After both the CRR and CRP solve their own local optimization problem to obtain  $d^*$  and  $s^*$ , the coordinator can solve the dual problem (8) to obtain  $p^*$ , which guarantees the constraint  $s^* \geq d^*$ , such that the locally optimal solution will become globally optimal.

Firstly, for the CRR, the locally optimal solution to (3) is

$$\begin{aligned} \frac{\partial B_r(d)}{\partial d} = \frac{\partial U(d)}{\partial d} - p = 0 &\implies \\ d^* &= d(p). \end{aligned} \quad (9)$$

Similarly, for the CRP, the locally optimal solution to (4) is

$$\begin{aligned} \frac{\partial B_p(s)}{\partial s} = p - \frac{\partial C(s)}{\partial s} = 0 &\implies \\ s^* &= s(p). \end{aligned} \quad (10)$$

Taking the resource price  $p$  in place of the Lagrangian multiplier  $\lambda$ , together with the locally optimal solution  $d^*$  and  $s^*$ , we rewrite the dual problem (8) as

$$\min_{p>0} D(p) = [U(d^*) - pd^*] + [ps^* - C(s^*)]. \quad (11)$$

The globally optimal solution to (11) is

$$\begin{aligned} \frac{\partial D(p)}{\partial p} = \frac{\partial U(d^*)}{\partial d^*} \frac{\partial d^*}{\partial p} - d^* - p \frac{\partial d^*}{\partial p} + s^* + p \frac{\partial s^*}{\partial p} \\ - \frac{\partial C(s^*)}{\partial s^*} \frac{\partial s^*}{\partial p} = 0. \end{aligned} \quad (12)$$

Note that from (9) and (10) we have  $\partial U(d^*) / \partial d^* = \partial C(s^*) / \partial s^* = p$ , so

$$\frac{\partial D(p)}{\partial p} = s^* - d^* = 0. \quad (13)$$

Overall, by jointly solving (9), (10), and (13), we can obtain the optimal resource price  $p^*$  and further calculate the specific  $d^*$  and  $s^*$ , such that both the local and global optimums are achieved. In other words, the coordinator wants to make the locally optimal solution be globally optimal by adjusting the resource price.

3.3. *Distributed Solution.* In order to preserve everyone's privacy, it is possible to approach the optimal resource price of the dual problem (11) in a distributed way [24–26]:

- (1) The coordinator begins with any initial resource price  $p^k \geq 0$  ( $k \in \mathbb{N}^+$  is the iteration index) and announces it to both the CRR and CRP.
- (2) On receiving the resource price  $p^k$ , the CRR updates its demand amount of computing resources  $d^k$  by solving the local optimization problem  $d^k = \arg \max_d [U(d) - p^k d]$  and feeds it back to the coordinator. Similarly, on receiving the resource price  $p^k$ , the CRP also updates its supply amount of computing resources  $s^k = \arg \max_s [p^k s - C(s)]$  and feeds it back to the coordinator too.
- (3) On receiving the local optimal computing resource demand  $d^k$  and supply  $s^k$ , the coordinator updates the resource price  $p^{k+1}$  for the next iteration using the following gradient projection method:

$$p^{k+1} = \left[ p^k - \gamma \frac{\partial D(p^k)}{\partial p^k} \right]^+ = [p^k - \gamma (s^k - d^k)]^+, \quad (14)$$

where  $\gamma > 0$  is the step size to adjust the convergence rate and  $[x]^+$  represents the larger one between  $x$  and 0.

- (4) Repeat from (1) to (3) until the resource price remains unchanged.

With the sufficiently small step size  $\gamma$ , the proposed distributed algorithm converges to the globally optimal supply  $s^*$  and demand  $d^*$ , as long as the primal problem (5) is feasible. The reason is as follows. Since the utility function  $U(\cdot)$  is concave and the cost function  $C(\cdot)$  is convex, and thus the relationship from the supply  $s$  and demand  $d$  to the price  $p$  is monotone, there exists the sufficiently small step size  $\gamma$  that guarantees the convergence of the gradient projection method [27]. The proposed distributed algorithm converges when  $0 < \gamma < 2/K$ , where  $K$  is the Lipschitz constant for the dual function:  $\|\nabla D(p_1) - \nabla D(p_2)\|_2 \leq K \|p_1 - p_2\|_2$ . The convergence of the following proposed distributed algorithms is similar, which will be omitted due to the space limitation.

The interaction among the CRR, CRP, and coordinator is shown in Figure 1. Intuitively, from (14), if the supply is larger than the demand, that is,  $s^k > d^k$ , the coordinator will drop the price, that is,  $p^{k+1} < p^k$ ; otherwise, if the supply is less than the demand, that is,  $s^k < d^k$ , the coordinator will rise the price, that is,  $p^{k+1} > p^k$ . In this way the iteration will converge to the globally optimal price which balances between the supply and demand.

Note that the globally optimal price which balances between the supply and demand will also achieve the

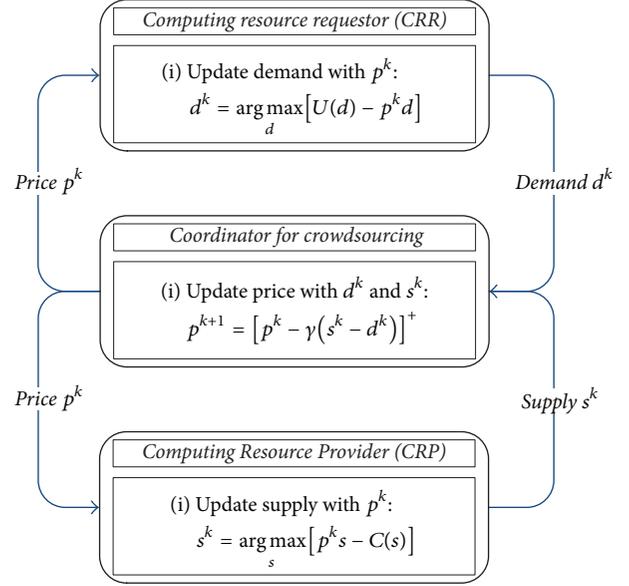


FIGURE 1: Interaction among CRR, CRP, and coordinator.

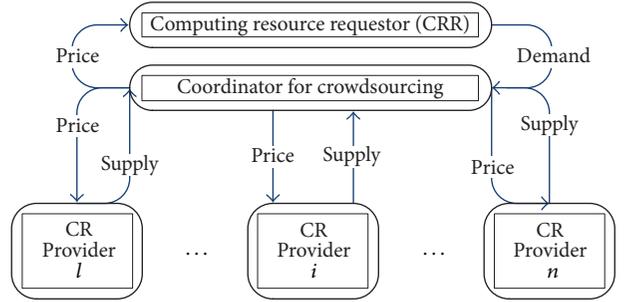


FIGURE 2: Without interaction among CRPs.

maximum benefit of both the CRR and CRP. Otherwise, if the supply is less than the demand, the exceeded computing resource demanded by the CRR will not be satisfied, which reduces its benefit; similarly, if the supply is larger than the demand, the exceeded computing resource supplied by the CRP will be wasted, which reduces its benefit too.

## 4. One-CRR and Multi-CRP Case

Now, consider another case for a mobile crowdsourcing system consisting of one CRR, multiple CRPs, and a coordinator.

4.1. *Without Interaction among CRPs.* Firstly, we focus on the interaction only between the coordinator and each CRP; that is, each CRP is expected to respond to the resource price announced by the coordinator. Under this paradigm, each CRP only communicates with the coordinator as depicted in Figure 2, without the interaction among CRPs.

The problem formulation of this case is similar to that in Section 3.1. The local optimization problem from the CRR

perspective is the same as (3). Similarly, in a certain time slot, under the resource price  $p$ , the benefit of each CRP by supplying the computing resource  $s_i$  is calculated as

$$B_p^i(s_i) = ps_i - C_i(s_i). \quad (15)$$

Each CRP wants to adjust its supply amount of computing resources so as to pursue the maximum benefit. Thus the local optimization problem from each CRP perspective is

$$\min_{s_i} ps_i - C_i(s_i). \quad (16)$$

From the system perspective, it is desirable that the satisfaction of the CRR is maximized and the sum of the expense of all CRPs is minimized. We take the utility function minus the sum of all cost functions as the objective with the constraint that the total supply should be at least equal to the demand. Thus the global optimization problem from the system perspective is

$$\max_{d,s} U(d) - \sum_{i=1}^n C_i(s_i) \quad (17)$$

$$\text{s.t.} \quad \sum_{i=1}^n s_i \geq d.$$

Note that the problem is a concave maximization problem, which can be solved by convex optimization techniques in a centralized way; for example, the Lagrangian is defined as

$$\begin{aligned} L(d, \mathbf{s}, \lambda) &= U(d) - \sum_{i=1}^n C_i(s_i) + \lambda \left( \sum_{i=1}^n s_i - d \right) \\ &= [U(d) - \lambda d] + \sum_{i=1}^n [\lambda s_i - C_i(s_i)], \end{aligned} \quad (18)$$

where  $\lambda$  is the Lagrangian multiplier associated with the constraint in (17).

Similarly, in order to solve problem (17) in a distributed way, we define the dual function as the maximum value of (18) over  $d, \mathbf{s}$ :

$$\begin{aligned} D(\lambda) &= \sup_{d,s} L(d, \mathbf{s}, \lambda) \\ &= \sup_d [U(d) - \lambda d] + \sum_{i=1}^n \sup_{s_i} [\lambda s_i - C_i(s_i)]. \end{aligned} \quad (19)$$

Furthermore, the Lagrange dual problem is

$$\min_{p>0} D(p) = [U(d^*) - pd^*] + \sum_{i=1}^n [ps_i^* - C_i(s_i^*)]. \quad (20)$$

We can solve the dual problem (20) instead of the primal problem (17) [23].

Similarly, in order to preserve everyone's privacy, it is possible to approach the optimal resource price of the dual problem (20) in a distributed way, which is similar to that in Section 3.3 [28–30]:

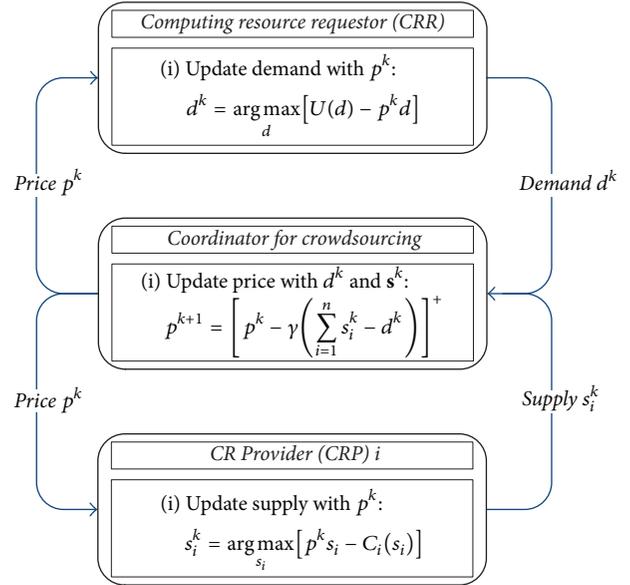


FIGURE 3: Interaction among CRR, each CRP, and coordinator.

- (1) The coordinator begins with any initial resource price  $p^k \geq 0$  and announces it to the CRR and all CRPs.
- (2) On receiving the resource price  $p^k$ , the CRR updates its demand amount of computing resources  $d^k = \arg \max_d [U(d) - p^k d]$  and feeds it back to the coordinator; similarly, on receiving the resource price  $p^k$ , each CRP also updates its supply amount of computing resources  $s_i^k$  by solving the local optimization problem  $s_i^k = \arg \max_{s_i} [p^k s_i - C_i(s_i)]$  and feeds it back to the coordinator too.
- (3) On receiving the local optimal computing resource demand  $d^k$  and supply  $s_i^k$ , the coordinator updates the resource price  $p^{k+1}$  for the next iteration using the following gradient projection method:

$$\begin{aligned} p^{k+1} &= \left[ p^k - \gamma \frac{\partial D(p^k)}{\partial p^k} \right]^+ \\ &= \left[ p^k - \gamma \left( \sum_{i=1}^n s_i^k - d^k \right) \right]^+. \end{aligned} \quad (21)$$

- (4) Repeat from (1) to (3) until the resource price remains unchanged.

The interaction among the CRR, each CRP, and coordinator is shown in Figure 3.

**4.2. With Interaction among CRPs.** Rather than focusing only on how each CRP behaves individually, we propose a framework with interaction among CRPs via message exchanges; for example, each CRP may share the information of its computing resource supply  $s_i$  among others. As

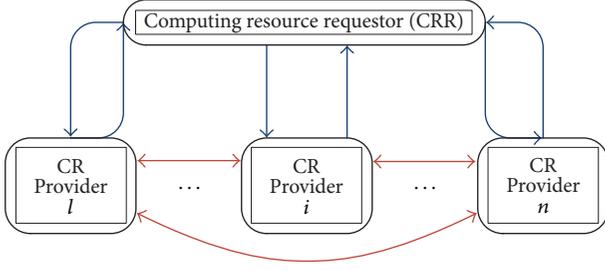


FIGURE 4: With interaction among CRPs.

depicted in Figure 4, the blue arrows represent the two-way communication between the CRR and each CRP, while the red bidirectional arrows correspond to the interaction among CRPs.

From (9) we know that  $p = \partial U(d)/\partial d$  is the local optimal resource price for the CRR. From the above we also know that the global optimum exists at  $d = \sum_{i=1}^n s_i$ , by means of  $\partial L/\partial \lambda = 0$  for (18). Therefore, in a certain time slot, if each CRP supplies the computing resource  $s_i$ , then the local optimal resource price for the CRR is calculated as

$$p = \left. \frac{\partial U(d)}{\partial d} \right|_{d=\sum_{i=1}^n s_i} \triangleq U' \left( \sum_{i=1}^n s_i \right). \quad (22)$$

**4.2.1. Game Theory.** Game theory is a study of selfish and rational players and a formal model of interactive decision-making situation [31]. A game  $G = \{N, S, \{P_i(\cdot)\}\}$  consists of the following three components [32].

(1) *Players.*  $N = \{1, 2, \dots, n\}$  is a finite set of players, where  $n$  is the total number of players in the game.

(2) *Strategies.*  $S = \times_i^n S_i$  is the strategy space of the game and each player  $i$  chooses a strategy  $s_i$  from its strategy set  $S_i$ . Generally we denote a strategy vector by  $\mathbf{s} = (s_i, \mathbf{s}_{-i})$ , where  $\mathbf{s}_{-i} \triangleq [s_1, \dots, s_{i-1}, s_{i+1}, \dots, s_n]$  are the strategies chosen by all the other players in the game.

(3) *Payoff Functions.*  $\{P_i(\cdot)\}$  is a finite set of payoff functions. The payoff  $P_i$  of the player  $i$  is determined by the strategy vector  $\mathbf{s}$ . Each selfish and rational player wishes to choose the optimal strategy  $s_i$  according to the other players' strategies  $\mathbf{s}_{-i}$  to maximize its own payoff  $P_i(s_i, \mathbf{s}_{-i})$ .

Nash equilibrium (NE) is the most important concept of equilibrium condition in game theory. NE is such a static stable strategy vector that no player has any benefit from unilaterally deviating from this strategy. A strategy vector  $\mathbf{s}^* = (s_i^*, \mathbf{s}_{-i}^*)$  is called NE if and only if  $P_i(\mathbf{s}^*) \geq P_i(s_i, \mathbf{s}_{-i}^*)$ ,  $\forall i \in N$ ,  $\forall s_i \in S_i$ .

**Theorem 1.** A game can be shown to have NE if the following conditions are satisfied [33]:

- (1) The player set is finite.
- (2) The strategy sets are closed, bounded, and convex.

(3) The payoff functions are continuous in strategy space and quasi-concave.

An S-modular game restricts payoff function  $\{P_i(\cdot)\}$  such that  $\forall i \in N$  either (23a) or (23b) is satisfied.

$$\frac{\partial^2 P_i(s)}{\partial s_i \partial s_j} \geq 0 \quad \forall j \neq i \in N \quad (23a)$$

$$\frac{\partial^2 P_i(s)}{\partial s_i \partial s_j} \leq 0 \quad \forall j \neq i \in N. \quad (23b)$$

When (23a) is satisfied, the game is said to be supermodular, while when (23b) is satisfied, the game is said to be submodular. We can use best response to converge to NE [33].

**4.2.2. Game among CRPs.** A basic modeling assumption in this paper is that each CRP behaves rationally in a self-interested manner. Each one wants to adjust its strategy to maximize its own payoff. We model the computing resource management in mobile crowdsourcing as a game among CRPs:

(1) *Players.* All CRPs in the mobile crowdsourcing system are the players in the game.

(2) *Strategies.* The strategy  $s_i$  of the player  $i$  is its computing resource supply.

(3) *Payoff Functions.* Taking (22) into (15), we have the payoff function of each CRP:

$$P_i(s_i, \mathbf{s}_{-i}) = U' \left( \sum_{i=1}^n s_i \right) s_i - C_i(s_i). \quad (24)$$

Taking (1) and (2) into (24), we have,  $\forall i \in N$ ,

$$\frac{\partial^2 P_i(s)}{\partial s_i \partial s_j} = -\alpha < 0 \quad \forall j \neq i \in N. \quad (25)$$

Based on Theorem 1, NE is considered to be the solution of the game. Meanwhile, the game corresponds to the submodular game, and we can use best response to converge to NE. Best response allows that, at each iteration, each player adapts its strategy to the strategies of others to maximize its own payoff. We design the best response algorithm as follows.

- (1) Initial condition: each player chooses a random strategy.
- (2) Adaption condition: each player chooses an optimal strategy according to the strategies of others to improve its own payoff:

$$s_i^* = \arg \max_{s_i \in S_i} P_i(s_i, \mathbf{s}_{-i}). \quad (26)$$

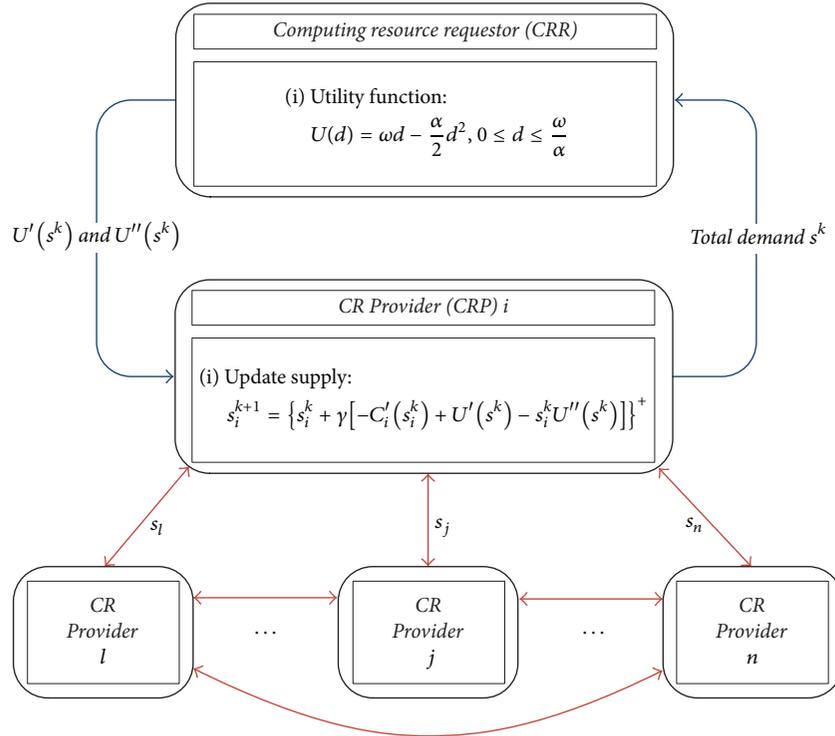


FIGURE 5: Interaction among CRR and each CRP.

Note that, at each iteration, each player updates its strategy while the others keep their strategies fixed.

- (3) Repeat (2) until each player does not revise its strategy.

**4.2.3. Distributed Solution.** Note that (26) is a concave maximization problem, which can be solved by convex optimization techniques in a centralized way. However, the arising challenge is that each CRP needs to know the exact utility function of the CRR. Since such information is private and the CRR does not want to reveal any, each CRP may not have sufficient information to solve problem (26). In order to preserve the CRR's privacy, it is possible to approach the optimal strategy of the optimization problem (26) in a distributed way [34–36]:

- (1) Each CRP shares the information of its current computing resource supply  $s_i$  among others.
- (2) The CRP  $i$  begins with any initial computing resource supply  $s_i^k \geq 0$  and aggregates the total supply of all CRPs, that is,  $s^k \triangleq s_i^k + \sum_{j=1, j \neq i}^n s_j^k$ , and sends it to the CRR.
- (3) On receiving the total supply, the CRR calculates the value of  $U'(s^k)$  and  $U''(s^k)$  and feeds them back to the CRP  $i$ .

- (4) On receiving the feedback, the CRP  $i$  updates its computing resource supply  $s_i^{k+1}$  for the next iteration using the following gradient projection method:

$$s_i^{k+1} = \left[ s_i^k + \gamma \frac{\partial P_i(s_i^k, \mathbf{s}_{-i}^k)}{\partial s_i^k} \right]^+ \quad (27)$$

$$= \left\{ s_i^k + \gamma \left[ -C'_i(s_i^k) + U'(s^k) + s_i^k U''(s^k) \right] \right\}^+.$$

- (5) Repeat from (2) to (4) until the CRP  $i$  does not revise its computing resource supply.

The interaction among the CRR and each CRP is shown in Figure 5.

## 5. Results and Discussion

We provide numerical examples to evaluate the proposed distributed approach.

**5.1. One-CRR and One-CRP Case.** Consider a mobile crowd-sourcing system with one CRR and one CRP. The simulation parameters are set as  $\omega = 3$ ,  $\alpha = 0.5$ ,  $a = 0.1$ ,  $b = 0.5$ , and  $c = 0$ . That is, we assume that the CRR has the utility function  $U(d) = 3d - (0.5/2)d^2$ , while the CRP has the cost function  $C(s) = 0.1s^2 + 0.5s$ . Firstly, in Figure 6, we fix the step size at  $\gamma = 0.1$ , while setting the initial price at  $p^1 = 0$  and  $p^1 = 3$ , respectively. It is shown that in both cases the price converges

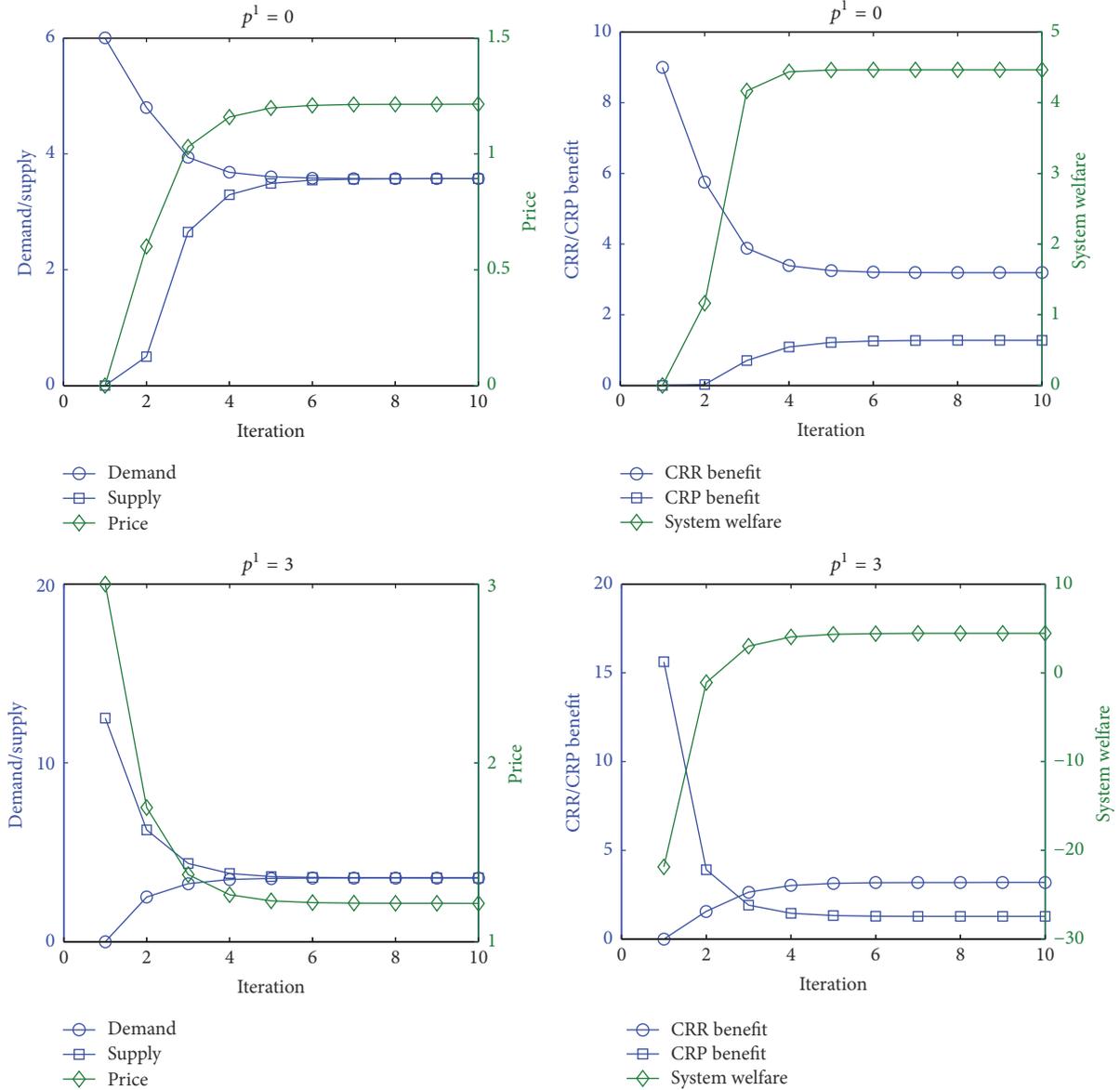


FIGURE 6: Computing resource management in a mobile crowdsourcing system of the one-CRR and one-CRP case.

to the global optimum which balances between the supply and demand. The system welfare achieves the best with the convergence of the price, where the locally optimal solution of both the CRR and CRP becomes globally optimal at the converged price.

Next, in Figure 7(a),  $\gamma$  is fixed while  $p^1$  varies from 0 to 3, to study how the initial price impacts the convergence performance. The figure indicates that the price will finally converge to the equilibrium regardless of any initial value, although the convergence rates may be different. Similarly, in Figure 7(b),  $p^1$  is fixed while  $\gamma$  varies from 0.025 to 0.25, to study the impact of the step size on the price convergence. We find that the smaller the step size, the slower the convergence rate, while the larger the step size, the faster the convergence rate, but the system may only approach

within a certain neighborhood of the equilibrium. This is a general characteristic of any gradient based method. In practice, we can first choose a large step size to ensure fast convergence and subsequently reduce the step size once the price starts oscillating around a certain value.

**5.2. One-CRR and Multi-CRP Case.** Consider a mobile crowdsourcing system with one CRR and three CRPs. The simulation parameters for the CRR are the same as those in Section 5.1, while the cost functions of different CRPs are assumed to be different. The initial price and step size are fixed at  $p_1 = 3$  and  $\gamma = 0.05$ .

**5.2.1. Without Interaction among CRPs.** In Figure 8, we first set  $b = 0.5$  while  $a$  varies from 0.1 to 0.3 for different CRPs.

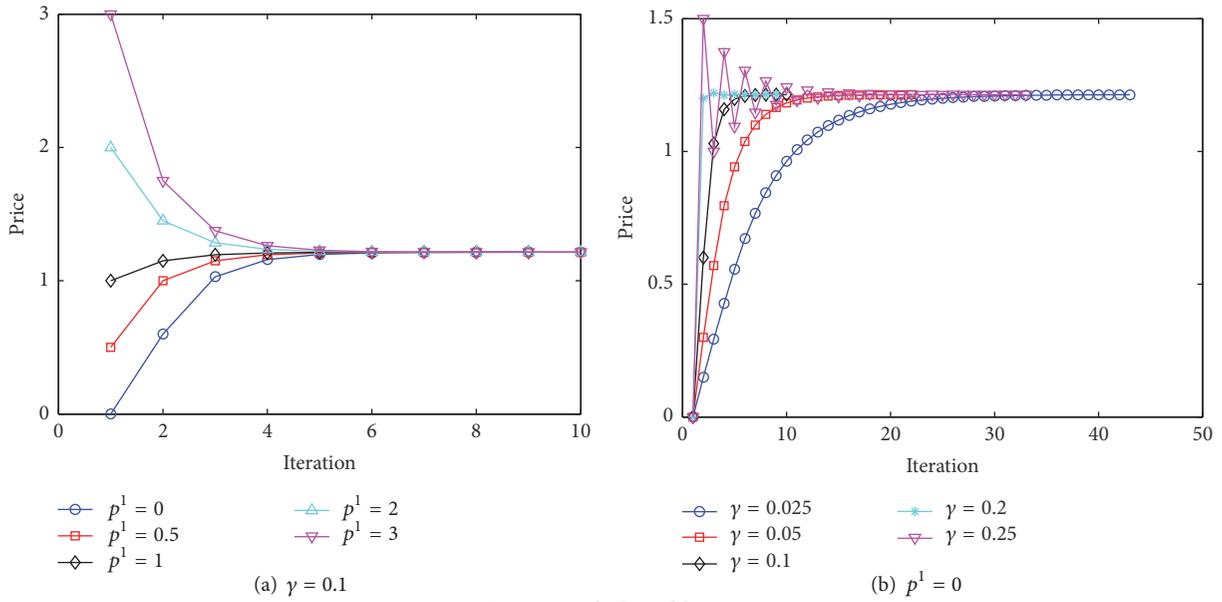


FIGURE 7: Impact of adjustable parameters.

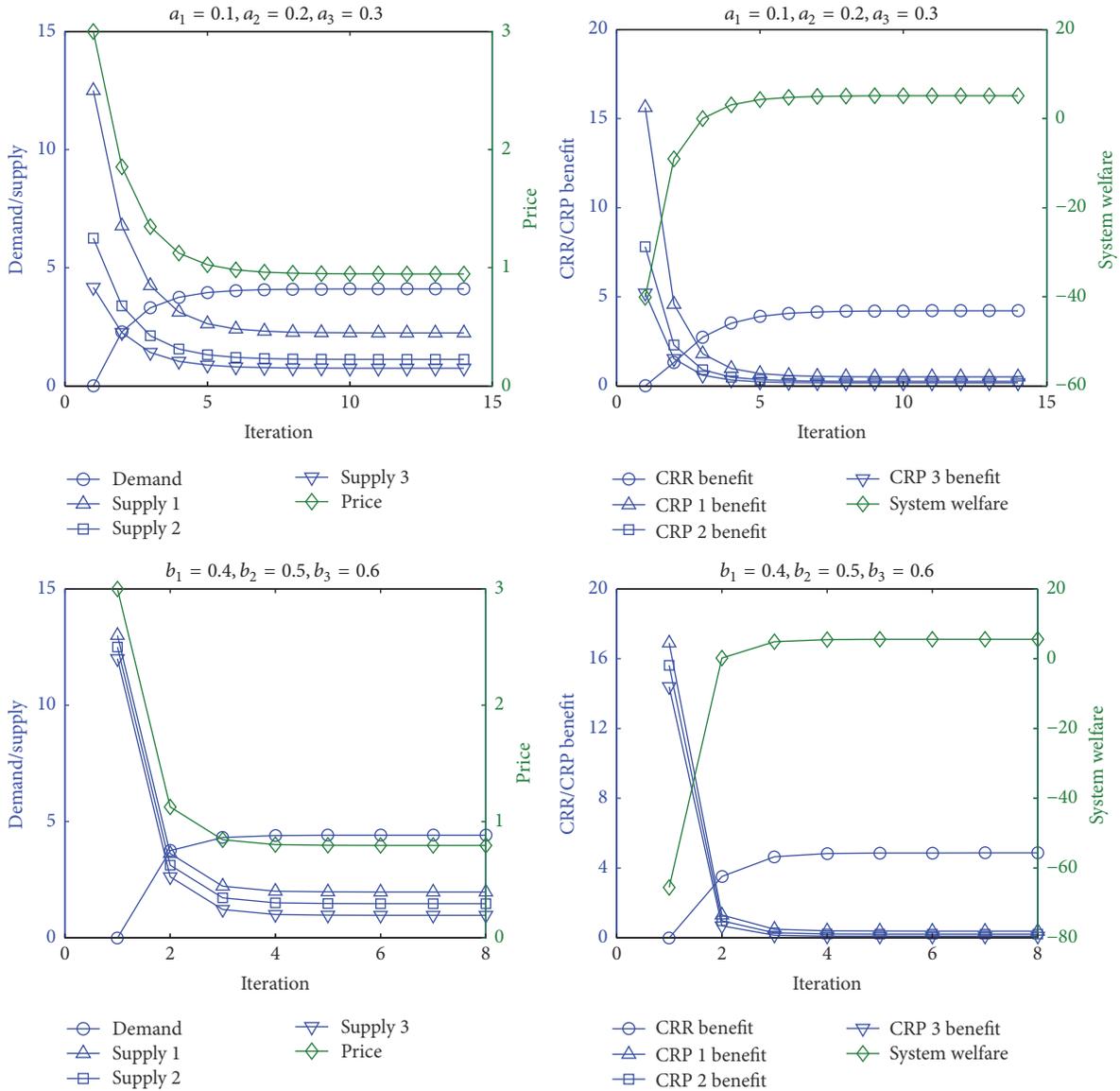


FIGURE 8: Computing resource management in a mobile crowdsourcing system of the one-CRR and multi-CRP case.

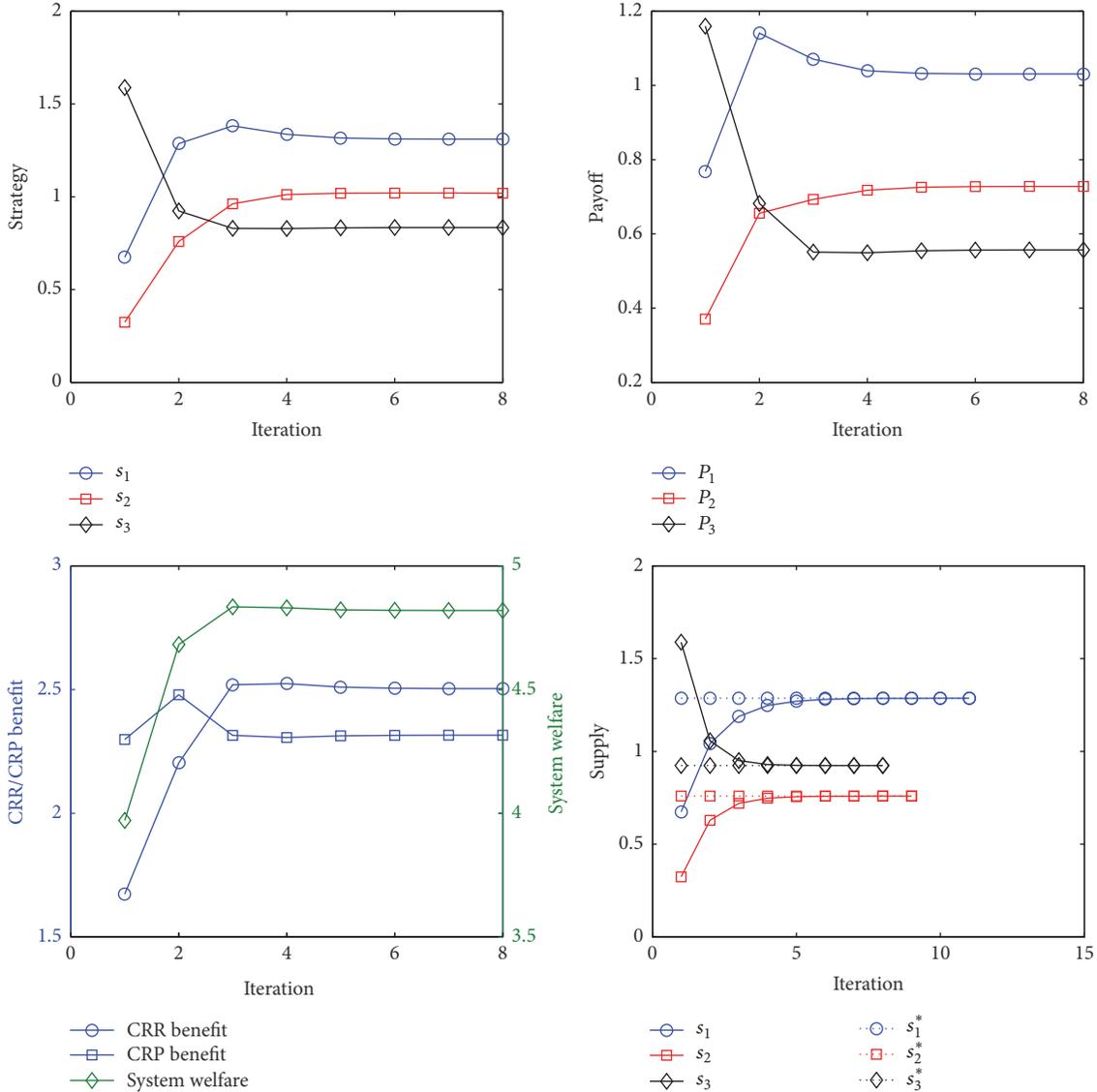


FIGURE 9: CRP game in a mobile crowdsourcing system of the one-CRR and multi-CRP case ( $a_1 = 0.1, a_2 = 0.2, a_3 = 0.3$ ).

It is shown that the larger the value of  $a$ , the less computing resource the CRP will supply. Then we set  $a = 0.1$  while  $b$  varies from 0.4 to 0.6 for different CRPs. The figure indicates that the larger the value of  $b$ , the less computing resource the CRP will supply. The system welfare achieves the best with the convergence of the price, where the locally optimal solution of the CRR and each CRP becomes globally optimal under the converged price.

5.2.2. *With Interaction among CRPs.* In Figure 9, we set  $b = 0.5$  while  $a$  varies from 0.1 to 0.3 for different CRPs. It is shown that the larger the value of  $a$ , the less computing resource the CRP will supply. Initially the supplies of the three CRPs are random, so their payoffs are very low. Then, at each iteration, each CRP chooses the most selfish strategy according to the strategies of others to improve its own payoff. We can see from the figure that the best response algorithm guarantees that the CRPs' strategies and payoffs converge to NE. The

convergence rate is considerably fast, which is desirable for the real-time requirement of mobile crowdsourcing. At NE, each CRP achieves the best payoff. Through the CRP game, each CRP can strategically adjust its supply to improve its individual benefit. We also evaluate the distributed approach to (26). The initial supplies of the three CRPs are the same as those in the CRP game and the step size is fixed at  $\gamma = 0.5$ . It is shown that the supplies of all CRPs converge to the optimal solution solved by a centralized way.

In Figure 10, we set  $a = 0.1$  while  $b$  varies from 0.4 to 0.6 for different CRPs. The figure indicates that the larger the value of  $b$ , the less computing resource the CRP will supply. We can see from the figure that the best response algorithm guarantees that the CRPs' strategies and payoffs converge to NE. At NE, each CRP achieves the best payoff. We also evaluate the distributed approach to (26). The initial supplies of the three CRPs are the same as those in the CRP game and the step size is fixed at  $\gamma = 0.5$ . It is shown that the

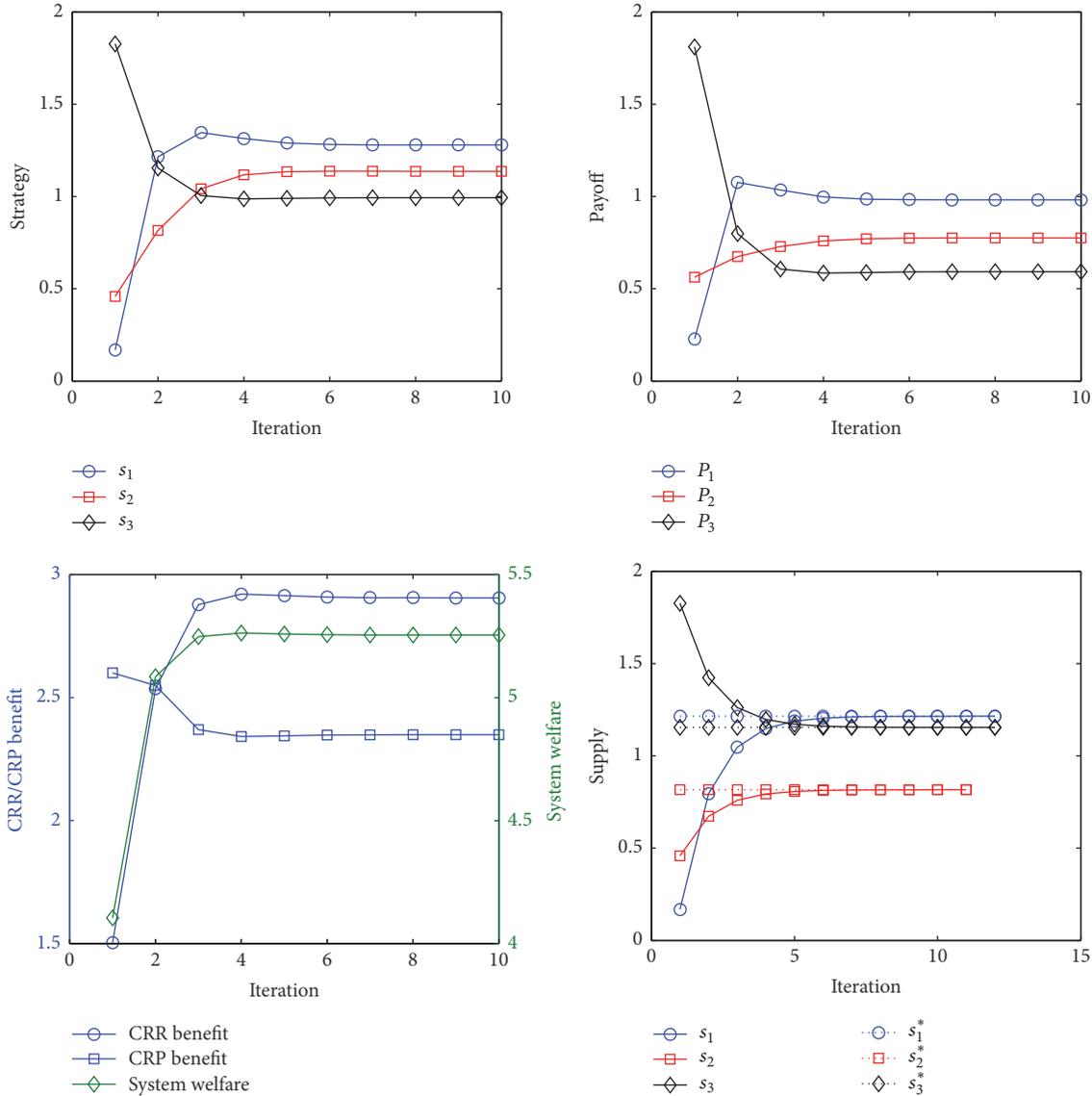


FIGURE 10: CRP game in a mobile crowdsourcing system of the one-CRR and multi-CRP case ( $b_1 = 0.4$ ,  $b_2 = 0.5$ ,  $b_3 = 0.6$ ).

supplies of all CRPs converge to the optimal solution solved by a centralized way.

In Section 4.1, we solve the computing resource management problem in mobile crowdsourcing while assuming that there is no interaction among CRPs. We refer to it as “solution 1” (without interaction). In Section 4.2, we assume the interaction among CRPs and formulate the CRP game to solve the computing resource management problem in mobile crowdsourcing. We refer to it as “solution 2” (CRP game). In Figure 11, we compare the performance of these two solutions. We first set  $b = 0.5$  while  $a$  varies from 0.1 to 0.3 for different CRPs. Then we set  $a = 0.1$  while  $b$  varies from 0.4 to 0.6 for different CRPs. It is shown that in both cases each CRP benefit in solution 1 outperforms that in solution 2, while the CRR benefit and system welfare are a little lower than those in solution 2. The reason is that the algorithm in Section 4.1 takes the system welfare into the first consideration, while the CRP

game in Section 4.2 guarantees that each CRP is selfish and rational who only wants to maximize its individual benefit, regardless of the CRR benefit and system welfare.

## 6. Conclusions

In this paper, we propose an optimal real-time pricing strategy for computing resource management in mobile crowdsourcing, which is based on utility maximization. It can be implemented in a distributed manner such that the real-time interaction among the CRR and CRPs is coordinated through a limited number of message exchanges. We show that there exist real-time prices that can align individual optimality with systematic optimality. We also take account of the interaction among CRPs and formulate the computing resource management as a game with Nash equilibrium achievable via best response. Simulation results demonstrate that, by using our proposed optimization-based real-time

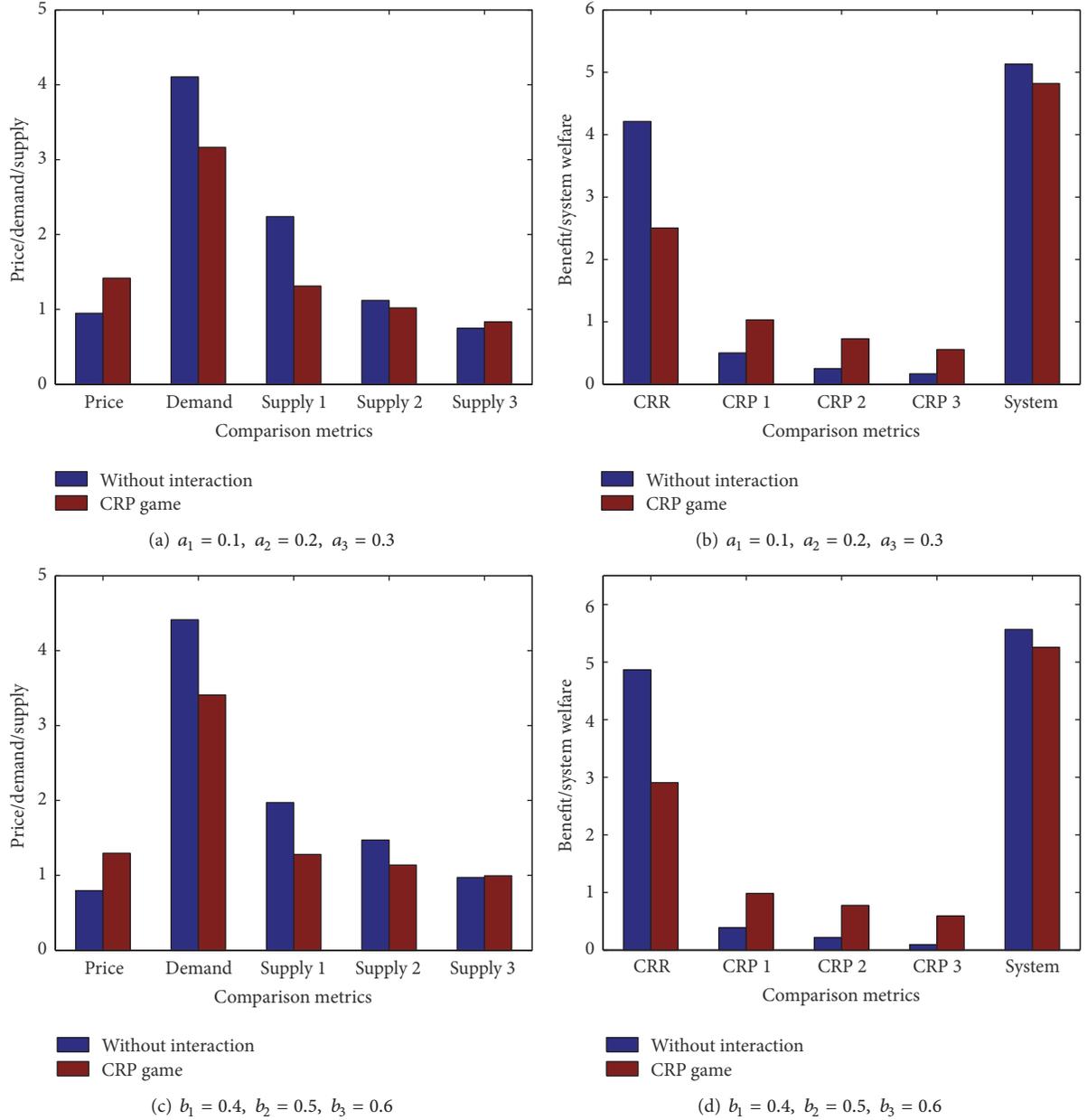


FIGURE 11: Comparison between solutions in Sections 4.1 and 4.2.

pricing strategy, not only the CRR but also CRPs will benefit. The coordinator in mobile crowdsourcing can thus use the optimal real-time pricing strategy to manage computing resources towards the benefit of the overall system.

### 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 State Key Laboratory of Industrial Control Technology, Zhejiang University, China.

### References

- [1] X. Li, R. Lu, X. Liang, X. Shen, J. Chen, and X. Lin, "Smart community: an internet of things application," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 68–75, 2011.
- [2] X. Hu, T. H. S. Chu, V. C. M. Leung, E. C.-H. Ngai, P. Kruchten, and H. C. B. Chan, "A survey on mobile social networks: applications, platforms, system architectures, and future research directions," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1557–1581, 2014.
- [3] Z. Ning, F. Xia, X. Hu, Z. Chen, and M. S. Obaidat, "Social-oriented adaptive transmission in opportunistic internet of smartphones," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 810–820, 2017.

- [4] M. La Polla, F. Martinelli, and D. Sgandurra, "A survey on security for mobile devices," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 446–471, 2013.
- [5] S. Abolfazli, Z. Sanaei, E. Ahmed, A. Gani, and R. Buyya, "Cloud-based augmentation for mobile devices: motivation, taxonomies, and open challenges," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 337–368, 2014.
- [6] Z. Ning, L. Liu, F. Xia, B. Jedari, I. Lee, and W. Zhang, "CAIS: a copy adjustable incentive scheme in community-based socially aware networking," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 3406–3419, 2017.
- [7] 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.
- [8] M. R. Rahimi, J. Ren, C. H. Liu, A. V. Vasilakos, and N. Venkatasubramanian, "Mobile cloud computing: a survey, state of art and future directions," *Mobile Networks and Applications*, vol. 19, no. 2, pp. 133–143, 2014.
- [9] R. Deng, R. Lu, C. Lai, T. H. Luan, and H. Liang, "Optimal Workload Allocation in Fog-Cloud Computing Toward Balanced Delay and Power Consumption," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1171–1181, 2016.
- [10] N. Fernando, S. W. Loke, and W. Rahayu, "Mobile cloud computing: a survey," *Future Generation Computer Systems*, vol. 29, no. 1, pp. 84–106, 2013.
- [11] F. H. Fitzek and M. D. Katz, *Mobile Clouds: Exploiting Distributed Resources in Wireless, Mobile and Social Networks*, John Wiley & Sons, 2013.
- [12] F. Tong, R. Zhang, and J. Pan, "One Handshake Can Achieve More: An Energy-Efficient, Practical Pipelined Data Collection for Duty-Cycled Sensor Networks," *IEEE Sensors Journal*, vol. 16, no. 9, pp. 3308–3322, 2016.
- [13] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: current state and future challenges," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 32–39, 2011.
- [14] X. Hu, T. H. S. Chu, H. C. B. Chan, and V. C. M. Leung, "Vita: a crowdsensing-oriented mobile cyber-physical system," *IEEE Transactions on Emerging Topics in Computing*, vol. 1, no. 1, pp. 148–165, 2013.
- [15] H. Ma, D. Zhao, and P. Yuan, "Opportunities in mobile crowd sensing," *IEEE Communications Magazine*, vol. 52, no. 8, pp. 29–35, 2014.
- [16] B. Guo, Z. Wang, Z. Yu et al., "Mobile crowd sensing and computing: the review of an emerging human-powered sensing paradigm," *ACM Computing Surveys*, vol. 48, no. 1, article 7, 2015.
- [17] A. Faggiani, E. Gregori, L. Lenzini, V. Luconi, and A. Vecchio, "Smartphone-based crowdsourcing for network monitoring: opportunities, challenges, and a case study," *IEEE Communications Magazine*, vol. 52, no. 1, pp. 106–113, 2014.
- [18] J. Ren, Y. Zhang, K. Zhang, and X. Shen, "Exploiting mobile crowdsourcing for pervasive cloud services: challenges and solutions," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 98–105, 2015.
- [19] Y. Zhang, C. Jiang, L. Song, M. Pan, Z. Dawy, and Z. Han, "Incentive mechanism for mobile crowdsourcing using an optimized tournament model," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 4, pp. 880–892, 2017.
- [20] S. S. Manvi and G. Krishna Shyam, "Resource management for Infrastructure as a Service (IaaS) in cloud computing: a survey," *Journal of Network and Computer Applications*, vol. 41, no. 1, pp. 424–440, 2014.
- [21] H. Zhou, B. Liu, Y. Liu et al., "A cooperative matching approach for resource management in dynamic spectrum access networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 2, pp. 1047–1057, 2014.
- [22] R. Deng, Y. Zhang, S. He, J. Chen, and X. Shen, "Maximizing network utility of rechargeable sensor networks with spatiotemporally coupled constraints," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 5, pp. 1307–1319, 2016.
- [23] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.
- [24] P. Samadi, H. Mohsenian-Rad, R. Schober, and V. W. S. Wong, "Advanced demand side management for the future smart grid using mechanism design," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1170–1180, 2012.
- [25] Y. S. Yun, Y. Xia, B. Behdani, and J. C. Smith, "Distributed algorithm for lifetime maximization in a delay-tolerant wireless sensor network with a mobile sink," *IEEE Transactions on Mobile Computing*, vol. 12, no. 10, pp. 1920–1930, 2013.
- [26] A. Safdarian, M. Fotuhi-Firuzabad, and M. Lehtonen, "A distributed algorithm for managing residential demand response in smart grids," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2385–2393, 2014.
- [27] N. Trichakis, A. Zymnis, and S. Boyd, "Dynamic Network Utility Maximization with Delivery Contracts," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 2907–2912, 2008.
- [28] J. Xu, J. Wang, Y. Zhu et al., "Cooperative distributed optimization for the hyper-dense small cell deployment," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 61–67, 2014.
- [29] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks," *IEEE Signal Processing Magazine*, vol. 31, no. 6, pp. 45–55, 2014.
- [30] C. Li, X. Yu, W. Yu, T. Huang, and Z.-W. Liu, "Distributed event-triggered scheme for economic dispatch in smart grids," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 5, pp. 1775–1785, 2016.
- [31] R. B. Myerson, *Game Theory: Analysis of Conflict*, Harvard University Press, 1991.
- [32] D. Bauso, "Game Theory with Engineering Applications," *Society for Industrial and Applied Mathematics (SIAM)*, 2016.
- [33] Z. Han, D. Niyato, W. Saad, T. Basar, and A. Hjørungnes, *Game Theory in Wireless and Communication Networks*, Cambridge University Press, 2011.
- [34] N. Li and J. R. Marden, "Designing games for distributed optimization," *IEEE Journal on Selected Topics in Signal Processing*, vol. 7, no. 2, pp. 230–242, 2013.
- [35] E. R. Stephens, D. B. Smith, and A. Mahanti, "Game Theoretic Model Predictive Control for Distributed Energy Demand-Side Management," *IEEE Transactions on Smart Grid*, vol. 6, no. 3, pp. 1394–1402, 2015.
- [36] J. Zhang, D. Qi, and G. Zhao, "A new game model for distributed optimization problems with directed communication topologies," *Neurocomputing*, vol. 148, pp. 278–287, 2015.